City Research Online

City, University of London Institutional Repository

Citation: Mukherjee, S. N. (1983). Nonlinear response of embedded structures under transient dynamic load. (Unpublished Doctoral thesis, The City University)

This is the accepted version of the paper.

This version of the publication may differ from the final published version.

Permanent repository link: https://openaccess.city.ac.uk/id/eprint/35363/

Link to published version:

Copyright: City Research Online aims to make research outputs of City, University of London available to a wider audience. Copyright and Moral Rights remain with the author(s) and/or copyright holders. URLs from City Research Online may be freely distributed and linked to.

Reuse: Copies of full items can be used for personal research or study, educational, or not-for-profit purposes without prior permission or charge. Provided that the authors, title and full bibliographic details are credited, a hyperlink and/or URL is given for the original metadata page and the content is not changed in any way.

City Research Online: http://openaccess.city.ac.uk/ publications@city.ac.uk

TO MY WIFE AND TO MY MOTHER

NONLINEAR RESPONSE OF EMBEDDED STRUCTURES UNDER TRANSIENT DYNAMIC LOAD

Somendra Nath Mukherjee

PH. D

****

The City University

Northampton Square

LONDON EC1V OHB

Department of Mechanical Engineering

ACKNOWLEDGEMENT

The author gratefully acknowledges the kind permission given by Brown, Boveri & Company Limited, Baden, Switzerland, to publish and reproduce illustrations, diagrams and computor printouts for this thesis.

DECLARATION

The University Librarian is granted the powers of discretion to allow the thesis to be copied in whole or in part without further reference to the author.

PREFACE

Vol. 2 of this thesis consists of two parts. Part 1 is the main NONSAP program and part 2 is the subroutine VELCOR, which is linked to the main body of NONSAP.

In subroutine VELCOR unwanted wave reflection problem from the artificial boundary is treated. Besides suppression of unwanted reflected waves, other associated problems inherent to the problem solution such as corrections to displacement, stress and strain vectors are carried out with adequate numerical accuracy (single precision).

VELCOR is presented seperately as part 2 of the program in order to show author's major contribution within the main body of NONSAP program. No computation with SR VELCOR alone is possible, because nodal point information, and element library are obtained from NONSAP, through common and data blocks.

TABLES OF CONTENTS

SUBJECT	SECTION	PAGE
Programm organisation	1	1
Equation solution	2	9
Step by Step Solution	3	13
Element library	4	19
Material model	5	25

Appendix:

Flow chart of Main

The Computer Program

Data Input

ABSTRACT

The current version of the computer program NONSAP for linear and nonlinear, static and dynamic finite element analysis is described. The solution capabilities, the numerical techniques used, the finite element library, the logical construction of the program and storage allocations are discussed.

The current version of NONSAP is extensively modified, revised and better organised by the author of this thesis than the original version (Univ. of Berkely). The 2-D and 3-D finite element models are redefined, in order to extend the scope of non-linear analysis. Reduced integration scheme are made available for the 3-D element. A stable integration scheme for the step by step analysis is incorporated. It is advised that for the wave propagation problem explicit integration scheme should be used to minimize the solution oscillation.

Following subroutines are corrected in NONSAP program

subroutine	correction
ADDCM	Add nodal dampers to global stiffness matrix
CDMOD	Effect of gravity load is added
MAXAEX	New subroutine is added which allocate space according to number of equations
READID	To generate identification array of boundary conditions (new program)
TDFE	2-D finite element library is improved for stress output
THDFE	3-D finite element library. Shape functions, and weights for Gauss integration are corrected
FREQ	New subroutine is added to calculate lowest Eigenvalues by Rayleigh method

1. PROGRAM ORGANIZATION

The complete solution process in program NONSAP is divided into three distinct phases:

1) Input phase

The input phase consists of three steps:

a) The control information and the nodal point input data are read and generated by the program. In this phase the equation numbers for the active degrees of freedom at each nodal point are establised.

b) The externally applied load vectors for each time (load) step are calculated and stored on tape (or other low-speed storage).

c) The element data are read and generated, the element connection arrays are calculated and all element information is stored on tape.

2) Assemblage of Constant Structure Matrices

Before the solution is carried out, the linear structure stiffness, mass and damping matrices are assembled and stored on tape (or other low-speed storage). In addition, the effective linear structure stiffness matrix is calculated and stored (see table 4.1, vol 1)

3) Step-by-Step Solution

During this phase the solution of Eq. 7.1.11 is obtained at all time points. In addition to the displacement, velocity, and acceleration vectors (whichever applicable), the element stresses are calculated and printed. Before the time integration

is performed, the lowest frequencies and corresponding mode shapes may be calculated.

The details of the step-by-step solution are presented in Table 7.1 of Vol. 1.

It need be noted that these basic steps are independent of the element type used and are the same for either a static or dynamic analysis. However, only those matrices actually required in the analysis are assembled. For example, no mass and damping matrices are calculated in a static analysis.

Program NONSAP is an in-core solver and the high-speed storage capacity of the program is determined by the maximum storage that is required during the three phases. Figures 1 to 3 show the dynamic storage allocations used in each phase. We note that, in general, maximum high speed storage is required during the step-by-step solution. However, in some cases the storage required during the input phase may govern the system size that can be solved.

Figures 1 to 3 show that the lowest high speed storage locations are reserved throughout the solution for element group information. For the analysis, the finite elements of the complete assemblage need be divided into element groups according to their type, the nonlinear formulation (see Sec. 7, Vol. 1), and the material models used (Appendix A). One element group must consist of the same type of elements, must use one nonlinear formulation and only one specific material model. The data pertaining to each individual element group need fit into the NUMEST storage locations, Fig. 1. Therefore, the minimum that used are pointed out. (Table X.1 in the Appendix summarizes the formulations with which the material models can be used).

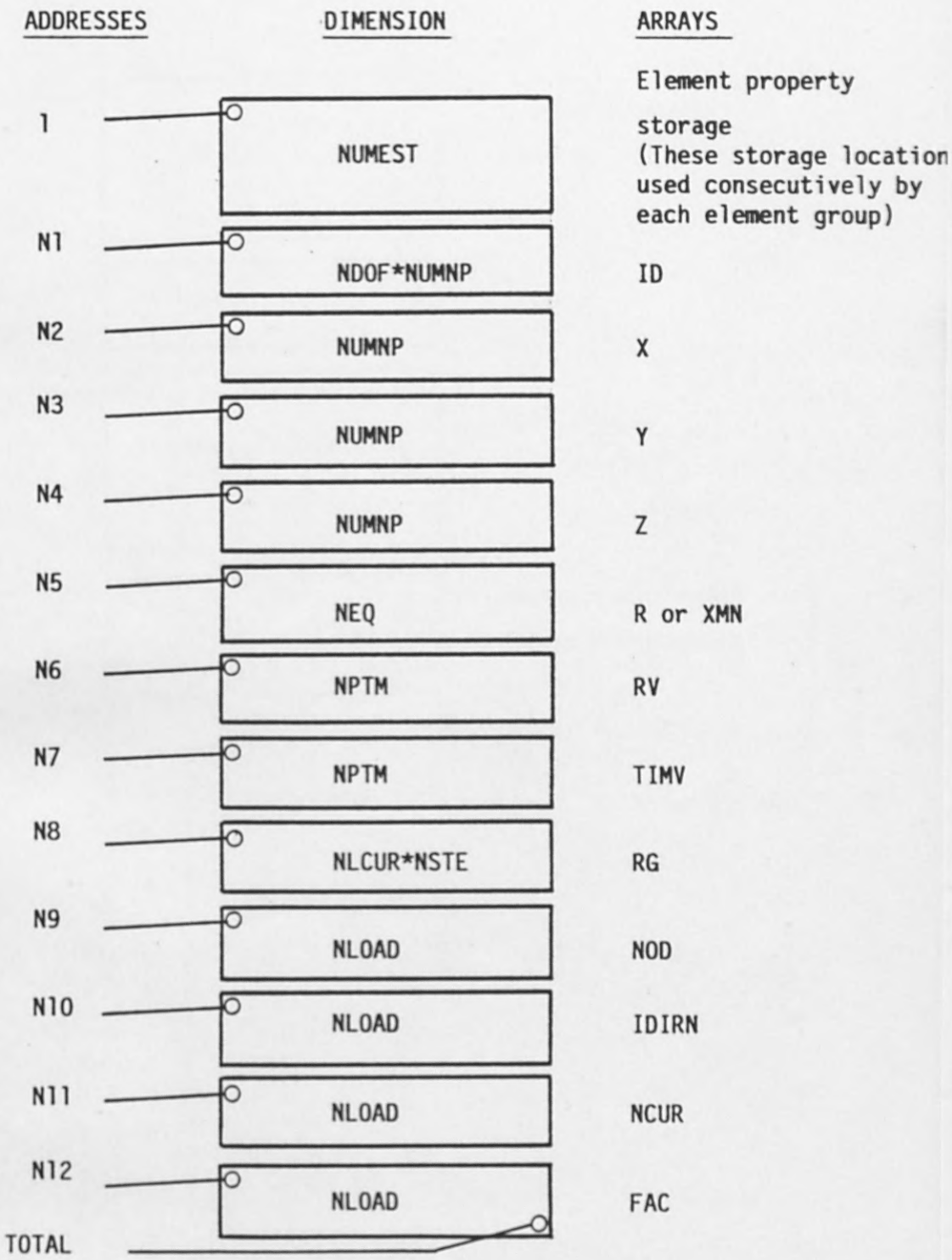


FIGURE 1 STORAGE ALLOCATION DURING INPUT PHASE

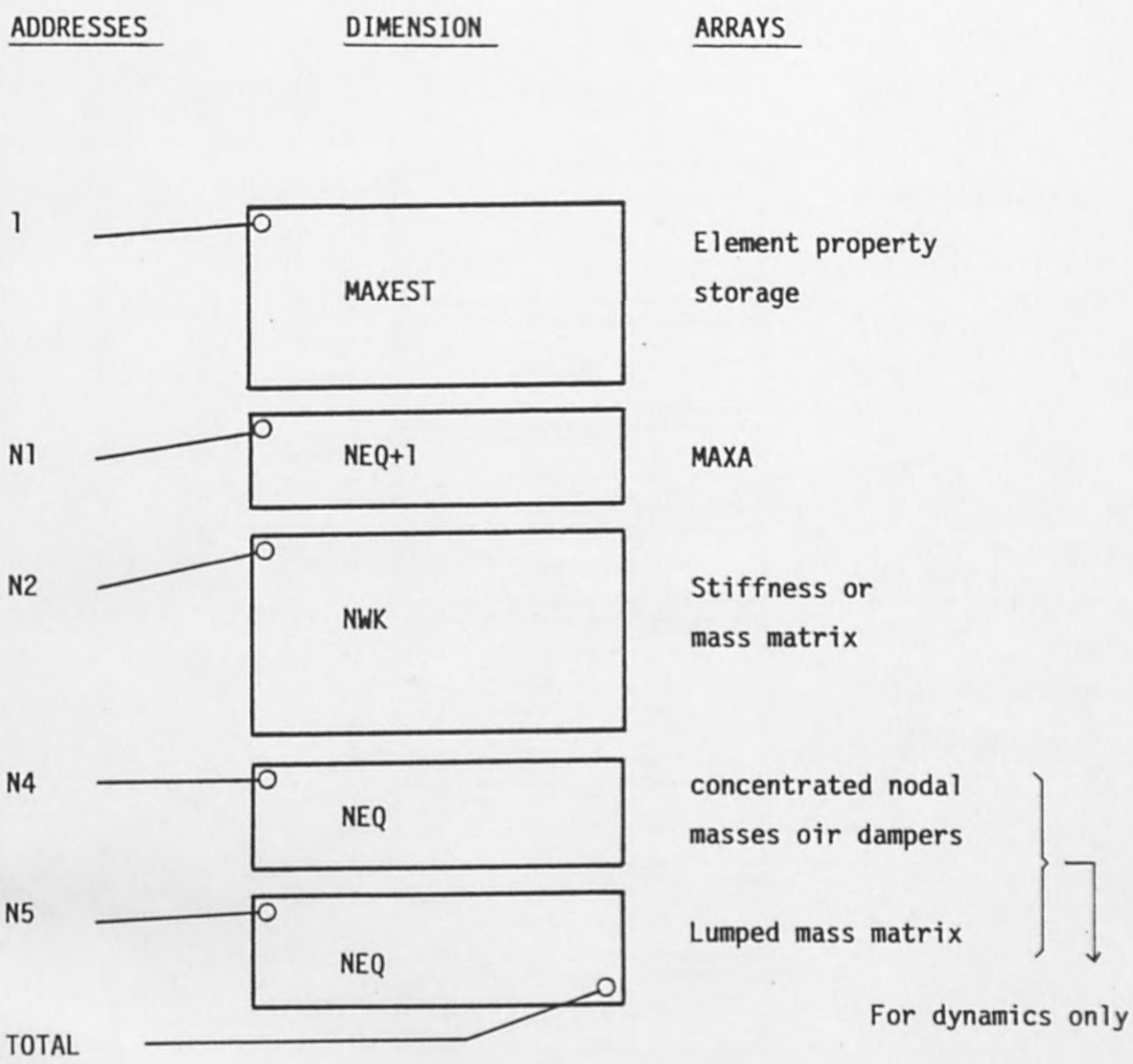
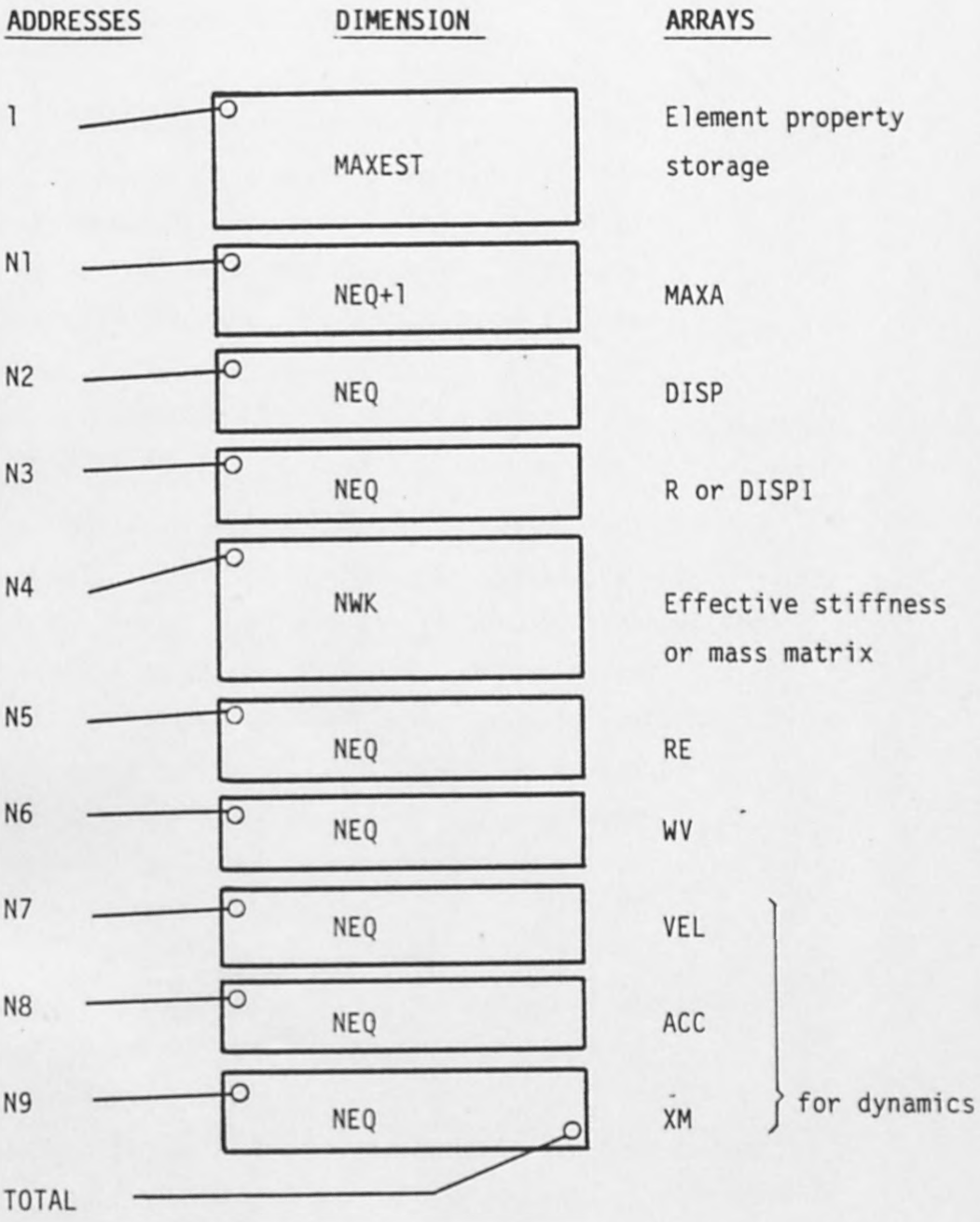


FIGURE 2 STORAGE ALLOCATION DURING MATRIX ASSEMBLAGE PHASE



Note: N7 = N5 + NDOF*NUMNP if NDO*NUMNP 2*NEQ

FIGURE 3 STORAGE ALLOCATION DURING TIME INTEGRATION

Isotropic and Orthotropic Linear Elastic Material

The stress-strain relationships are defined by means of the constant Young's moduli and Poisson's ratios.

Mooney-Rivlin Material Model

A hyperelastic incompressible material model is available for the analysis of rubber-like materials. The stress-strain relationship is defined using the Mooney-Rivlin material constants. In NONSAP the model can only be used in plane stress analysis.

Elastic-Plastic Material Models

Elastic-plastic analysis using a plastic potential function can be carried out. The plasticity relations available are those based on the use of the von Mises yield condition and the Drucker-Prager yield condition. Both forms of describing material behavior have been employed extensively in practice. Using the von Mises criterion, linear iso-hardening can be assumed. In analysis using the Drucker-Prager yield condition, the material is assumed to be elastic perfectly plastic.

NUMEST should be specified is equal to the locations required to store the data pertaining to any one of the elements.

The use of element groups reduces input-output transfers during the solution process, since the data of the elements is retrieved in blocks during the solution and element stress calculations (see Sec. 7, Vol. 1). Usually, NUMEST is some reasonalbe fraction of the total number of high speed storage locations available, and is not reset for each problem. During the input phase the program calculates the exact number of high speed storage locations required for each element group, and NUMEST is reset to MAXEST, which is the actual maximum of locations needed, see Figs. 2 and 3. Therefore, an optimum of high speed storage allocation is obtained during the step-by-step solution. Figure 4 shows the tape storage used for the element group information.

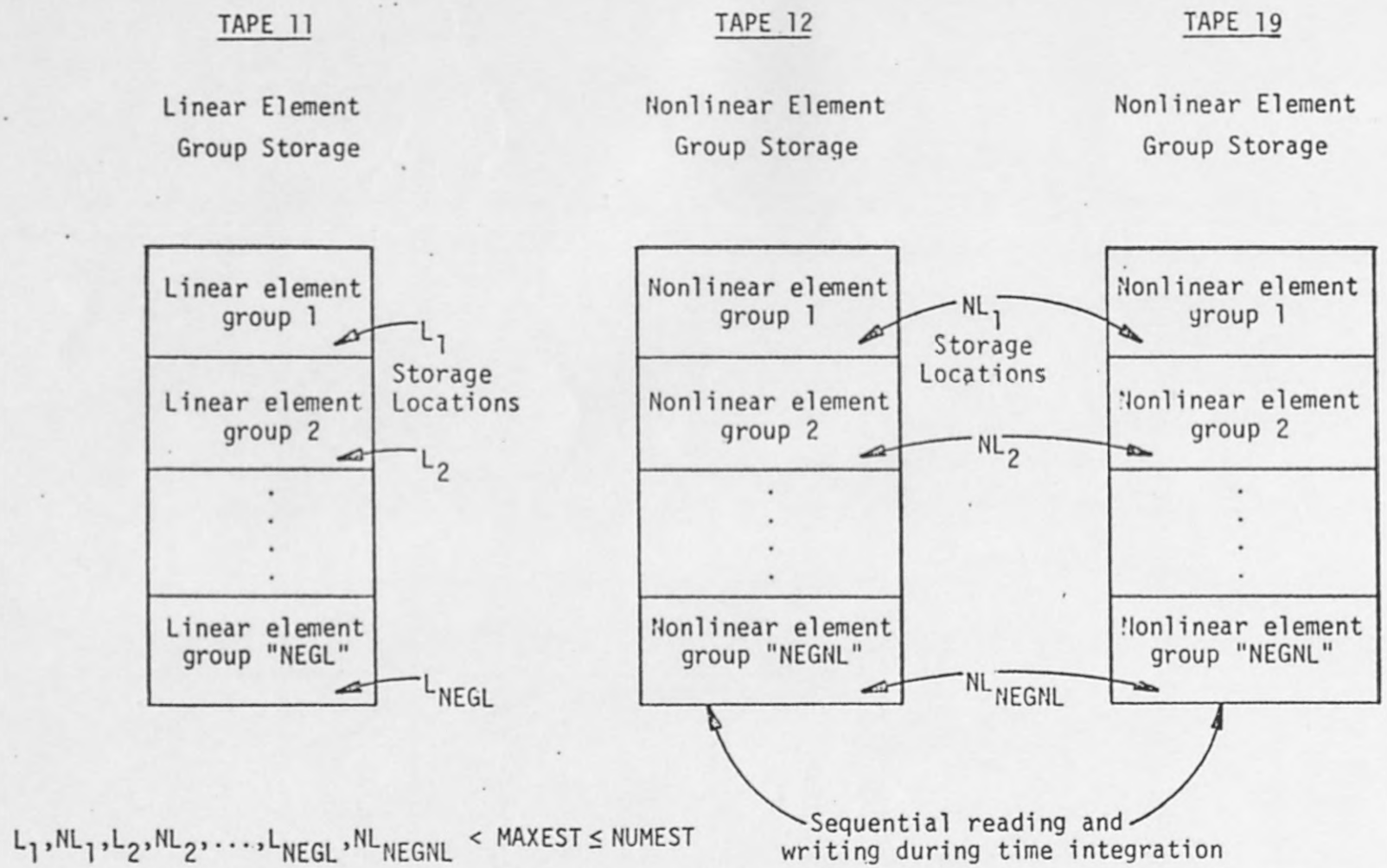


FIGURE 4 AUXILIARY STORAGE ORGANIZATION FOR ELEMENT GROUP INFORMATION

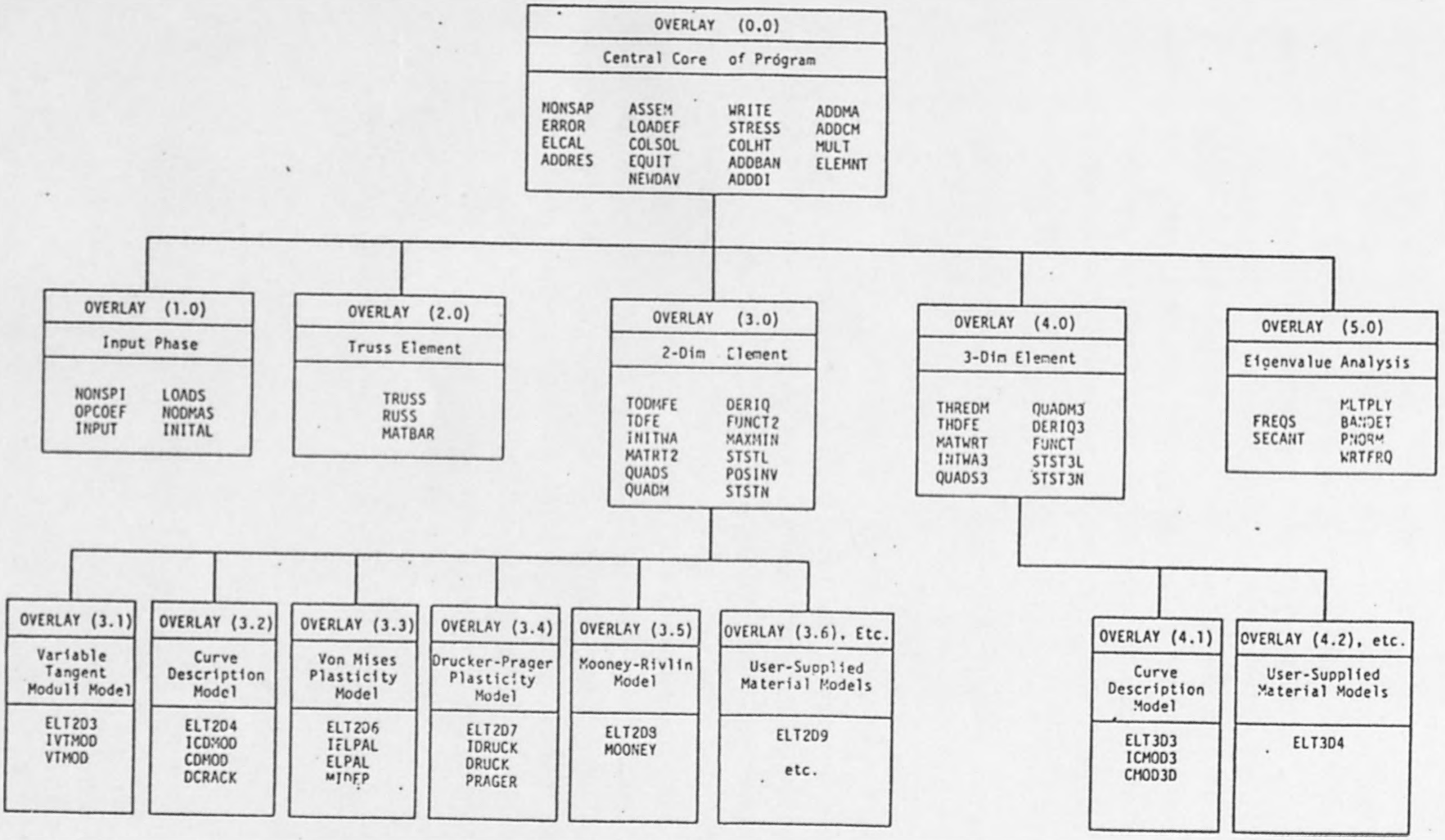


FIGURE 5. OVERLAY STRUCTURE OF NONSAP

In program NONSAP a compacted storage scheme is used in which all structure matrices are stored as one-dimensional arrays, and only the elements below the skyline of a matrix are processed (Ref. 4, Vol. 1). Figure 6 shows, as an example, the element pattern in a typical stiffness matrix before and after triangularization. It should be noted that, in general, zero elements within the skyline do not remain zero during the equation solution and must be stored, whereas all elements outside the skyline do not need to be considered. Therefore, by storing and processing in the equation solution only the elements within the skyline, a minimum number of high speed storage locations is used.

2. EQUATION SOLUTION

The solution of equations is obtained using the linear equation solver COLSOL. This subroutine uses Gauss elimination on the positive definite symmetrical system of equations. The algorithm performs a minimum number of arithmetic operations, since only the elements within the skyline of the matrix are processed, i.e. there are no operations with elements that remain zero during the solution. The algorithm is used in all analysis types, i.e. in linear, nonlinear, static or dynamic analysis, and consists of the LDL^T decomposition of the stiffness matrix (or effective stiffness matrix), and the reduction and back-substitution of the (effective) load vector.

LINEAR ANALYSIS

STATIC ANALYSIS

1) Calculate linear structure stiffness matrix K.

2) Triangularize K and store on tape 17.

DYNAMIC ANALYSIS

1) Calculate linear structure stiffness matrix K.

2) If Rayleigh damping is specified store K on tape 14.

3) If frequency analysis is to be performed, store K on tape 20.

4) Add mass and damping effects to K to obtain linear effective stiffness matrix $\hat{K}$.

5) Triangularize $\hat{K}$ and store on tape 17.

6) Calculate mass matrix M and store on tape 4 (if consistent mass matrix) or tape 17 (if diagonal mass matrix).

7) If concentrated nodal dampers specified, store nodal damping vector C_d on tape 14.

	Tape Layout		Tape Layout
Tape 14:	(not used)	Tape 14:	K (if Rayleigh damping) M (consistent mass case only) C_d (if nodal damping)
Tape 17:	$K = LDL^T$	Tape 17:	$\hat{K} = LDL^T$ M (diagonal mass case only)
Tape 20:	(not used)	Tape 20:	K (for frequency analysis only)

FIGURE 6 ASSEMBLAGE OF CONSTANT STRUCTURE MATRICES

NONLINEAR ANALYSIS

STATIC ANALYSIS

1) Calculate linear structure stiffness matrix K, i.e., that part of the total structure stiffness matrix which corresponds to the linear element groups.

2) Store K on tape 14 and on tape 17.

DYNAMIC ANALYSIS

1) Calculate linear structure stiffness matrix K.

2) Store K on tape 14.

3) If frequency analysis is to be performed, store K on tape 10.

4) Add mass and damping effects to K to obtain linear effective stiffness matrix $\hat{K}$.

5) Store $\hat{K}$ on tape 17.

6) Calculate mass matrix M and store on tape 14 (if consistent mass matrix) or on tape 17 (if diagonal mass matrix).

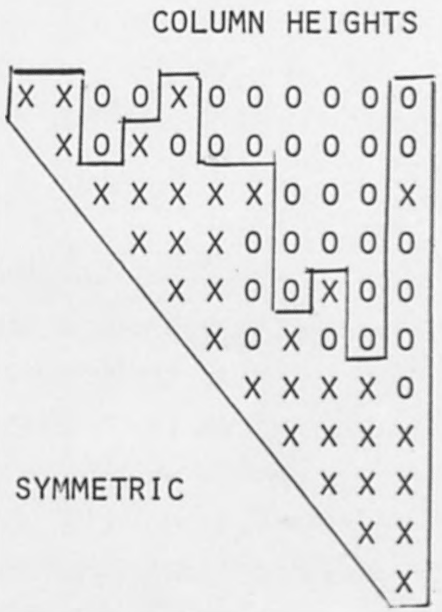
7) If concentrated nodal dampers specified, store nodal damping vector C_d on tape 14.

	Tape Layout		Tape Layout	
Tape 14:	K	Tape 14:	K	
			M	(consistent mass case only)
			C_d	(if nodal damping)
Tape 17:	K	Tape 17:	$\hat{K}$	
			M	(diagonal mass case only)
Tape 20:	(not used)	Tape 20:	K	(for frequency analysis only)

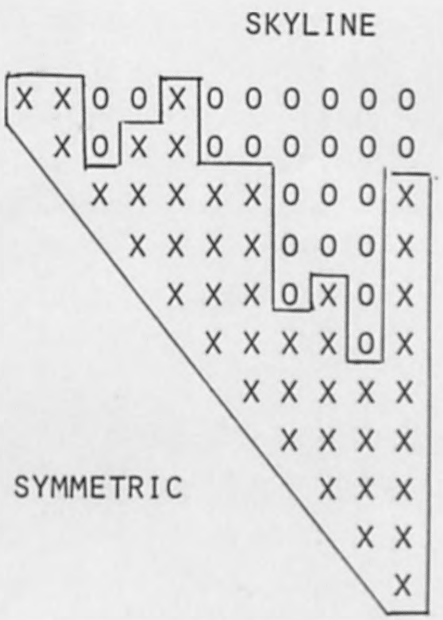
FIGURE 7 (CONT) ASSEMBLAGE OF CONSTANT STRUCTURE MATRICES

X = NONZERO ELEMENT

O = ZERO ELEMENT



ELEMENTS IN ORIGINAL STIFFNESS MATRIX



ELEMENTS IN DECOMPOSED STIFFNESS MATRIX

FIGURE 8 TYPICAL ELEMENT PATTERN IN A STIFFNESS MATRIX

3. STEP-BY-STEP SOLUTION

The main phase in the analysis is the step-by-step solution of the equilibrium equations, Eq. (7.1.11. Vol. 1). The algorithm used was presented in Table 7.1. The aim in this chapter is to describe in more detail the actual computer solution. Since the program can perform static and dynamic, linear and nonlinear analysis, it is convenient to consider in the following the different analysis types separately.

3.1 Linear Static Analysis

In a linear static analysis, all element groups are linear and only the linear stiffness matrix is calculated in the matrix assemblage phase. The stiffness matrix is triangularized before entering the step-by-step solution phase. It should be noted that this solution corresponds to a linear dynamic analysis, in which mass and damping effects are neglected. Therefore, by specifying time varying loads, the solution can be obtained for multiple load conditions.

Figure 9 shows the tape operations used in the analysis.

3.2 Linear Dynamic Analysis

In a linear dynamic analysis all elements are linear, with mass and possibly damping effects included. The mass matrix may be diagonal (lumped mass analysis) or banded (consistent mass analysis) and additional concentrated masses may be specified at selected degrees of freedom. The damping matrix C is assumed to be of the form

$$C = \alpha M + \beta K + C_d \qquad (3.1)$$

where α and β are the Rayleigh damping coefficients and C_d is diagonal damping matrix, assembled from concentrated dampers which are specified at selected degrees of freedom (Ref. 19, Vol. 1). In Eq. (3.1), K and M are the linear stiffness and mass matrices of the complete element assemblage.

The tape operations performed during a linear dynamic analysis depend on the characteristics of the mass and damping matrices employed. Figure 10 illustrates the various possibilities for storage and retrieval of the matrices used.

3.3 Nonlinear Static Analysis

In nonlinear static analysis linear and nonlinear element groups are defined. Damping and mass effects are neglected.

Before the step-by-step solution, the linear stiffness matrix corresponding to the linear elements of the complete element assemblage was calculated (see Fig. 6). This matrix is now updated in preselected load steps by the stiffness matrices of the nonlinear elements to form the current tangent stiffness matrix. The interval of load steps in which a new tangent stiffness matrix is to be formed is input to the program.

Depending on the nonlinear formulations and the nonlinear material models used, and also depending on the magnitude of the load steps, the accuracy of the solution may be significantly increased using equilibrium iteration. In the program the interval of load steps, in which equilibrium iterations are to be performed, can be defined in the input control data.

The storage of the matrices and tape operations carried out in the analysis are shown in Fig. 11.

3.4 Nonlinear Dynamic Analysis

A nonlinear dynamic analysis is carried out essentially in the same way as a nonlinear static analysis, but mass and possibly damping effects are included. The mass and damping matrices are defined as in linear dynamic analysis, where the Rayleigh damping coefficient β is now applied to the linear stiffness matrix of the element assemblage. It should be noted that the structure mass and damping matrices are calculated before the step-by-step solution, see Fig. 6.

The tape storage scheme and program flow in a nonlinear dynamic analysis are given in Fig. 12.

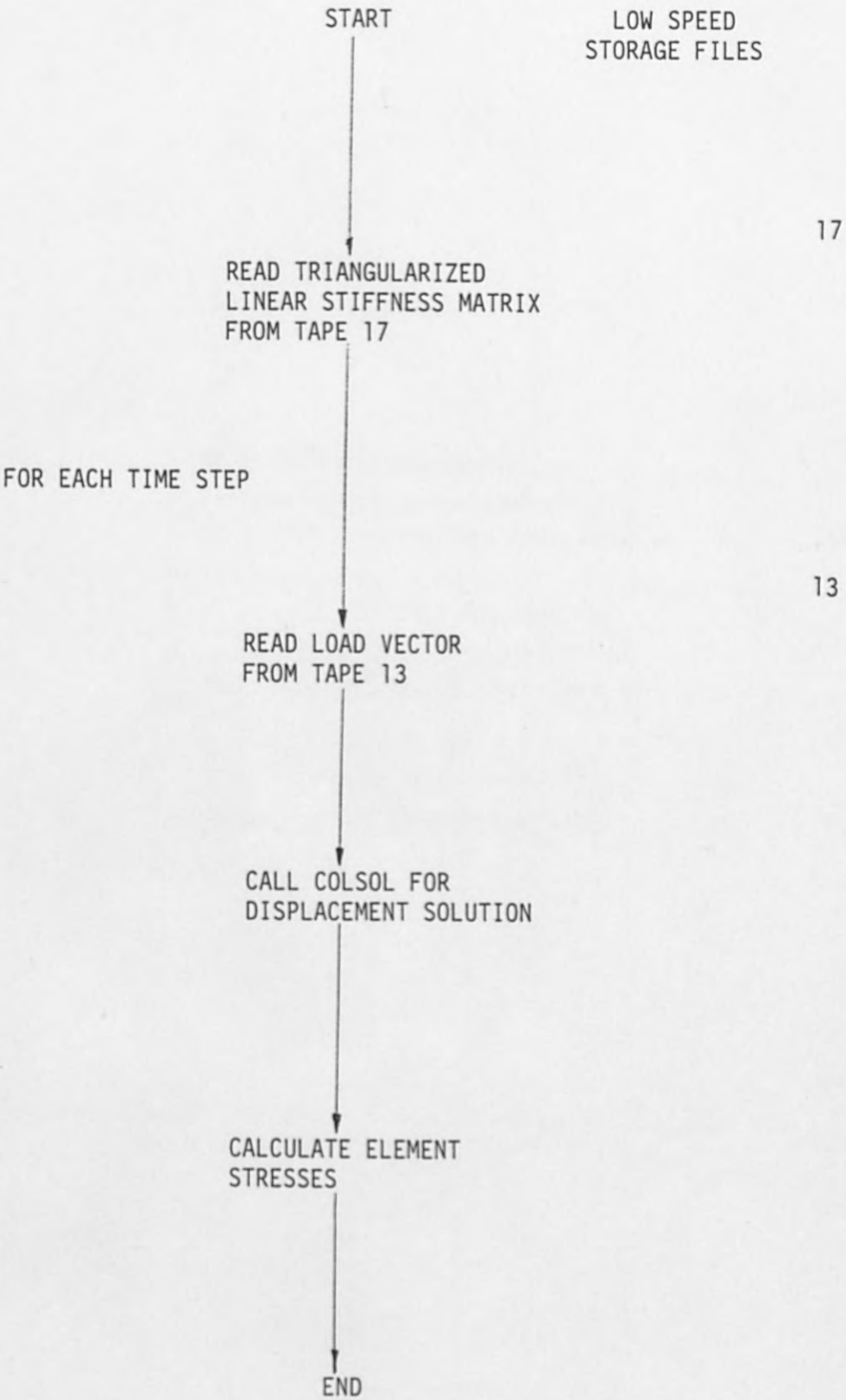


FIGURE 9 FLOW CHART FOR STEP-BY-STEP SOLUTION IN LINEAR STATIC ANALYSIS

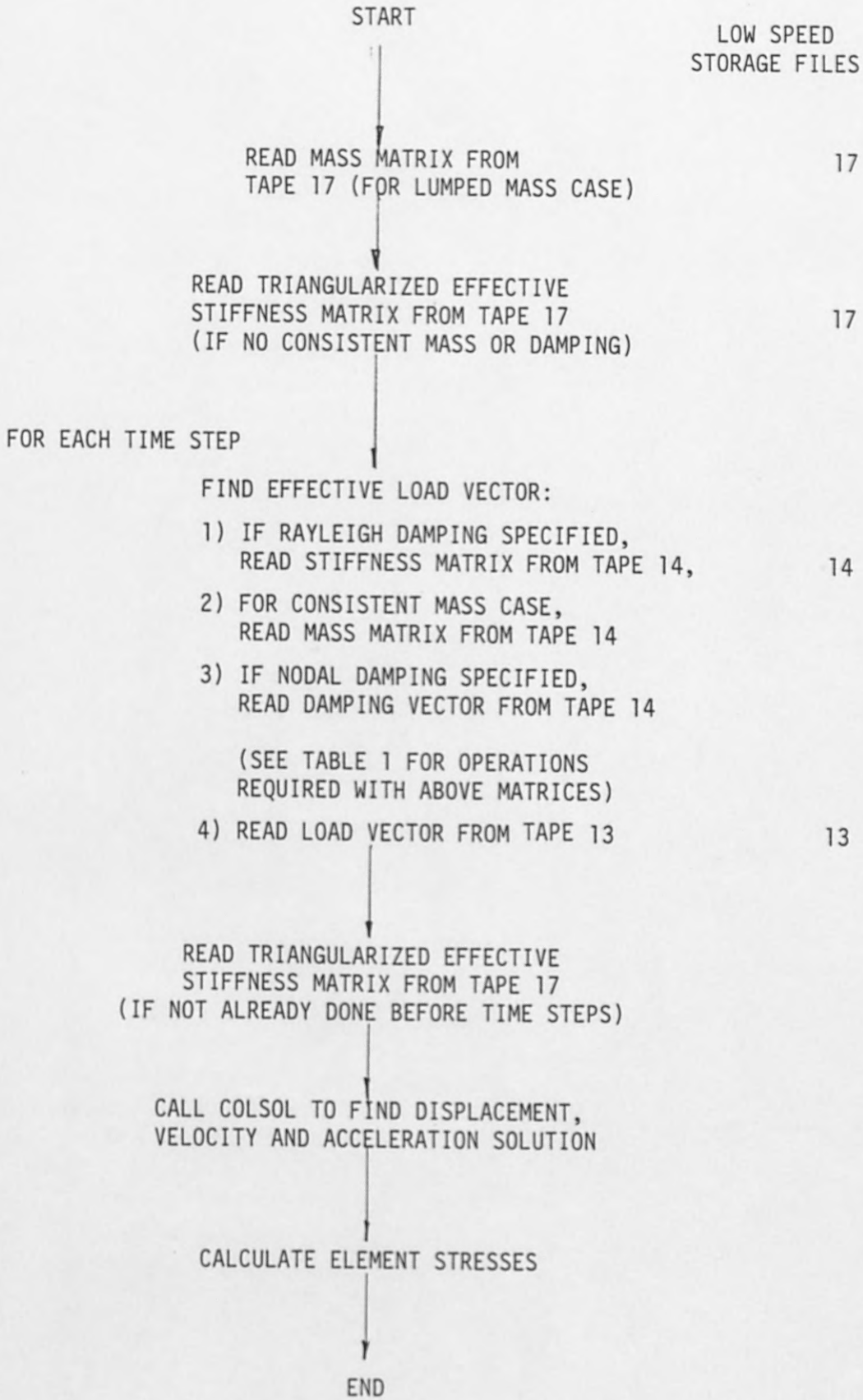


FIGURE 10 FLOW CHART FOR STEP-BY-STEP SOLUTION IN LINEAR DYNAMIC ANALYSIS

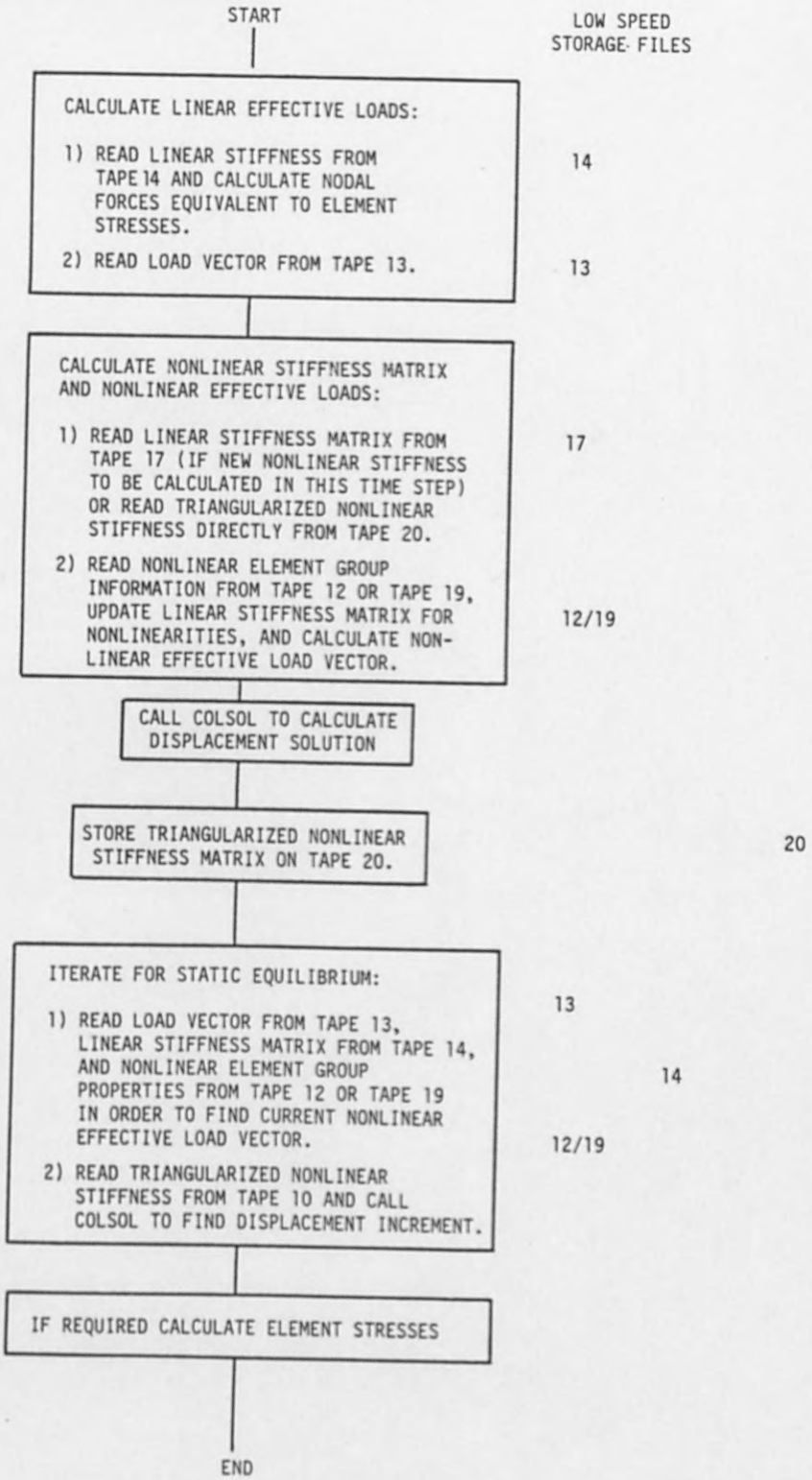


FIGURE 11 FLOW CHART FOR STEP-BY-STEP SOLUTION IN NONLINEAR STATIC ANALYSIS

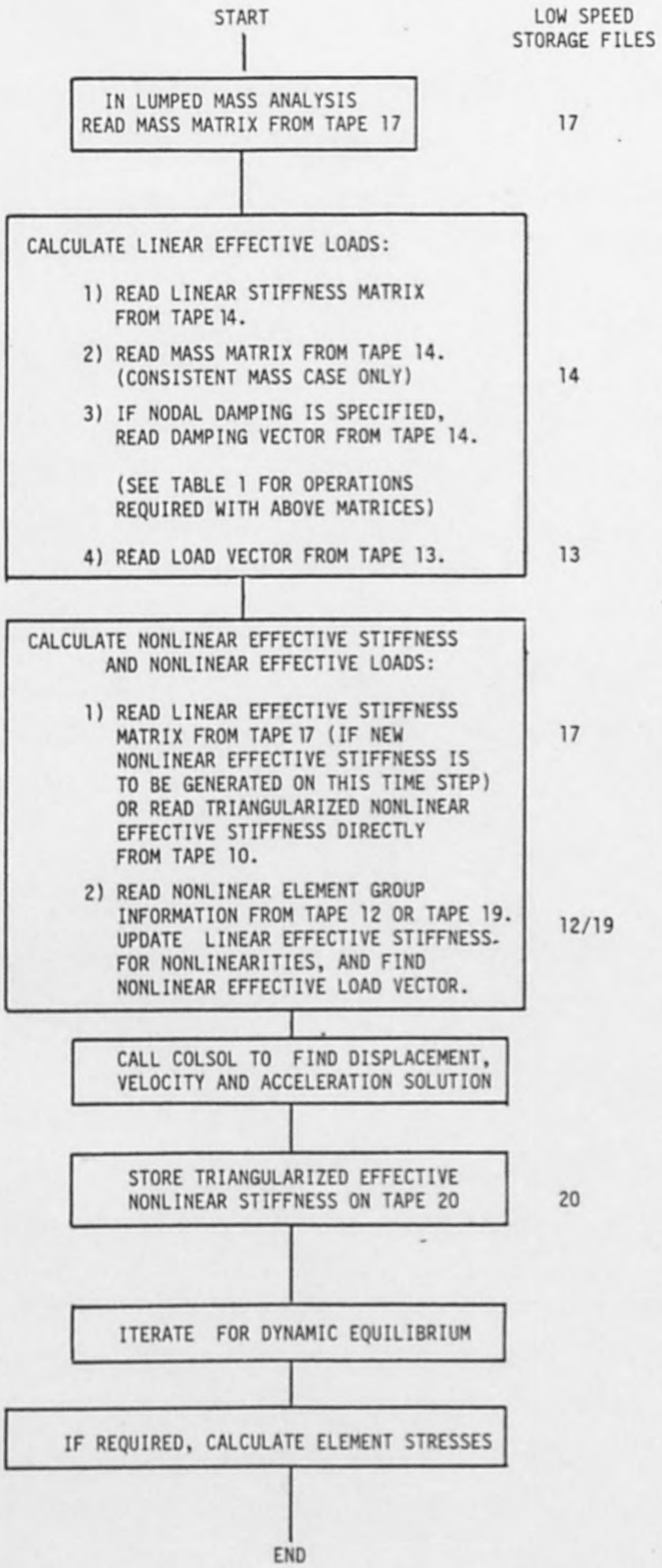


FIGURE 12 FLOW CHART FOR STEP-BY-STEP SOLUTION IN NONLINEAR DYNAMIC ANALYSIS

4. THE ELEMENT LIBRARY

In the current version of program NONSAP all finite elements are isoparametric (or subparametric) elements.
Corresponding to the nonlinearities in the system, four different analysis procedures may be considered for a finite element:

Linear Elastic Analysis

The displacements of the element are assumed to be negligibly small and the strains infinitesimal. The material is isotropic or orthotropic linear elastic.

Materially Nonlinear Only Analysis

The displacements of the element are negligibly small, and the strains are infinitesimal. The material stress-strain description is nonlinear.

Total Lagrangian Formulation

The element may experience large displacements and large strains. The material stress-strain relationship is linear or nonlinear.

Updated Lagrangian Formulation

The element may experience large displacements and large strains. The material stress-strain description is linear or nonlinear.

The linear elastic analysis does not allow for any nonlinearities, whereas the materially nonlinear only analysis includes material nonlinearities, but no geometric nonlinearities(Ref. 19 Vol. 1). The different linear and nonlinear material models currently available in NONSAP are described in Section 5. The total Lagrangian and updated Lagrangian formulations may include all nonlinearities, and which formulation should be employed depends essentially on the definition of the material model used, as described in the next chapter.

In the following, the finite elements currently available in NONSAP are briefly described.

4.1 Truss Element

A three-dimensional truss element is available in NONSAP. The element is assumed to have constant area, and may be used in linear elastic analysis, materially nonlinear and/or large displacement geometrically nonlinear analysis. In the large displacement analysis, the updated Lagrangian formulation is used, but small strains are assumed in the calculation of element stresses.

The nonlinear elastic model is described in Section 5. The truss element can be used to specify nonzero boundary displacements.

4.2 Plane Stress and Plane Strain Element

A variable-number-node isoparametric finite element is available for two-dimensional plane stress and plane strain analysis. The element may have from 3 to 8 nodes, where any one of the nodes 5 to 8 can be omitted. The variable-number-node option allows effective modelling from coarse to finer finite element meshes.

The plane stress and plane strain element can be used in all four formulations. The material models available are described in the next chapter.

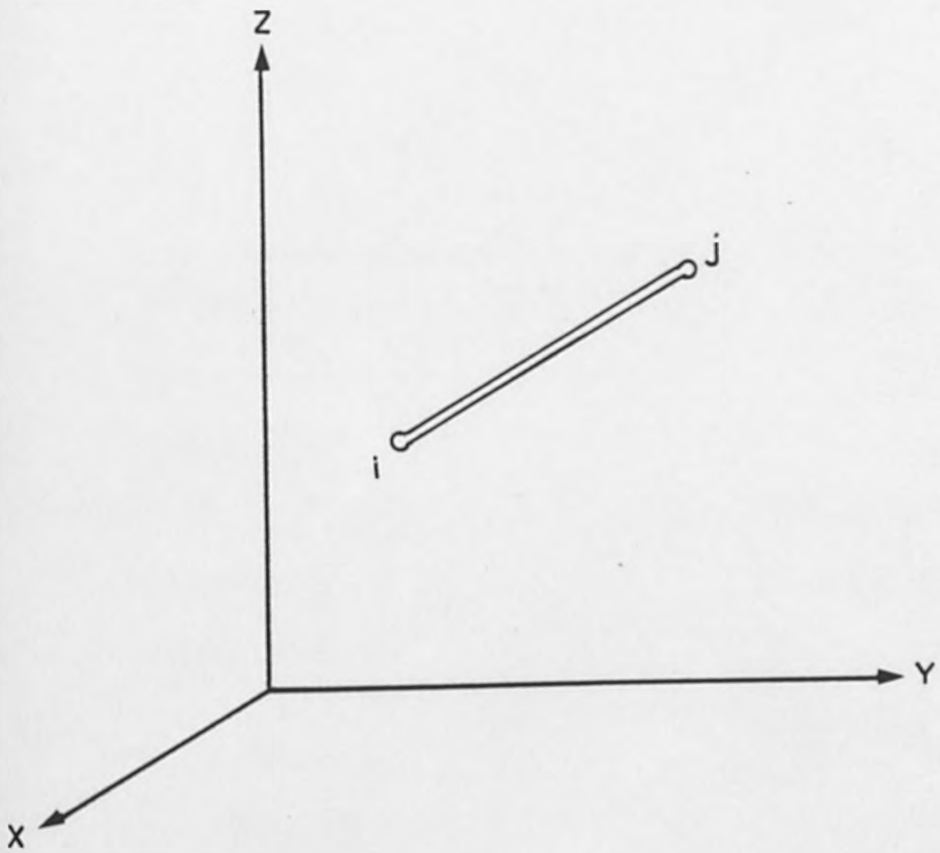
4.3 Axisymmetric Shell or Solid Element

The variable-number-node element described above is also available for axisymmetric two-dimensional analysis of shells or solids (with axisymmetric loading).

4.4 Three-Dimensional Solid or Thick Shell Element

A general three-dimensional isoparametric element with a variable number of nodes from 8 to 21 can be used. The first 8 nodes are the corner nodes of the element, nodes 9 to 20 correspond to midside nodes and node 21 is a center node. The element can be used for three-dimensional analysis of solids and thick shells. As for the two-dimensional elements, the possibility of choosing different element node configurations allows effective finite element modelling.

The three-dimensional element can currently only be used in linear isotropic analysis and in nonlinear analysis with material nonlinearities only.



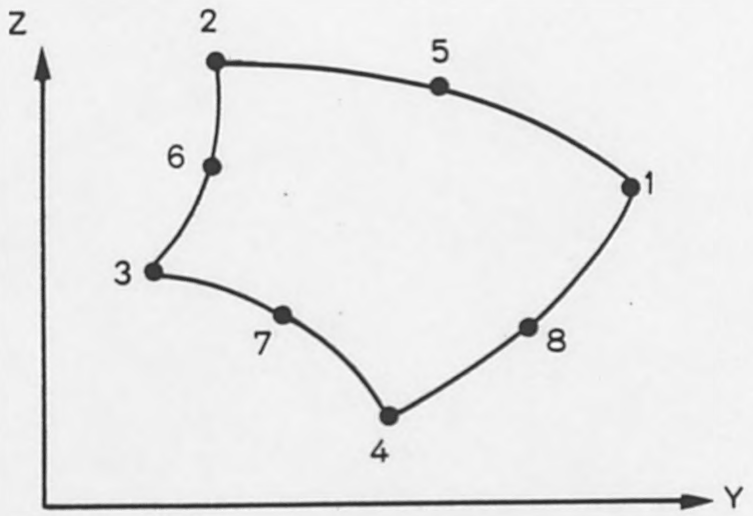
AVAILABLE
NONLINEAR FORMULATIONS

1. LINEAR ANALYSIS
2. MATERIALLY NONLINEAR ONLY
3. UPDATED LAGRANGIAN WITH LARGE DISPLACEMENTS BUT SMALL STRAINS

AVAILABLE
MATERIAL MODELS

1. LINEAR ELASTIC
2. NONLINEAR ELASTIC

FIGURE I TRUSS ELEMENT



AVAILABLE

NONLINEAR FORMULATIONS

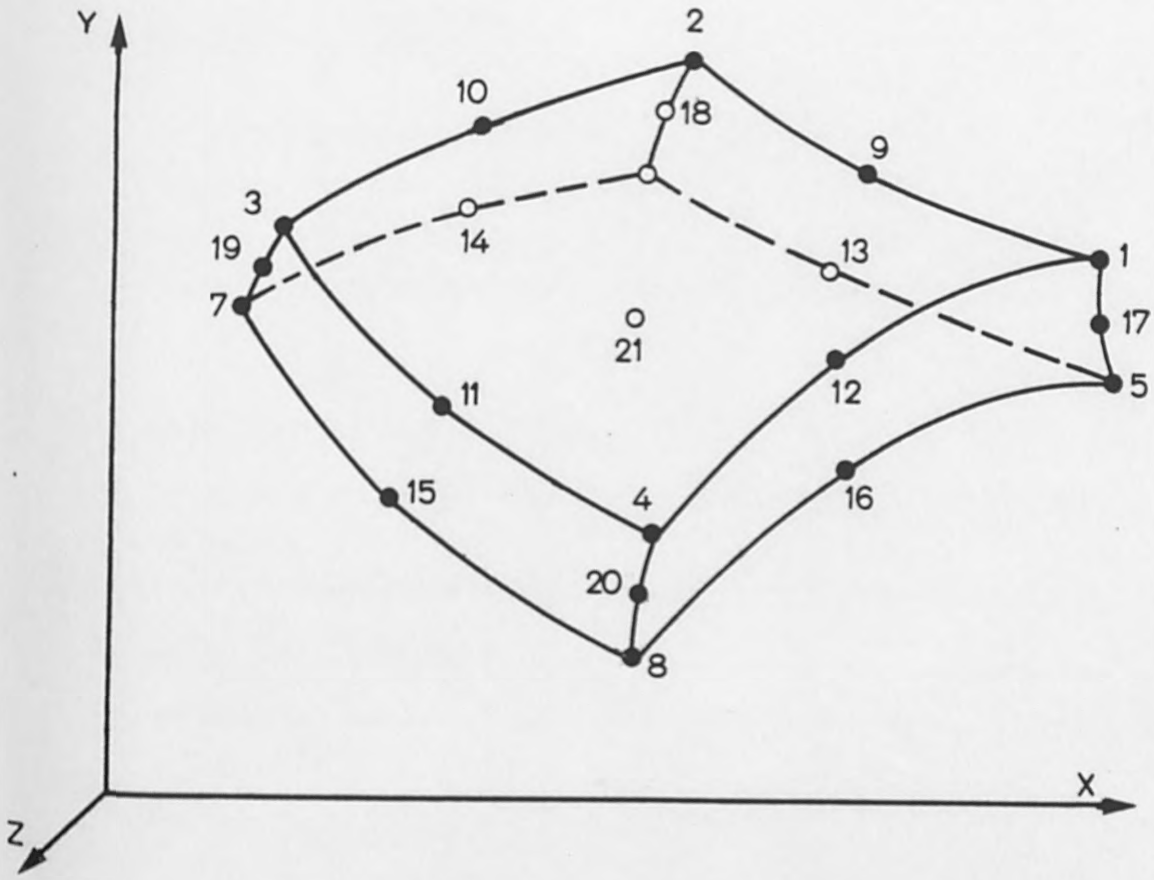
1. LINEAR ANALYSIS
2. MATERIALLY NONLINEAR ONLY
3. UPDATED LAGRANGIAN
4. TOTAL LAGRANGIAN

AVAILABLE

MATERIAL MODELS

1. LINEAR ISOTROPIC ELASTIC
2. LINEAR ORTHOTROPIC ELASTIC
3. VARIABLE TANGENT MODULI MODEL
4. CURVE DESCRIPTION NONLINEAR MODEL (WITH OR WITHOUT TENSION CUT-OFF ASSUMPTION)
5. PLASTICITY MODELS (VON MISES OR DRUCKER-PRAGER YIELD CONDITION)
6. NONLINEAR, ISOTROPIC INCOMPRESSIBLE ELASTIC (MOONEY-RIVLIN MATERIAL)

FIGURE 2 TWO-DIMENSIONAL PLANE STRESS, PLANE STRAIN AND AXISYMMETRIC ELEMENTS



AVAILABLE

NONLINEAR FORMULATIONS

1. LINEAR ANALYSIS
2. MATERIALLY NONLINEAR ONLY

AVAILABLE

MATERIAL MODELS

1. LINEAR ISOTROPIC ELASTIC
2. CURVE DESCRIPTION NONLINEAR MODEL

FIGURE 3 THREE-DIMENSIONAL SOLID AND THICK SHELL ELEMENT

5. THE MATERIAL MODELS

The largest number of material models is available for two-dimensional analysis, since it is anticipated that the two-dimensional elements will be used in most analyses. For the same reason, also all three nonlinear formulations can be used for the two-dimensional elements.

5.1 Truss-Element Material Models

The truss element material behavior can be described by means of two models.

Linear Elastic Material

The material can be linear elastic defined ba Young's modulus only.

Nonlinear Elastic Material

The nonlinear elastic material behavior is defined by specifying the stress as a piece-wise linear function of the current (infinitesimal) strain. Thus, the total stress and the tangent modulus are directly defined in terms of the total strain.

5.2 Two-Dimensional Element Material Models

The stress-strain relationship of the two dimensional elements can be described by various linear and nonlinear material models. In the definition of a material model, it may have been assumed that a specific nonlinear formulation is used. The application of the different material models is discussed in (Ref. 18 Vol. 1).

Isotropic and Orthotropic Linear Elastic Material

The stress-strain relationships are defined by means of the constant Young's moduli and Poisson's ratios.

Mooney-Rivlin Material Model

A hyperelastic incompressible material model is available for the analysis of rubber-like materials. The stress-strain relationship is defined using the Mooney-Rivlin material constants. In NONSAP the model can only be used in plane stress analysis.

Elastic-Plastic Material Models

Elastic-plastic analysis using a plastic potential function can be carried out. The plasticity relations available are those based on the use of the von Mises yield condition and the Drucker-Prager yield condition. Both forms of describing material behavior have been employed extensively in practice. Using the von Mises criterion, linear isotropic hardening can be assumed. In analysis using the Drucker-Prager yield condition, the material is assumed to be elastic perfectly plastic.

The Curve Description Model

The curve description model is used in essentially the same way as the variable tangent moduli model. The model also describes the response of geological materials. In the model, the instantaneous bulk and shear moduli are defined by piecewise linear functions of the current volume strain. An explicit yield condition is not used, and whether the material is loading or unloading is defined by the history of the volume strain only.

In the analysis of some problems, tensile stresses due to applied loading cannot exceed the gravity in-situ pressure. In such conditions the model can be used to simulate tension cut-off, i.e., the material model assumes reduced stiffness in the direction of the tensile stresses which exceed in magnitude the gravity pressure.

5.3 Three-Dimensional Element Material Models

In principle, most two-dimensional models would also be applicable in three-dimensional analysis. However, in the current version of NONSAP, only the isotropic linear elastic model and the curve description model (without tension cut-off capability) are available.

II. MASTER CONTROL CARDS

Card 1 (I5, 6I1, I4, 3I5, 2F10.0, I5)

notes	columns	variable	entry
(1)	1 - 5	NUMNP	Total number of nodal points; EQ.0; program stop
(2)	6	IDOF(1)	Master X-translation code; EQ.0; admissible EQ.1; deleted
	7	IDOF(2)	Master Y-translation code
	8	IDOF(3)	Master Z-translation code
	9	IDOF(4)	Master X-rotation code
	10	IDOF(5)	Master Y-rotation code
	11	IDOF(6)	Master Z-rotation code
(3)	12 - 15	NEGL	Number of linear element groups; EQ.0; all elements are nonlinear
(4)	16 - 20	NEGNL	Number of nonlinear element groups; EQ.0; all elements are linear
(5)	21 - 25	MODEX	Flag indicating solution mode; EQ.0; data check only EQ.1; execution EQ.2; restart
(6)	26 - 30	NSTE	Number of solution time steps
(6)	31 - 40	DT	Time step increment
(7)	41 - 50	TSTART	Time at solution start
(8)	51 - 55	IPRI	Output printing interval; EQ.0; default set to "1"

NOTES/

(1) The total number of nodes (NUMNP) controls the amount of data to be read in Section III. If NUMNP.EQ.0, the program terminates execution.

(2) The codes IDOF(1), IDOF(2), ..., IDOF(6) given in cc 6-11 can have values of "0" or "1" and are used to conserve storage and reduce input for special problems not requiring a full six degrees of freedom at each node. If IDOF(I).EQ.1, the I-th (I=1,2,...,6) nodal displacement does not exist in this problem, and any reference in the data to the I-th node degree of freedom is ignored by the program. If, for example, this problem concerns the analysis of a 2/D continuum, the convention is that 2/D elements must be described in the global Y,Z plane and that only the global Y and Z translations are defined for the 2/D continuum elements; hence, IDOF(I) for I of 1,4,5 and 6 must be "1".

Table II.1 contains the IDOF sets for the current element types operational in the program. This table was prepared assuming a general orientation of the element type in X,Y,Z space.

(3) The program distinguishes between linear and nonlinear elements. Linear elements have their stiffness matrices formed only once, and the formulation excludes consideration of either geometric or material nonlinearities; linear elements are used to represent those regions of a model which can be assumed to behave linearly and should be used (where possible) to improve solution efficiency.

An element group is a series of elements of a particular type (e.g., TRUSS, 2/D CONTINUUM, etc.) in which element numbers are assigned in ascending sequence beginning with "1" and ending with the total number of elements in that particular group. The organization of elements into groups is somewhat arbitrary and is best illustrated by the following example.

ELEMENT TYPE	MASTER BOUNDARY CONDITION CODES					
	IDOF(1)	IDOF(2)	IDOF(3)	IDOF(4)	IDOF(5)	IDOF(6)
1. TRUSS	0	0	0	1	1	1
2. 2/D CONTINUUM	1	0	0	1	1	1
3. 3/D CONTINUUM	0	0	0	1	1	1

TABLE II.1 - Possible Finite Element Degrees of Freedom for the Various Element Types Currently Operational in NONSAP

II. MASTER CONTROL CARDS (continued)

Suppose that a model contains 20 linear TRUSS elements and 33 linear 2/D CONTINUUM elements, 11 of which have isotropic material properties. Assuming that the 2/D isotropic properties are not given as equivalent orthotropic properties, a minimum of three (3) linear element groups (i.e., NEGL.EQ.3) is required in this example--

LINEAR GROUP NUMBER	ELEMENT TYPE	RANGE OF ELEMENT NUMBERS	TYPE OF MATERIAL MODEL
1	TRUSS	1 - 20	AE, constant
2	2/D CONTINUUM	1 - 11	Isotropic
3	2/D CONTINUUM	1 - 22	Orthotropic

Further, suppose that it is desirable to separate the 20 TRUSS elements into two groups of 5 and 15 elements and the 22 2/D CONTINUUM orthotropic elements into two groups, one with 8 and the other with 14 elements; then, NEGL.EQ.5 and

1	TRUSS	1 - 5	AE, constant
2	TRUSS	1 - 15	AE, constant
3	2/D CONTINUUM	1 - 11	Isotropic
4	2/D CONTINUUM	1 - 14	Orthotropic
5	2/D CONTINUUM	1 - 8	Orthotropic

Since NONSAP is an overlaid program, the order of the element groups (linear and nonlinear) is most efficient when all TRUSS element groups, 2/D CONTINUUM element groups,... are grouped together. In addition, all element groups using the same nonlinear material model are best input together.

It need be noted that the program must read data for all of the "NEGL" groups before reading any data for the nonlinear groups.

The one restriction that must be noted when organizing elements into groups (either linear or nonlinear) is that only one type of material model is allowed for the elements in that group. Also, it is permissible to model a structure with nonlinear elements only, in which case NEGL.EQ.0 and NEGNL.GE.1.

(4) Nonlinear elements include the effects of material and/or geometric nonlinearities in the formation of stiffness and stress recovery matrices. The type of nonlinearities to be associated with an individual group of elements is defined by means of data given on the element group control card. In separating elements into nonlinear groups, note that only one type of material model is allowed for the group. The order in which groups are input was discussed in note (3) above.

The total number of element groups processed by the program is the sum NEGL+NEGNL . Also, NEGNL can be zero, but then the number of linear element groups must be at least one.

(5) The MODEX parameter determines whether the program is to check the data without executing an analysis (i.e., MODEX.EQ.0) or if the program is to solve the problem. In the data check only mode, the program only reads and prints all data. Therefore, the MODEX.EQ.0 option should be used for data checking. If MODEX.EQ.2, the problem is a restart job. Refer to Appendix A for setting up a restart job.

(6) DT is the solution time step, Δt, and is used similarly for static, quasi-static or dynamic problems. DT for static problems is not used to perform step-by-step time integration for system response, but rather is used as an equivalent "load step" or "loading increment". All forcing functions are input as tables of f(t) versus t, and the loads at solution step "n" (where "n" is between "1" and "NSTE") are found in the f(t) tables by linear interpolation at $t=n\Delta t$. NSTE must be at least one, except in the special case when frequencies and mode shapes are to be calculated but no time integration solution is desired.

(7) The time at solution start (TSTART) is an input convenience for restart (i.e., MODEX.EQ.2) jobs. TSTART in a restart job would be the final time to which a previous solution was run and saved for use in supplying initial conditions for this job. Since the time counter is incremented from TSTART, none of the forcing function tables need be revised (i.e., shifted in time) for the restart job.

(8) The print interval determines at which solution step interval program results are to be printed. If IPRI.EQ.4, output is produced at the end of solution steps 4, 8, 12, etc.

If IPRI is larger than the total number of solution steps (NSTE), then no output will be printed during the course of solution. If IPRI.LE.NSTE, then print directives must be given for displacements, velocities, and acceleration on Card 6 in this section and for the stress components on the element cards (Section IX.4, X.4, or XI.4).

II. MASTER CONTROL CARDS (continued)

Card 2 (4I5)

notes	columns	variable	entry
(1)	1 - 5	IMASS	Control flag indicating static or dynamic analysis; EQ.0; static analysis GT.0; dynamic analysis Also flag indicating mass matrix type; EQ.0; no mass effects EQ.1; lumped (diagonal) mass EQ.2; consistent mass matrix
(2)	6 - 10	IDAMP	Flag indicating damping type; EQ.0; no damping effects EQ.1; Rayleigh damping
(3)	11 - 15	IMASSN	Number of concentrated nodal masses
(3)	16 - 20	IDAMPN	Number of concentrated nodal dampers

NOTES/

(1) The control flag on static or dynamic analysis (IMASS) determines whether or not the program is to solve a problem including the contribution of inertia forces to system equilibrium, i.e. whether to solve a static or dynamic problem.

If IMASS.EQ.0, the program will solve a static (or quasi-static) problem, and no storage will be allocated for either the system mass matrix or the system velocity and acceleration vectors.

If IMASS.EQ.1 or 2, the analysis will be a dynamic analysis. For IMASS.EQ.1 a lumped (diagonal) mass matrix is allowed, in which case the diagonal mass coefficients are stored as a vector, and for IMASS.EQ.2 a consistent mass matrix is generated (from element data). It should be noted that the computation of the effective load vector at each solution time step is considerably more expensive when the consistent mass option is requested, and the extra effort involved in a consistent (as opposed to the diagonal) mass analysis may not be justified in many (if not most) problems.

The mass matrix is constant and is only formed once before the time integration is started.

NOTES/

(2) The damping matrix flag (IDAMP) determines whether or not the program is to include in a dynamic analysis Rayleigh damping. The Rayleigh damping coefficients are defined in Section V. Rayleigh damping can only be included when a dynamic analysis is specified (IMASS.GT.0).

(3) In a dynamic analysis, i.e. IMASS.GT.0, additional concentrated masses and/or additional concentrated dampers can be specified at selected degrees of freedom. The concentrated masses and concentrated dampers are input in Sections VI and VII.

A dynamic analysis with concentrated masses only (i.e., no lumped mass or consistent mass effects) can be accomplished by specifying IMASSN.GE.1, IMASS.EQ.1, and setting the mass density on all element cards to 0.0.

Card 3 (I5)

note	columns	variable	entry
(1)	1 - 5	IEIG	Flag indicating frequency solution mode; EQ.0; no frequency solution EQ.1; find lowest frequencies and associated mode shapes

NOTES/

(1) The control parameter IEIG determines if the program is to solve for vibration mode shapes and frequencies before solving for dynamic response of the model. If IEIG.EQ.1, then additional information which controls the eigensolution must be supplied in Section XII.

If frequencies and mode shapes only are desired (i.e. no step-by-step solution) set NSTE.EQ.0 on card 1 of this section.

II. MASTER CONTROL CARDS (continued)

Card 4 (4I5,E10.4)

note	columns	variable	entry
(1)	1 - 5	ISREF	Number of time steps between reforming effective stiffness matrix EQ.0; default set to "1"
(2)	6 - 10	NUMREF	Number of allowable stiffness reformations in each time step
(3)	11 - 15	IEQUIT	Number of time steps between equilibrium iterations EQ.0; default set to "1"
(3)	16 - 20	ITEMAX	Maximum number of equilibrium iterations permitted EQ.0; default set to "15"
(3)	21 - 30	RTOL	Relative tolerance used to measure equilibrium convergence EQ.0; default set to "1.E-3"

NOTES/

(1) The stiffness matrix reformation interval (ISREF) is ignored if the model is composed of linear element groups only; i.e., NEGNL.EQ.0. For linear problems the matrix is formed and decomposed only once.

For models containing nonlinear element groups, the system stiffness matrix is reformed every ISREF solution steps. For example, if ISREF.EQ.3, the stiffness matrix is formed in solution steps 1 (always), 3, 6, etc. based on conditions known at the end of steps 0, 2, 5, etc., respectively. Only the nonlinear portion of the complete system stiffness matrix is reformed; the linear portion is saved and reinstated when the complete matrix is calculated.

(2) NUMREF is currently ignored.

NOTES/

(3) If a structure is represented by nonlinear element groups with material models which allow for "equilibrium iteration", then the parameter IEQUIT determines at what solution step interval the program is to iterate for system equilibrium. ITEMAX is the maximum number of cycles of iteration allowed in the solution step and RTOL is used to measure convergence of the iteration in terms of change in system displacements. For example, if IEQUIT.EQ.5, ITEMAX.EQ.12 and RTOL.EQ.0.002, then providing the material model(s) allow for iteration, up to 12 cycles of iteration will be performed at solution steps 5,10,15, etc. with convergence declared if

$$\left| \frac{||u^{(n-1)}|| - ||u^{(n)}||}{||u^{(n)}||} \right| \leq 0.002$$

where $||u^{(n)}||$ is the Euclidean norm of the system displacement vector at cycle "n" of the iteration.

Whether or not a material model specifically allows iteration for equilibrium depends on how the model was incorporated into the program; the sections on the element material models distinguish between models that do and do not allow iteration.

Equilibrium iteration can only be performed if

(a) the structure contains at least one nonlinear element group, and

(b) the material models used to represent all nonlinear element groups allow for the possibility of equilibrium iteration.

The parameters IEQUIT, ITEMAX and RTOL will not be used if

(a) the structure is represented with linear elements only, or

(b) any one of the material models associated with a nonlinear element group does not allow for equilibrium iteration.

II. MASTER CONTROL CARDS (continued)

Card 5 (I10, 2F10.0)

note	columns	variable	entry
(1)	1 - 10	IOPE	Time integration method used EQ.0; default set to "1" EQ.1; Wilson's theta method EQ.2; Newmark's method
(2)	11 - 20	OPVAR(1)	First Integration Parameter 1) If IOPE.EQ.1 parameter is THETA EQ.0; default set to 1.4 2) If IOPE.EQ.2 parameter is DELTA EQ.0; default set to 0.5
(2)	21 - 30	OPVAR(2)	Second Integration Parameter 1) If IOPE.EQ.1 parameter is not applicable 2) If IOPE.EQ.2 parameter is ALPHA EQ.0; default as specified in note below

NOTES/

(1) For static problems (IMASS.EQ.0) this card is read, but the information is not used. For dynamic analysis IOPE determines which time integration algorithm will be used in the step-by-step solution.

(2) OPVAR(1) and OPVAR(2) store the integration parameters associated with the time integration method specified by IOPE.

a) Wilson's θ-method uses one parameter (THETA), which is usually specified as 1.4.

b) Newmark's method uses two parameters. DELTA is specified by the user (or the default value of 0.5 is used) and ALPHA is either set to a default value by the formula

ALPHA = 0.25*(DELTA + 0.50)**2,

or specified directly by the user.

II. MASTER CONTROL CARDS (continued)

Card 6 (4I5)

note	columns	variable	entry
(1)	1 - 5	NPB	Number of blocks of displacement/ velocity/acceleration printout EQ.0, print all nodal point components
(2)	6 - 10	IDC	Displacement printout code EQ.0, no displacement printout EQ.1, print displacements
(2)	11 - 15	IVC	Velocity printout code EQ.0, no velocity printout EQ.1, print velocities
(2)	16 - 20	IAC	Acceleration printout code EQ.0, no acceleration printout EQ.1, print accelerations

NOTES/

(1) For large meshes it is usually not necessary to print displacements, velocities, and accelerations at every node. Hence, nodes for which printout is desired are grouped into NPB printout blocks. Each block of nodes is defined by the node numbers of the first and last node in the block (see next card).

If NPB.EQ.0 all nodal quantities are printed regardless of the values of IDC, IVC, and IAC.

(2) The displacement solution at the nodes within the blocks is printed if IDC.EQ.1. In dynamic problems the velocity and/or acceleration solutions also are printed if IVC.EQ.1 and/or IAC.EQ.1.

II. MASTER CONTROL CARDS (continued)

Card 7 (16I5)

note	columns	variable	entry
(1)	1 - 5	IPNODE(1,1)	First node of printout block no. 1.
	6 - 10	IPNODE(2,1)	Last node of printout block no. 1.
	11 - 15	IPNODE(1,2)	First node of printout block no. 2.
		etc.	

NOTES/

(1) Two entries are expected for each printout block, namely, the first node of the block and the last node of the block. All nodal points between these two nodes will be included in the printout block.

If NPB.EQ.0 leave this card blank.

III. NODAL POINT DATA (A1,I4,A1,I4,5I5,3F10.0,I5)

note	columns	variable	entry
(1)	1	CT	Symbol describing the coordinate system for this node; EQ. ; (blank) cartesian (X,Y,Z) EQ.X; X-cylindrical
(2)	2 - 5	N	Node (joint) number; GE.1 and LE.NUMNP
(3)	6	PSF	Print suppression flag (ignored unless N.EQ.1); EQ. ; (blank) no suppression EQ.A; suppress ordered list of node coordinates EQ.B; suppress list of equation numbers EQ.C; both A and B, above
(4)	7 - 10	ID(1,N)	X-translation boundary code
	11 - 15	ID(2,N)	Y-translation boundary code
	16 - 20	ID(3,N)	Z-translation boundary code
	21 - 25	ID(4,N)	X-rotation boundary code
	26 - 30	ID(5,N)	Y-rotation boundary code
	31 - 35	ID(6,N)	Z-rotation boundary code
(5)	36 - 45	X(N)	X (or z)-coordinate
	46 - 55	Y(N)	Y (or r)-coordinate
	56 - 65	Z(N)	Z (or θ)-coordinate (degrees)
(6)	66 - 70	KN	Node number increment for node data generation; EQ.0; no generation

NOTES/

(1) Special cylindrical coordinate systems are allowed for the specification of node coordinates. If an "X" is entered in card column one (1), then the entries in cc 36-65 are with respect to a cylindrical (r,θ,z) system rather than the standard cartesian (X,Y,Z) system. The generator (i.e., the z-axis) of the "X-cylindrical" system coincides with the global X-axis, and θ is zero (0) in the global X-Y plane--

SYSTEM	z-axis	θ-zero plane	TRANSFORMATIONS
X	X	X-Y	$X = z$ $Y = r\cos\theta$ $Z = r\sin\theta$

Cylindrical input is merely a user convenience for locating nodes in the (X,Y,Z) system, and no other references to the cylindrical system are implied; i.e., boundary conditions, output displacements, etc. are still referenced to the (X,Y,Z) system.

(2) Nodal data must be defined for all (NUMNP) nodes. Node data may be input directly (i.e., each node on its own individual card) or the generation option may be used if applicable (see note 6, below). Admissible node numbers range from "1" to the total number of nodes (NUMNP). Node numbers may not be repeated or omitted.

(3) The print suppression flag (PSF) is used to eliminate the second printing of ordered node coordinates or to suppress printing of equation number assignments (or both). The PSF character is entered on the card for node one (1) only.

(4) Boundary condition codes can only be assigned the following values (M = 1,2,...,6)--

ID(M,N) = 0; unspecified (free) displacement

ID(M,N) = 1; deleted (fixed) displacement

NOTES/(4) (continued)

An unspecified [ID(M,N)=0] degree of freedom is free to translate or rotate as the solution dictates. Concentrated forces (or moments) may be applied in this degree of freedom.

One system equilibrium equation is required for each unspecified degree of freedom in the model. The maximum number of equilibrium equations is always less than six (6) times the total number of nodes in the system.

Deleted [ID(M,N)=1] degrees of freedom are removed from the final set of equilibrium equations. Deleted degrees of freedom are fixed (points of external reaction), and any loads applied in these degrees of freedom are ignored by the program. Nodes that are used for geometric reference only (i.e., nodes not assigned to elements) must have all six (6) degrees of freedom deleted. Nodal degrees of freedom having undefined stiffness (such as rotations in an all TRUSS model, out-of-plane components in a two-dimensional planar model, etc.) must be deleted.

Independent of the actual entries posted for the ID(M,N) in cc 7-35 of the Nodal Data, any master boundary condition deletions (i.e., IDOF(M).EQ.1, M=1,2,...,6) which are given in cc 6-11 of Card 1 in Section II will be used for all nodes. Suppose that all rotations X,Y,Z have been deleted by means of the master codes (i.e., IDOF(M).EQ.1, M=4,5,6), then ID(M,N) (M=4,5,6) will be set to "1", and data in cc 21-35 is ignored by the program.

Two-dimensional elements must use the Y-Z plane, i.e. plane strain, plane stress, and axisymmetric elements can only be defined in the Y-Z plane, and nodal degrees of freedom having undefined stiffness must be deleted.

(5) For the case CT (cc 1) equal to the character "X", the data input in cc 36-65 are interpreted as the cylindrical (r,θ,z) coordinates of node "N". The table in note (1) contains the formulae used by the program to compute the cartesian (X,Y,Z) coordinates of node N from the cylindrical coordinate values given in cc 36-65. The origin of cartesian and cylindrical systems is the same point. Cylindrical coordinate data (if input on the card) are printed as read in the first list of node coordinates. The second listing of coordinates (if not suppressed with a PSF of "A" or "C") is an ordered printing of all nodes with coordinates converted to the (X,Y,Z) system. Note that boundary condition codes always reference the (X,Y,Z) system even if the node happens to be located using the cylindrical coordinate option.

III. NODAL POINT DATA (continued)

NOTES/ (continued)

(6) Node cards need not be input in node order sequence; eventually, however, all nodes in the set [1,NUMNP] must be defined. Node data for a series of nodes

$$[N_1,\ N_1+1*KN_1,\ N_1+2*KN_1,\ \ldots\ ,N_2]$$

may be generated from information given on two (2) cards in sequence--

CARD 1 -- $CT_1,N_1,ID(N_1,1),\ldots,ID(N_1,6),X(N_1),\ldots,KN_1$

CARD 2 -- $CT_2,N_2,ID(N_2,1),\ldots,ID(N_2,6),X(N_2),\ldots,KN_2$

KN_1 is the node generation parameter given on the first card in the sequence. The first generated node is N_1+1*KN_1; the second generated node is N_1+2*KN_1, etc. Generation continues until node number N_2-KN_1 is established. Note that the node difference N_2-N_1 must be evenly divisible by KN_1.

Coordinate generation is possible only if both nodes are given in the same system (i.e., CT_1 and CT_2 are the same). Since generation is processed using cartesian coordinates, intermediate nodes between N_1 and N_2 are located at equal intervals along the line between the two points (i.e., linear interpolation in the (X,Y,Z) system).

IV. APPLIED LOADS DATA

1. Control card (3I5)

note	columns	variable	entry
(1)	1 - 5	NLOAD	Number of cards used to prescribe loads acting at the nodes
(2)	6 - 10	NLCUR	Number of load curves (time functions)
(2)	11 - 15	NPTM	Maximum number of points used to describe any one of the load curves

NOTES/

(1) NLOAD determines the number of cards to be read in Section IV.3, below. The loads defined in Section IV.3 are concentrated node forces/moments that do not change direction as the structure deforms; i.e., the applied node forces are conservative loads.

(2) Time dependent loads are applied to the structure by means of load (or time) function [i.e., f(t)] references and function multipliers assigned with the loads. At time t the value of f(t) is found by linear interpolation in the table of f(t) vs. t; f(t) times the multiplier is the magnitude of the applied load at t. NPTM is the maximum number of [f(t),t] pairs used to describe any one of the NLCUR functions; an individual function may have fewer than NPTM [f(t),t] points as input, but no function can be input with more than NPTM points. At least two points are required per function; otherwise interpolation in time is not possible.

2. Load function data

Input NLCUR sets of the following data cards in order of increasing load function number.

a. Control data (2I5)

note	columns	variable	entry
	1 5	NTF	Time function number; GE.1 and LE.NLCUR
	6 - 10	NPTS	Number of points (i.e., f(t), t pairs) used to input this time function; GE.2 and LE.NPTM

b. [f(t),t] data (8F10.0)

note	columns	variable	entry
(1)	1 - 10	TIMV(1)	Time at point 1, t_1
	11 - 20	RV (1)	Function value at point 1, $f(t_1)$
	21 - 30	TIMV(2)	Time at point 2, t_2
	31 - 40	RV (2)	Function value at point 2, $f(t_2)$
	. . .	. . .	. . .
	71 - 80	RV (4)	Function value at point 4, $f(t_4)$
	Next card (if required)		
(2)	1 - 10	TIMV(5)	Time at point 5, t_5
	11 - 20	RV (5)	Function value at point 5, $f(t_5)$
	. . .	. . .	. . .

NOTES/

(1) Time values at successive points must increase in magnitude (i.e., TIMV(1)<TIMV(2)<TIMV(3), etc.), and TIMV(1) must be equal to zero (i.e., TIMV(1).EQ.0.0) otherwise an error condition is declared. The last time value for the function [i.e., TIMV(NPTS)] must be greater than or equal to the time at the end of solution; i.e., TIMV(NPTS) ≥ TSTART + NSTE*DT otherwise an error condition is declared.

(2) Input as many cards in this section as are required to define NPTS points, four points per card.

3. Nodal Loads Data (3I5,F10.0)

Skip this section if NLOAD.EQ.0; otherwise input NLOAD cards in this section.

note	columns	variable	entry
(1)	1 - 5	NOD	Node number to which this load is applied; GE.1 and LE.NUMNP
	6 - 10	IDIRN	Degree of freedom number for this load component; EQ.0; solution terminated EQ.1; X-translation EQ.2; Y-translation EQ.3; Z-translation EQ.4; X-rotation EQ.5; Y-rotation EQ.6; Z-rotation
	11 - 15	NCUR	Load curve number that describes the time dependence of the load; GE.1 and LE.NLCUR
	16 - 25	FAC	Function multiplier used to scale f(t) for the load at "t"; EQ.0.0; default set to "1.0"

NOTES/

(1) If the same degree of freedom (IDIRN) at the same node (NOD) is given a multiple number of times, the program combines the loads algebraically with no error diagnostic.

V. RAYLEIGH DAMPING SPECIFICATION (2F10.0)

Omit this card if IDAMP.EQ.0

note	columns	variable	entry
(1)	1 - 10	ADAMP	Rayleigh damping coefficient α
(1)	11 - 20	BDAMP	Rayleigh damping coefficient β

NOTES/

(1) Rayleigh damping is defined as $C = \alpha M + \beta K$, where α and β are input as above.

It need be noted that β is applied to the linear stiffness matrix of the element assemblage.

VI. CONCENTRATED NODAL MASSES (I10,6F10.0)

Skip this section if IMASSN.EQ.0, on card 2 of Section II. Otherwise, input IMASSN cards as follows.

note	columns	variable	entry
(1)	1 - 10	N	Node number; GE.1 and LE.NUMNP
(2)	11 - 20	XMASS(1)	X-direction mass
	21 - 30	XMASS(2)	Y-direction mass
	31 - 40	XMASS(3)	Z-direction mass
(3)	41 - 50	XMASS(4)	X-rotational mass
	51 - 60	XMASS(5)	Y-rotational mass
	61 - 70	XMASS(6)	Z-rotational mass

NOTES/

(1) Input IMASSN cards; node order is not important. Repeating nodes accumulates mass at the node.

(2) Mass components input for deleted (or non-existant) degrees of freedom are ignored by the program without a diagnostic message.

(3) Rotational degrees of freedom are currently not used (see Table II.1).

VII. CONCENTRATED NODAL DAMPERS (I10,6F10.0)

Skip this section if IDAMPN.EQ.0, on card 2 of Section II, above. Otherwise, input IDAMPN cards as follows.

note	columns	variable	entry
(1)	1 - 10	N	Node number; GE.1 and LE.NUMNP
(2)	11 - 20	XDAMP(1)	X-direction damper
	21 - 30	XDAMP(2)	Y-direction damper
	31 - 40	XDAMP(3)	Z-direction damper
(3)	41 - 50	XDAMP(4)	X-rotation damper
	51 - 60	XDAMP(5)	Y-rotation damper
	61 - 70	XDAMP(6)	Z-rotation damper

NOTES/

(1) Input IDAMPN cards; node order is not important. Repeating nodes accumulates dampers at the node.

(2) Damper components input for deleted (or non-existent) degrees of freedom are ignored by the program without a diagnostic message.

(3) Rotational degrees of freedom are currently not used (See Table II.1).

VIII. INITIAL CONDITIONS

Initial conditions for the element are defined in this section. Initial conditions may be established using one (1) of three (3) methods--

METHOD 1 - For MODEX.EQ.2, this is a restart job. Refer to Appendix A for setting up a restart job. The variable "ICON" appearing on the card below is read by the program, but ignored; i.e., the control card (Section VIII.a) must still be input.

METHOD 2 - For MODEX.NE.2, and initial conditions of all zero, input ICON.EQ.0 with no additional data; all vector components are then automatically initialized to zero at time of solution start, TSTART.

METHOD 3 - For MODEX.NE.2, and known non-zero initial conditions, input ICON.EQ.1 and read the system vectors in compacted form from cards as described in Section VIII.b, below.

a. Control Card (I5)

note	columns	variable	entry
(1)	1 - 5	ICON	Flag indicating the type of initial conditions; EQ.0 and MODEX.NE.2, zero initial conditions are generated automatically EQ.1 and MODEX.NE.2, non-zero initial conditions are read from data cards immediately following

b. Card Input of System Vectors (6E12.6)

For the case MODEX.NE.2 and ICON.EQ.1 , the program performs the following read operations:

```
      READ (5,1000)  (DIS(K),K=1,NEQ)
      READ (5,1000)  (VEL(K),K=1,NEQ)
      READ (5,1000)  (ACC(K),K=1,NEQ)
 1000 FORMAT (6E12.6)
```

where DIS/VEL/ACC are the system initial displacement/

velocity/acceleration vectors, respectively. The variable NEQ is the total number of freedoms retained for evaluation; i.e., six (6) times the total nodes minus (-) all deletions provided by fixed boundary condition specifications.

The list of equation numbers can be obtained in Section III (variable PSF) and can be identified conveniently from the displacement (velocity and acceleration) print-out of a previous solution.

For the case of a static solution, the VEL/ACC system initial vectors are not read from card input. A static solution is performed if IMASS.EQ.0 (Section II, card 2).

ELEMENT INPUT

Input as many blocks of data in these sections as there are total element groups. Linear element groups (NEGL,total) are input first, and nonlinear elements (NEGNL groups, total) follow the linear element group data. Therefore, whether the elements in a group are linear or nonlinear, depends on whether the element group belongs to the first NEGL groups or the last NEGNL groups.

In any one group all elements input must be the same type; e.g., if nonlinear TRUSS elements are given as input, then all elements in the group must be nonlinear. Furthermore, in any one group, only one material model can be used, e.g., if the group consists of TRUSS elements, then either the material of all elements in the group is linear elastic, or it is nonlinear elastic. However, a number of different sets of material constants for a specific model can be used.

Since the program is an overlay system, in order to avoid unnecessary manipulation of overlays, it is most efficient to group all element groups of one kind and using one material model together.

IX. TRUSS ELEMENTS

TRUSS elements are two-node members allowed arbitrary orientation in the X,Y,Z system. The TRUSS transmits axial force only, and in general is a six (6) degree of freedom element (i.e., three global translation components at each end of the member), see Fig. IX.1.

1. Element control card (2014)

note	columns	variable	entry
	1 - 4	NPAR(1)	Enter the number "1"
(1)	5 - 8	NPAR(2)	Number of TRUSS elements in this group; GE.1
(2)	9 - 12	NPAR(3)	Type of nonlinear analysis EQ.0 default set to "1" EQ.1 materially nonlinear only EQ.2 updated Lagrangian formulation
(3)	56 - 60	NPAR(15)	Material model number EQ.1 linear elastic EQ.2 nonlinear elastic
(4)	61 - 64	NPAR(16)	Number of different sets of section/material properties GE.1 EQ.0, default set to "1"
(5)	65 - 68	NPAR(17)	Number of material model constants per set EQ.0 if NPAR(15).EQ.1 GE.4 if NPAR(15).EQ.2

NOTES/

(1) TRUSS element numbers begin with one (1) and end with the total number of elements in this group, NPAR(2). Element data are input in Section IX.4, below.

(2) The parameter NPAR(3) is applicable only if the element group is nonlinear. If NPAR(3).EQ.1, no geometric nonlinearities are taken into account, i.e. the geometric stiffness matrix is not included. If NPAR(3).EQ.2 large displacement effects are included in the analysis, but small strains are assumed in the calculation of element forces.

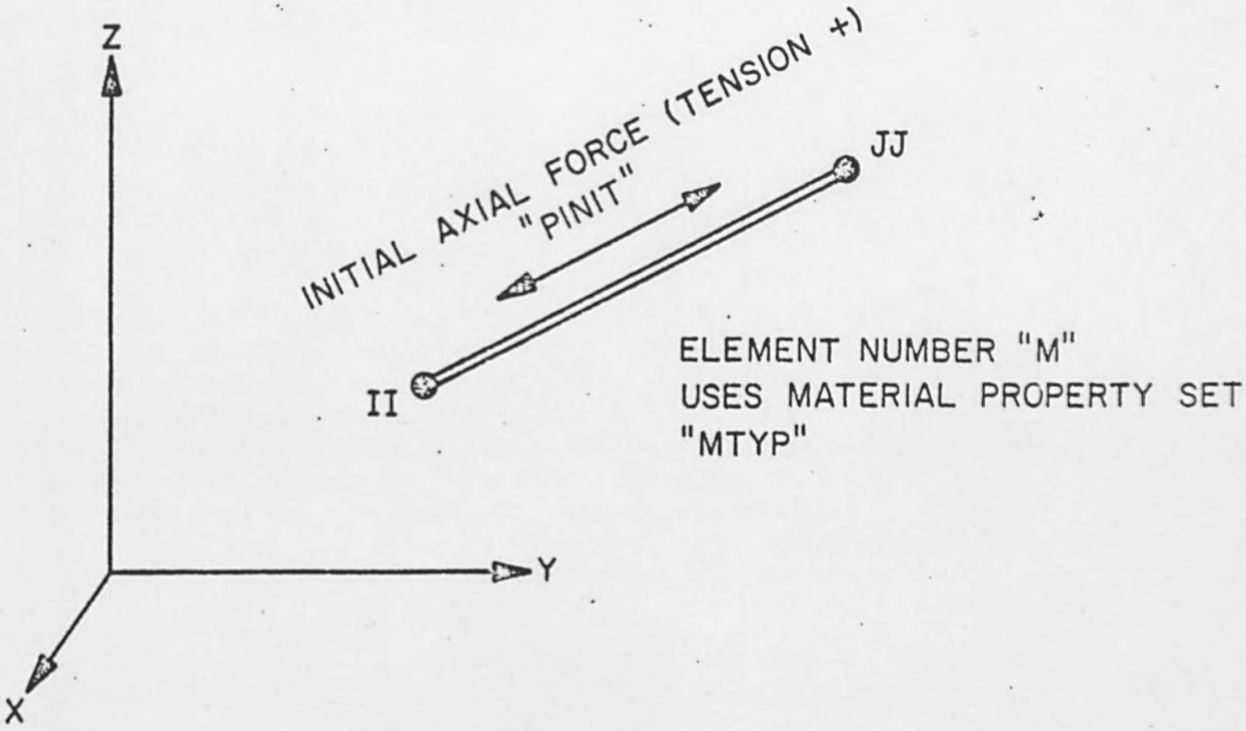


FIGURE IX.1 TRUSS ELEMENT

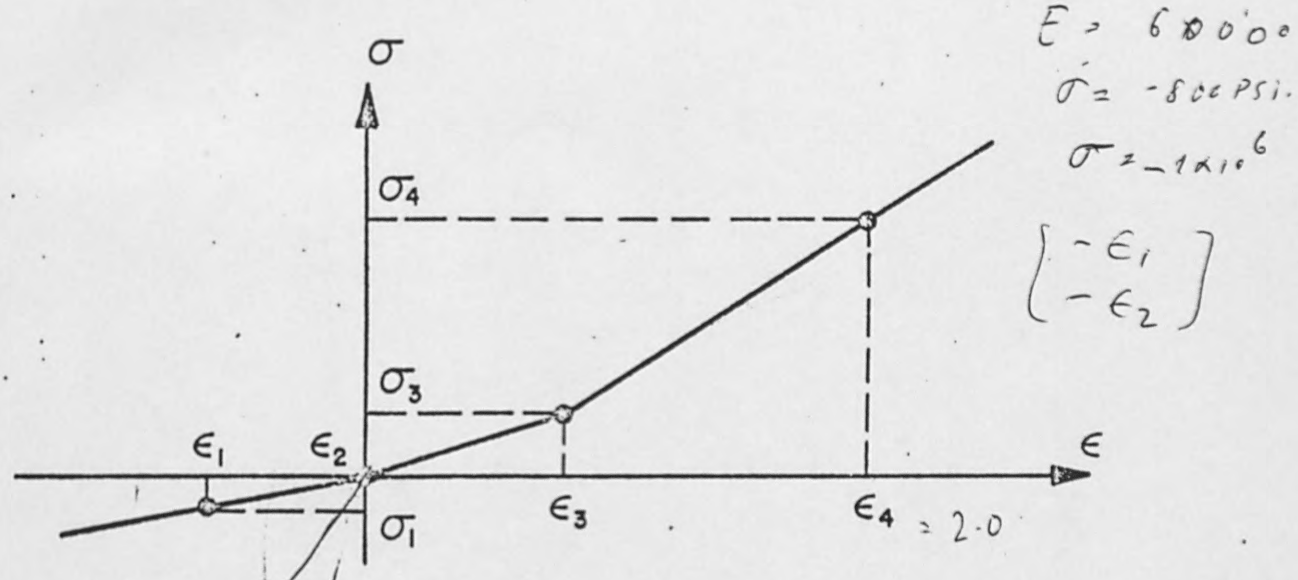


FIGURE IX.2 NONLINEAR MATERIAL MODEL FOR TRUSS

NOTES

(3) In any one element group only one material model can be used, and this model type is defined by the entry NPAR(15). If NPAR(15).EQ.1 the model is defined by Young's modulus only and NPAR(17).EQ.0. If NPAR(15).EQ.2 the stress-strain curve is input in Section IX.3 below.

The model defined for the element group must be consistent with the nonlinear formulation used (defined by NPAR(3)) and the requirement of equilibrium iteration as defined on card 4 of the Master Control Cards (Section II). As stated in note (3) of card 4, Section II, equilibrium iterations can only be performed if the model allows for iteration, and if at least one nonlinear element group is used in the analysis. Table IX.1 summarizes the formulations, material models and possibilities of equilibrium iteration that may be used.

(4) The variable NPAR(16) defines the number of sets of material/section properties to be read in Sections IX.2 and IX.3 below.

(5) NPAR(17) is the variable NCON used in Section IX.3(c).

FORMULATION	MATERIAL MODEL	POSSIBILITY OF EQUILIBIRUM ITERATION
LINEAR ANALYSIS	LINEAR ELASTIC	NO
MATERIALLY NONLINEAR ONLY	NONLINEAR ELASTIC	YES
UPDATED LAGRANGIAN FORMULATION	LINEAR ELASTIC	YES
	NONLINEAR ELASTIC	YES

TABLE IX.1 POSSIBLE ANALYSES USING TRUSS ELEMENTS

IX. TRUSS ELEMENTS (continued)

2. Linear Elastic Material/Section Property Cards

Skip this set of cards if NPAR(15).NE.1 .
Otherwise read NPAR(16) sets of cards.

a. material number card (I5)

columns	variable	entry
1 - 5	N	Material/section number GE.1; and LE.NPAR(16)

b. property card (4F10.0)

note	columns	variable	entry
(1)	1 - 10	E(N)	Young's modulus
	11 - 20	AREA(N)	Cross-sectional area
	21 - 30	DEN(N)	Mass density (for dynamic analysis)
	31 - 40	STRAI(N)	Initial strain (axial)

NOTES/

(1) NPAR(16) different linear elastic materials are input in this section, provided NPAR(15).EQ.1. Note that one material/section is defined to have the same Young's modulus, area, mass density and initial strain.

3. Nonlinear Elastic Material/Section Property Cards

Skip this set of cards if NPAR(15).NE.2, otherwise read NPAR(16) sets of cards.

a. material number card (I5)

columns	variable	entry
1 - 5	N	Material/section number GE.1 and LE.NPAR(16)

b. section property card (3F10.0)

note	columns	variable	entry
(1)	1 - 10	AREA(N)	Cross-sectional area
	11 - 20	DEN(N)	Mass density for dynamic analysis
	21 - 30	STRAI(N)	Initial axial strain

c. stress-strain curve card (8F10.0)

note	columns	variable	entry
(2)	1 - 10	PROP(1,N)	Strain at point 1, e^1
	11 - 20	PROP(2,N)	Strain at point 2, e^2
	.	.	.
	.	.	.
	.	.	.
		PROP(NCON/2,N)	Strain at point NCON/2
		PROP(NCON/2+1,N)	Stress at point 1, σ^1
		PROP(NCON/2+2,N)	Stress at point 2, σ^2
		.	.
		.	.
		.	.
		PROP(NCON,N)	Stress at point NCON/2

NOTES/

(1) One section property card is defined to have the same area, density and initial strain.

NOTES/(continued)

(2) The stress-strain curve is defined by straight lines between the input points (e^i,σ^i). From the stress-strain curve total stresses and the tangent modulus are evaluated for a given strain (see Fig. IX.2).

The variable NCON was defined in Section IX.1 by the variable NPAR(17).

This model can only be used in a nonlinear element group.

IX. TRUSS ELEMENTS (continued)

4. Element Data Cards (4I5,F10.0,3I5)

NPAR(2) elements must be input and/or generated in this section in ascending sequence beginning with "1".

note	columns	variable	entry
	1 - 5	M	TRUSS element number; GE.1 and LE.NPAR(2)
(1)	6 - 10	II	Node number at one end
	11 - 15	JJ	Node number at other end GE.1 and LE.NUMNP
	16 - 20	MTYP	Material property set number; GE.1 and LE.NPAR(16)
(2)	21 - 30	PINIT	Initial axial force in the TRUSS; GT.0, tension
	31 - 35	IPS	Flag for printing axial stress in TRUSS element; EQ.0, no printing EQ.1, print element stress
(3)	36 - 40	KG	Node generation increment used to compute node numbers for missing elements; EQ.0, default set to "1"

NOTES/

(1) Refer to Figure IX.1.

(2) PINIT is the axial force in the TRUSS at zero node displacements, and zero initial strain in the truss. The initial strain defined for the material property set number gives rise to an additional force in the truss.

(3) Elements must be input in increasing element number order. If cards for elements [M+1,M+2,...,M+J] are omitted, these "J" missing elements are generated using MTYP and PINIT of element "M" and by incrementing the node numbers of successive elements with the value "KG"; KG is taken from the first card of the element generation sequence (i.e., from the "M-th" element card).

X. 2/D CONTINUUM ELEMENTS

2/D CONTINUUM elements are 4- to 8-node isoparametric quadrilaterals which must be input in the global Y-Z plane. Figure X.1 shows some typical 2/D CONTINUUM elements. The element can represent either planar (plane stress or plane strain condition) or axisymmetric solids, as illustrated in Fig. X.2. In both cases, each element node has two (2) translational degrees of freedom.

When the element is used to represent an axisymmetric solid or shell, the global Z-axis is the axis of revolution. All elements must be located in the +Y half-plane. These conventions are illustrated in Fig. X.2.

X. 2/D CONTINUUM ELEMENTS (continued)

1. Element Group Control Card (20I4)

note	columns	variable	entry
	1 - 4	NPAR(1)	Enter the number "2"
(1)	5 - 8	NPAR(2)	Number of 2/D CONTINUUM elements in this group; GE.1
(2)	9 - 12	NPAR(3)	Code indicating type of nonlinear analysis; EQ.0, default set to "1" EQ.1, material nonlinear analysis only EQ.2, total Lagrangian EQ.3, updated Lagrangian
	17 - 20	NPAR(5)	Element type code; EQ.0, axisymmetric EQ.1, plane strain EQ.2, plane stress
(3)	25 - 28	NPAR(7)	Maximum number of nodes used to describe any one element; GE.4 and LE.8 EQ.0, default set to "8"
(4)	37 - 40	NPAR(10)	Numerical integration order to be used in Gauss quadrature formulae; EQ.0, default set to "2" GE.2 and LE.4
(5)	49 - 52	NPAR(13)	Number of stress output location tables; EQ.0, print stresses at integration points
(6)	57 - 60	NPAR(15)	Material model number; GE.1 and LE.12 EQ.1, linear isotropic EQ.2, linear orthotropic EQ.3, variable tangent moduli EQ.4, curve description model EQ.5, curve description model with tension cut-off (plane strain only) EQ.6, elastic-plastic (von Mises)

X. 2/D CONTINUUM ELEMENTS (continued)

note	columns	variable	entry
			EQ.7, elastic-plastic (Drucker-Prager) EQ.8, incompressible nonlinear elastic (Mooney-Rivlin/ plane stress only) GE.9, user-supplied
(6)	61 - 64	NPAR(16)	Number of different sets of material properties; GE.1
(6)	65 - 68	NPAR(17)	Number of constants per property set; EQ.0, if NPAR(15).LT.9 GE.1, if NPAR(15).GE.9
(6)	69 - 72	NPAR(18)	Dimension of storage array required for element history; EQ.0, if NPAR(15).LT.9 GE.0, if NPAR(15).GE.9

NOTES/

(1) 2D/CONTINUUM element numbers begin with one (1) and end with the total number of elements in this group, NPAR(2). Element data are input in Section X.4 below.

(2) NPAR(3) is applicable for nonlinear element groups only and determines if geometrical nonlinearities are to be included in the analysis. If NPAR(3).EQ.1 displacements and strains are assumed to be infinitesimal. In the total Lagrangian and updated Lagrangian formulations all geometric effects (large displacements and large strains) are included in the analysis.

(3) NPAR(7) limits the number of nodes that can be used to describe any of the elements in this group. A minimum of 4 and a maximum of 8 nodes are used to describe the 2/D CONTINUUM elements. For example, for the set of elements shown in Fig. X.1, NPAR(7) would be set to "7". Constant strain triangles are obtained by having nodes 3 and 4 coincide.

(4) For rectangular elements an integration order of "2" is sufficient. If the element is distorted, a higher integration order need be used. Notice that apart from the larger computational effort in the calculation of the element matrices, more working storage may be required if a nonlinear material model is used (see note (6) below). The consistent mass matrix is always calculated with an integration order of 3.

NOTES/ (5) Element stresses are calculated at the points defined in the stress output location table assigned to the element (see Section X.3). NPAR(13) defines the total number of stress output location tables input in Section X.3. NPAR(13).EQ.0 for all nonlinear element models, i.e. MODEL.GT.2 in Section X.2.

(6) Only one material model (defined by the value of NPAR(15)) is allowed in an element group. If NPAR(15) is 1,2,...,8, the model exists in the current library for 2/D CONTINUUM elements, and the entries for NPAR(17) and NPAR(18) are ignored by the program. If, however, NPAR(15) is "9" or larger, material constants must be read into a storage array, "PROP", which has dimensions NPAR(17), NPAR(16); i.e., property constants are stored as,

```
((PROP(I,J),I=1,NPAR(17)),J=1,NPAR(16))
```

Non-library models require that NPAR(17) be given (input) as a positive integer. In addition, NPAR(18) must be specified for non-standard nonlinear models for use in allocating storage in a working array, "WA", which contains element history. The NPAR(18) parameters which characterize element history must be retained for materially nonlinear elements so that properties can be chosen in the current solution step. The working storage array (WA) is dimensioned IDWA by NPAR(2); where

```
IDWA   =  NPAR(18)*(NPAR(10)**2)
NPAR(2)  =  number of elements
```

i.e. for each integration point of each element, NPAR(18) storage locations are preserved during the solution. See Appendix B for a discussion on how to incorporate a new material model.

The model defined for the element group must be consistent with the nonlinear formulation used (defined by NPAR(3)), the element type (defined by NPAR(5)), and the requirement of equilibrium iteration as defined on card 4 of the Master Control Cards (Section II). As stated in note (3) of card 4, Section II, equilibrium iterations can only be performed if the model allows for iterations, and if at least one nonlinear element group is used in the analysis. Table X.1 summarizes the formulations, material models, and iteration possibilities for 2/D CONTINUUM elements.

FORMULATION	MATERIAL MODEL	POSSIBILITY OF EQUILIBRIUM ITERATION
LINEAR ANALYSIS	linear isotropic	NO
	linear orthotropic	
MATERIALLY NONLINEAR ONLY	variable tangent moduli model	NO
	curve description model (plane strain or axisymmetric only)	NO
	curve description model with tension weakening (plane strain only)	NO
	elastic-plastic, von Mises yield condition	YES
	elastic-plastic, Drucker-Prager yield condition	YES
TOTAL LAGRANGIAN FORMULATION	linear isotropic	YES
	linear orthotropic	YES
	variable tangent moduli model	NO
	elastic-plastic, von Mises yield condition	YES
	incompressible non-linear elastic, Mooney-Rivlin material, (plane stress only)	YES
UPDATED LAGRANGIAN FORMULATION	linear isotopic	YES

X. 2/D CONTINUUM ELEMENTS (continued)

2. Material Property Data

NPAR(16) sets of cards must be input in this section. Card "a" (material number card") is the same for all material models, but card(s) "b" ("material property card(s)") depend on the material model number (NPAR(15).

a. material number card (I5,F10.0)

note	columns	variable	entry
	1 - 5	N	Material property set number; GE.1 and LE.NPAR(16)
(1)	6 - 15	DEN(N)	Mass density of the material used in calculation of the mass matrix, ρ; GE.0.0

NOTES/

(1) The mass density defined is used directly in the calculation of the element matrix, i.e., no acceleration constants are applied to the variable DEN(N).

b. material property card(s) (8F10.0)

For MODEL "1" (NPAR(15).EQ.1) → [linear isotropic]

note	columns	variable	entry
(1)	1 - 10	PROP(1,N)	Young's modulus, E
	11 - 20	PROP(2,N)	Poisson's ratio, ν

NOTES/

(1) MODEL 1 is a linear library model defined by two (2) positive constants (E,ν); i.e., if NPAR(15).EQ.1, NPAR(17) is set to "2" by default. MODEL 1 can be used with linear or nonlinear element groups. Since the material constants are independent of history, NPAR(18) is set to "0" by default.

For MODEL "2" (NPAR(15).EQ.2) → [linear orthotropic]

note	columns	variable	entry
(1)	1 - 10	PROP(1,N)	a-direction modulus, E_a
	11 - 20	PROP(2,N)	b-direction modulus, E_b
	21 - 30	PROP(3,N)	c-direction modulus, E_c
	31 - 40	PROP(4,N)	strain ratio, ν_{ab}
	41 - 50	PROP(5,N)	strain ratio, ν_{ac}
	51 - 60	PROP(6,N)	strain ratio, ν_{bc}
	61 - 70	PROP(7,N)	shear modulus, G_{ab}

NOTES/

(1) MODEL 2 is linear (i.e., NPAR(18).EQ.0 by default). The model requires seven(7) property constants; i.e., NPAR(17) .EQ.7 by default if NPAR(15).EQ.2. The constants (E_a,E_b, ...,G_{ab}) are defined in a material coordinate system (a,b,c) in which "c" corresponds to the out-of-plane direction (or global X), and the a-b plane coincides with the plane of the elements (Y-Z). a,b are principal material axes, and β is the angle (in degrees) between element edge 1-2 and the material +a-axis (Figure X.3). The material angle β is read individually for each element in Section X.4, below. MODEL 2 can be used with linear or geometrically nonlinear elements.

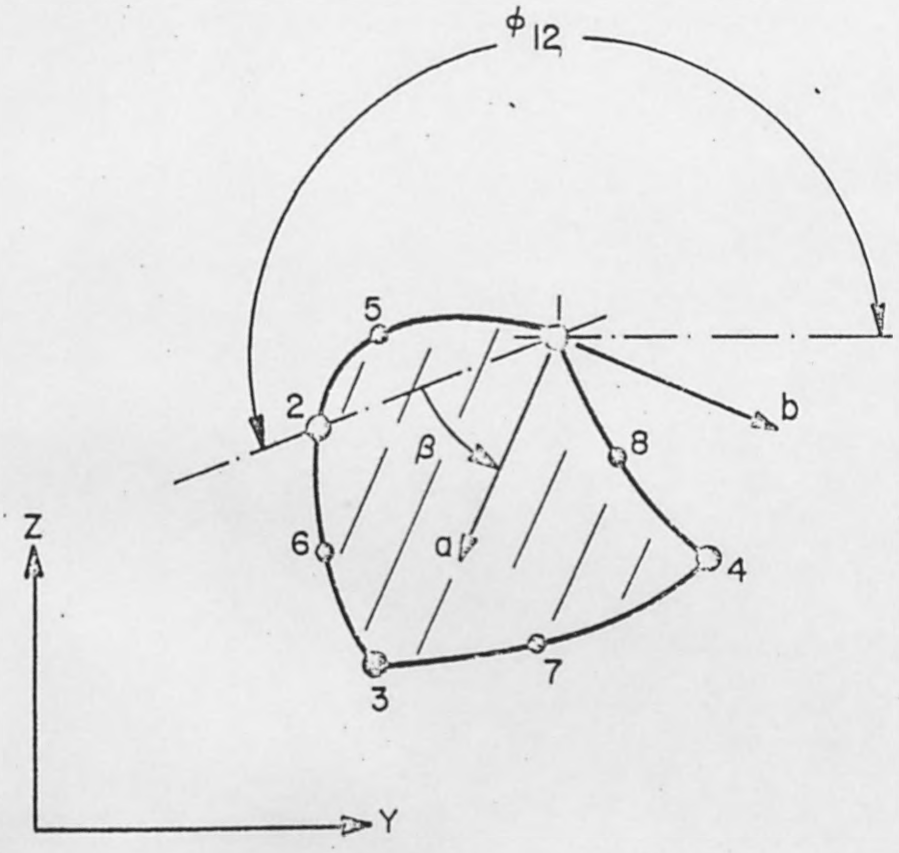


FIGURE X.3 PRINCIPAL IN-PLANE MATERIAL AXES ORIENTATION FOR THE LINEAR ORTHOTROPIC MATERIAL MODEL

For MODEL "3" (NPAR(15).EQ.3) → [variable tangent moduli]

note	columns	variable	entry	
(1)	1 - 10	PROP(1,N)	Unloading bulk modulus, K_{UN}	
	11 - 20	PROP(2,N)	K_0	Constants to define the loading bulk modulus
	21 - 30	PROP(3,N)	K_1	
	31 - 40	PROP(4,N)	K_2	
	41 - 50	PROP(5,N)	G_0	Constants to define the loading and unloading shear moduli
	51 - 60	PROP(6,N)	$\gamma_1 (<0)$	
	61 - 70	PROP(7,N)	$\bar{\gamma}_1 (<0)$	

NOTES/

(1) MODEL 3 is a nonlinear library material model requiring seven (7) constants (i.e., NPAR(17).EQ.7 by default) per property set. Ten variables, namely the previously calculated stresses, strains, the maximum pressure ever reached and a loading parameter, need be stored per integration point, i.e., NPAR(18) = 10 for this model.

The material model cannot be used in plane stress analysis.

NOTATION

For the notation used in the description of the nonlinear material models refer to the report "Static and Dynamic Geometric and Material Nonlinear Analysis," SESM Report No. 74-4 , February 1974.

X. 2/D CONTINUUM ELEMENTS (continued)

For MODEL "4" or "5" (NPAR(15).EQ.4) or (NPAR(15).EQ.5) → [curve description model]

a. Material moduli cards (supply NPAR(16) sets of cards)

note	columns	variable	entry	
first card				
(1)	1 - 10	PROP(1,N)	e_V^1	volume strain at point "1"
	11 - 20	PROP(2,N)	e_V^2	....
	21 - 30	PROP(3,N)	e_V^3	
	31 - 40	PROP(4,N)	e_V^4	
	41 - 50	PROP(5,N)	e_V^5	
	51 - 60	PROP(6,N)	e_V^6	
	61 - 70	PROP(7,N)	K_{LD}^1	loading bulk modulus at point "1"
	71 - 80	PROP(8,N)	K_{LD}^2	....
second card				
	1 - 10	PROP(9,N)	K_{LD}^3	
	11 - 20	PROP(10,N)	K_{LD}^4	
	21 - 30	PROP(11,N)	K_{LD}^5	
	31 - 40	PROP(12,N)	K_{LD}^6	
	41 - 50	PROP(13,N)	K_{UN}^1	unloading bulk modulus at point "1"
	51 - 60	PROP(14,N)	K_{UN}^2	....
	61 - 70	PROP(15,N)	K_{UN}^3	
	71 - 80	PROP(16,N)	K_{UN}^4	
third card				
(2)	1 - 10	PROP(17,N)	K_{UN}^5	
	11 - 20	PROP(18,N)	K_{UN}^6	
	21 - 30	PROP(19,N)	G_{LD}^1	loading shear modulus at point "1"
	31 - 40	PROP(20,N)	G_{LD}^2	....
	41 - 50	PROP(21,N)	G_{LD}^3	
	51 - 60	PROP(22,N)	G_{LD}^4	
	61 - 70	PROP(23,N)	G_{LD}^5	
	71 - 80	PROP(24,N)	G_{LD}^6	

NOTES/

(1) MODELS 4 and 5 are nonlinear material models for which NPAR(17).EQ.24, by default. To characterize history, twelve (12) variables, namely, the previously calculated stresses and strains, the maximum volumetric strain ever reached in loading, ground pressure, the volume strain due to ground pressure, and the angle of "cracking" need be saved for each integration point, i.e. NPAR(18).EQ.12.

Note that

$$e_V^J > e_V^{J-1}, \quad K_{UN}^J \geq K_{LD}^J \quad \text{and} \quad G_{LD}^J < 1.5\, K_{LD}^J$$

where "J" is the J-th table input point. All six (6) input points must be defined, as shown in Fig. X.4 Linear interpolation is used to define the loading and unloading bulk moduli, ${}^tK_{LD}$, ${}^tK_{UN}$, and the loading shear modulus, ${}^tG_{LD}$, at time t. Note that these three cards must be supplied NPAR(16) times.

(2) The unloading shear modulus is calculated as $({}^tK_{UN}/{}^tK_{LD})\ {}^tG_{LD}$.

b. Tension cut-off definition card

This card defines the parameters to be used in the tension cut-off model, i.e., MODEL 5. Omit this card if MODEL.EQ.4.

note	columns	variable	entry
(1)	1 - 10	GAMMA	material density
(2)	11 - 20	STIFAC	normal stiffness reduction factor
	21 - 30	SHEFAC	shear stiffness reduction factor

NOTES/

(1) GAMMA is used to calculate the in-situ gravity pressure at the element integration points. It is assumed that the material provides a hydrostatic pressure distribution, calculated at an element integration point using

$$p = \sum_{i=1}^{N} h_i\, p_i$$

where h_i are the element interpolation functions, p_i is the pressure at the element nodes, and N is the total number of

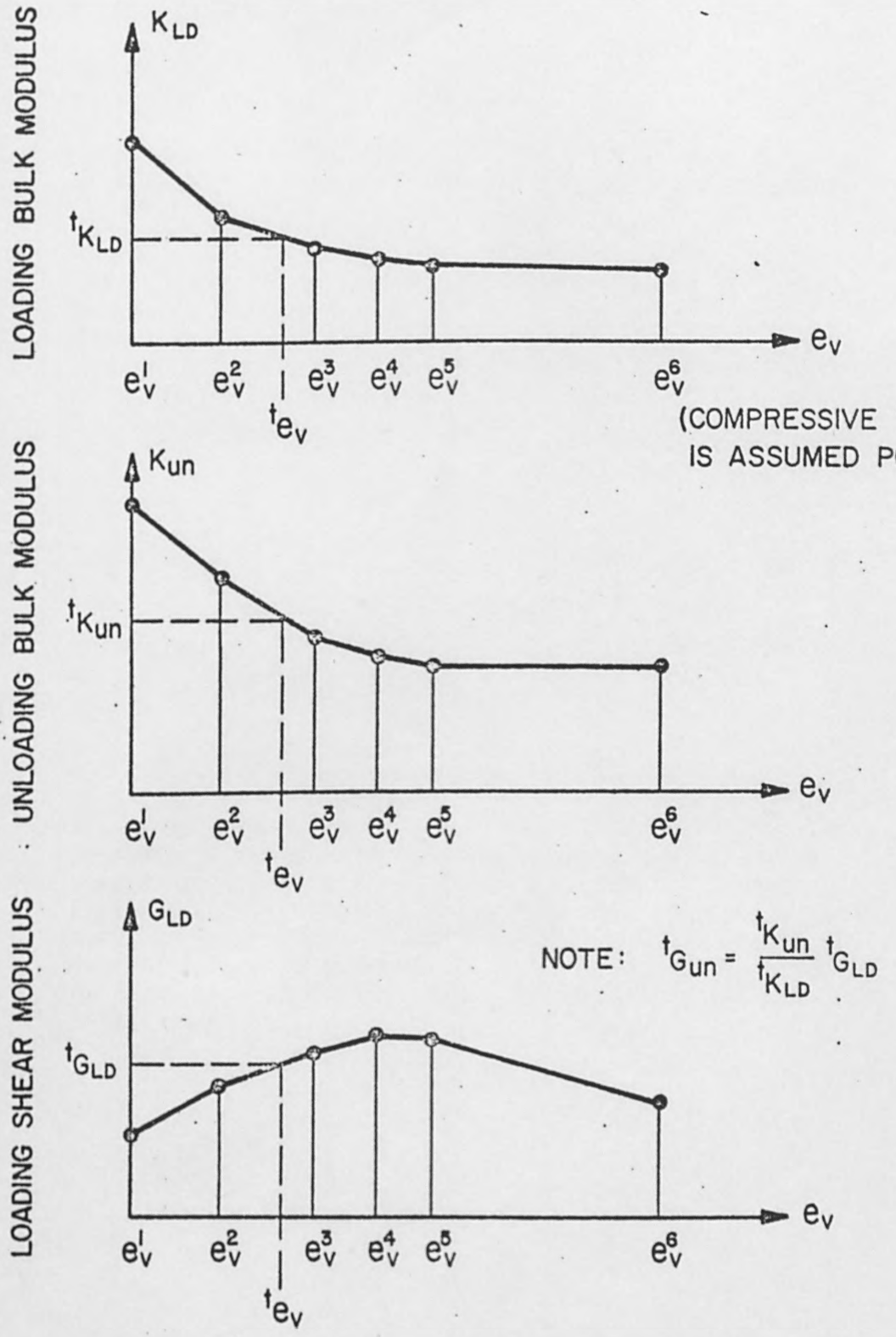


FIGURE X.4 MODULI VERSUS VOLUME STRAIN FOR THE CURVE DESCRIPTION MODEL

NOTES/(continued)

(1) element nodes. The pressure p_i is calculated using P_i = GAMMA * Z_i, where Z_i is the Z-coordinate of node i. This convention defines the assumed orientation of the Y-Z coordinate system. (All two-dimensional analyses must be carried out in the Y-Z plane.) From these hydrostatic pressures and the bulk moduli input above the in-situ volume strain at every integration point is calculated. When finding bulk and shear moduli from the curve description shown in Fig. X.4, in-situ volume strain is included. That is, for MODEL 5

$$t_{e_V} = {}^{t}e_V^P + {}^{t}e_V^L$$

where ${}^{t}e_v$ is total volume strain (abscissa of curves in Fig. X.4), ${}^{t}e_V^P$ is the in-situ volume strain, and ${}^{t}e_V^L$ is the volume strain due to applied loads. Note that for MODEL 4 t_{e_V} is assumed to equal ${}^{t}e_V^L$.

(2) MODEL 5 contains the assumption that the material cannot withstand tensile stresses. When the tensile principal stress at an integration point exceeds the in-situ gravity pressure, a crack forms perpendicular to this principal stress and the stiffness across the crack is reduced by the factor STIFAC. Also, the shear stiffness at that integration point is reduced by the factor SHEFAC, which is meant to model the loss in shear strength due to cracking. Model 5 is to be used with plane strain elements only and unloading is not permitted.

For MODEL "6" (NPAR(15).EQ.6) → [elastic-plastic material, von Mises yield condition]

note	columns	variable	entry
(1)	1 - 10	PROP(1,N)	Young's modulus, E
(1)	11 - 20	PROP(2,N)	Poisson's ratio, ν
(2)	21 - 30	PROP(3,N)	Yield stress in simple tension, σ_y
(3)	31 - 40	PROP(4,N)	Strain hardening modulus, E_t

NOTES/

(1) MODEL 6 is a nonlinear library material model for which NPAR(17).EQ.4 by default. NPAR(18).EQ.10, since 10 history-dependent variables must be stored for each integration point.

The Young's modulus and Poisson's ratio define the initial elastic behavior of the material.

(2) The yield stress in simple tension defines the initial yield conditions.

(3) Linear isotropic strain hardening is assumed and E_t is the tangential modulus (after yield) obtained from a uniaxial tension test.

For MODEL "7" (NPAR(15).EQ.7) → [elastic-plastic material, Drucker-Prager yield condition]

note	columns	variable	entry
(1)	1 - 10	PROP(1,N)	Young's modulus, E
	11 - 20	PROP(2,N)	Poisson's ratio, ν
	21 - 30	PROP(3,N)	Angle of friction, θ
	31 - 40	PROP(4,N)	Cohesion, C

NOTES/

(1) MODEL 7 is a nonlinear library material model requiring four (4) constants (i.e., NPAR(17).EQ.4 by default) per material property set. Eleven (11) parameters need to be stored per integration point, i.e. NPAR(18).EQ.11 by default. The model assumes elastic perfectly plastic conditions.

For MODEL "8" (NPAR(15).EQ.8) → [incompressible nonlinear elastic material, Mooney-Rivlin model]

note	columns	variable	entry
(1)	1 - 10	PROP(1,N)	C_1
	11 - 20	PROP(2,N)	C_2

NOTES/

(1) MODEL 8 is a nonlinear library material model for which NPAR(17) is set to "2". Since the material is elastic, no history-dependent variables need be stored at integration points and NPAR(18) is set to "0". This rubber model can only be used in plane stress analysis.

For MODEL "≥9" (NPAR(15).GE.9) → (user-supplied material model)

note	columns	variable	entry
(1)	1 - 10	PROP(1,N)	
	11 - 20	PROP(2,N)	
	21 - 30	PROP(3,N)	
	⋮	etc.	

NOTES/

(1) MODEL "≥9" is a non-standard model (i.e., is not in the material model library) and requires that property data be read

(PROP(J,N),J=1,NPAR(17),N=1,NPAR(16))

where NPAR(17) is user-supplied (i.e., no default value exists). Subroutines which allocate storage locations (in "WA") for element history (if the model is nonlinear) and which calculate the incremental stress-strain law and current stresses must be supplied. For a discussion on how to incorporate a user-supplied material model, see Appendix B.

X. 2/D CONTINUUM ELEMENTS (continued)

3. Stress Output Table Cards (9I5)

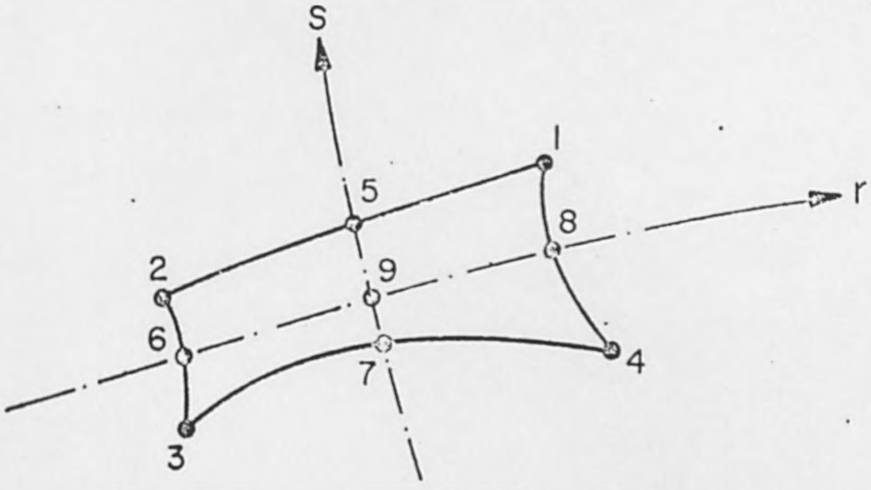
Skip this section if stresses at integration points are to be printed, i.e. NPAR(13).EQ.0; otherwise supply NPAR(13) cards.

note	columns	variable	entry
(1)	1 - 5	ITABLE(N,1)	Stress output location 1
	6 - 10	ITABLE(N,2)	Stress output location 2
	11 - 15	ITABLE(N,3)	Stress output location 3
	16 - 20	ITABLE(N,4)	Stress output location 4
	21 - 25	ITABLE(N,5)	Stress output location 5
	26 - 30	ITABLE(N,6)	Stress output location 6
	31 - 35	ITABLE(N,7)	Stress output location 7
	36 - 40	ITABLE(N,8)	Stress output location 8
	41 - 45	ITABLE(N,9)	Stress output location 9

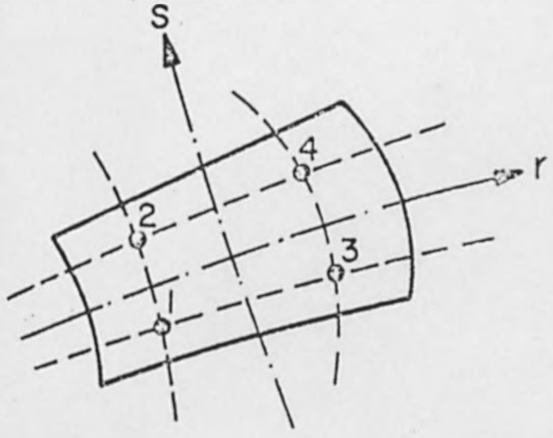
NOTES/

(1) Stress tables are defined to provide flexibility in element stress output requests. Each element can refer to one of the tables defined, and the element stresses are then calculated at the points specified in the table. Refer to Fig. X.5 for selection of the stress calculation points. The first "0" entry in a table will terminate that table. For example, if ITABLE(N,1).EQ.7 and ITABLE(N,2).EQ.0, then stresses will be printed at point 7 only whenever this stress table is referred to.

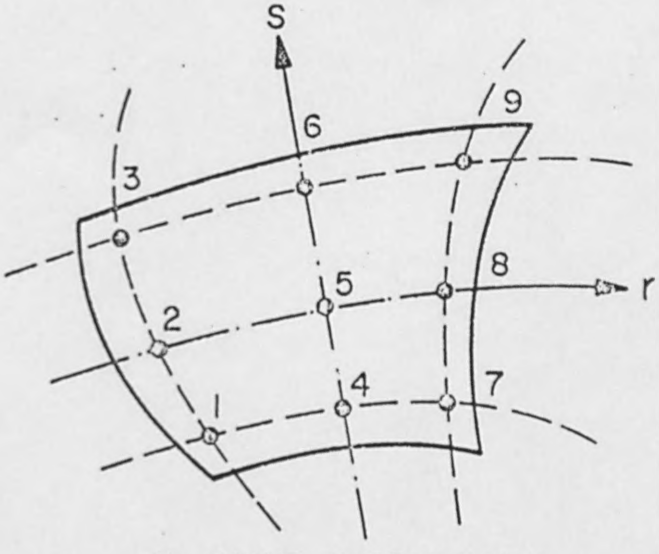
The stress tables are only used when MODEL "1" or "2" define the material behavior.

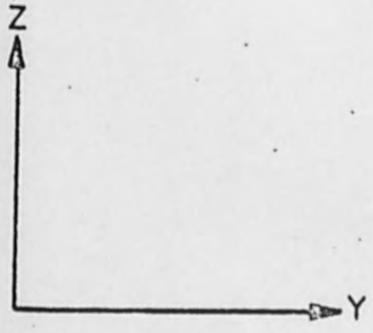


STRESS PRINTOUT TABLE POINTS FOR
2/D CONTINUUM ELEMENTS



2-POINT INTEGRATION





3-POINT INTEGRATION

PRINTING SEQUENCE FOR INTEGRATION
POINT STRESSES FOR 2/D CONTINUUM ELEMENTS

FIGURE X.5 STRESS PRINT-OUT CONVENTION
FOR 2/D CONTINUUM ELEMENTS

X. 2/D CONTINUUM ELEMENTS (continued)

4. Element Data Cards (I5,I3,I2,2F10,10I5)

note	columns	variable	entry
(1)	1 - 5	M	2/D CONTINUUM element number; GE.1 and LE.NPAR(2)
(2)	6 - 8	IEL	Number of nodes used to describe this element; EQ.0, default set to "NPAR(7)" LE.NPAR(7)
(3)	9 - 10	IPS	Number of the stress table to be used for stress calculations; EQ.0, no stress output for this element
	11 - 20	BET	Material angle, ß, in degrees to be used in connection with material model "2" (linear orthotropic)
(4)	21 - 30	THIC	Element thickness; GT.0, plane stress only
	31 - 35	MTYP	Material property set number assigned to this element; GE.1 and LE.NPAR(16)
(1)	36 - 40	KG	Node generation parameter used to compute node numbers for missing elements (given on first card of a sequence); EQ.0, default set to "1"
(2)	41 - 45	NOD(1)	Global node number of element nodal point 1
	46 - 50	NOD(2)	Global node number of element nodal point 2
	51 - 55	NOD(3)	Global node number of element nodal point 3
	56 - 60	NOD(4)	Global node number of element nodal point 4
	61 - 65	NOD(5)	Global node number of element nodal point 5
	66 - 70	NOD(6)	Global node number of element nodal point 6
	71 - 75	NOD(7)	Global node number of element nodal point 7

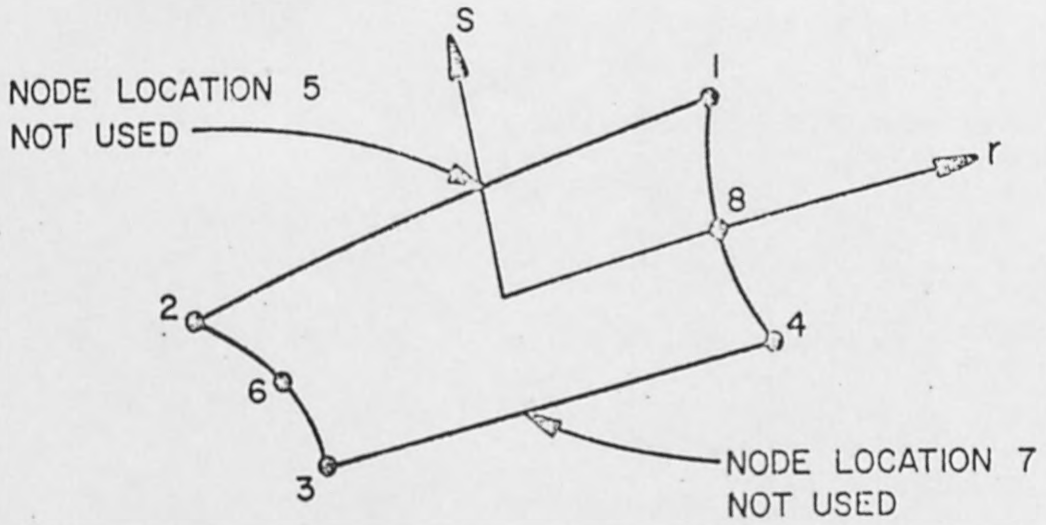
note	columns	variable	entry
	76 - 80	NOD(8)	Global node number of element nodal point 8

NOTES/

(1) Elements must be input in ascending element number order. If data cards for elements [M+1,M+2,...,M+J] are omitted, these "J" missing elements are generated using IEL,IPS,BET, THIC, and MTYP given on the card for element "M" and by incrementing node numbers of successive elements with the value "KG"; the value of KG used for incrementation is taken from the M-th element card, and only the non-zero nodes appearing on the M-th element card are incremented when generating missing element data.

(2) The number of nodes in element "M" is defined by "IEL". However, all 8 entries for NOD(I) are read from the element data card; if IEL.LT.8 the particular node locations not used in this element need be input as "0" in NOD(I). Fig. X.6 defines the input sequence that must be observed for element node input.

For example, the 6-node element (IEL.EQ.6) shown below



is defined by

NOD(I) = [X X X X 0 X 0 X]

XI. 3/D CONTINUUM ELEMENTS (continued)

1. Element Group Control Card (2014)

note	columns	variable	entry
	1 - 4	NPAR(1)	Enter the number "3"
(1)	5 - 8	NPAR(2)	Number of 3/D CONTINUUM elements in this group; GE.1
(2)	9 - 12	NPAR(3)	Flag indicating type of nonlinear analysis; EQ.0, default set to "1" EQ.1, material nonlinear analysis only EQ.2, total Lagrangian EQ.3, updated Lagrangian
(3)	25 - 28	NPAR(7)	Maximum number of nodes used to describe any one element; EQ.0, default set to "21"
(4)	37 - 40	NPAR(10)	Numerical integration order to be used in Gauss quadrature formula for r-s plane; EQ.0, default set to "2" GE.2 and LE.4
(4)	41 - 44	NPAR(11)	Numerical integration order to be used in Gauss quadrature formula for t-direction; EQ.0, default set to "2" GE.2 and LE.4
(5)	49 - 52	NPAR(13)	Number of stress output location tables; EQ.0, print stresses at integration points
(6)	57 - 60	NPAR(15)	Material model number; GE.1 and LE.8 EQ.1, linear elastic isotropic EQ.3, curve description model
(6)	61 - 64	NPAR(16)	Number of different sets of material properties; GE.1

XI. 3/D CONTINUUM ELEMENTS (continued)

note	columns	variable	entry
(6)	65 - 68	NPAR(17)	Number of constants per material property set; EQ.0, if NPAR(15).LE.3 GE.1, if NPAR(15).GT.3
(6)	69 - 72	NPAR(18)	Dimension of storage array required for element history; EQ.0, if NPAR(15).LE.3 GE.0, if NPAR(15).GT.3

NOTES/

(1) 3/D CONTINUUM element numbers begin with one (1) and end with the total number of elements in this group, NPAR(2). Data for the individual elements are input in Section XI.4 below.

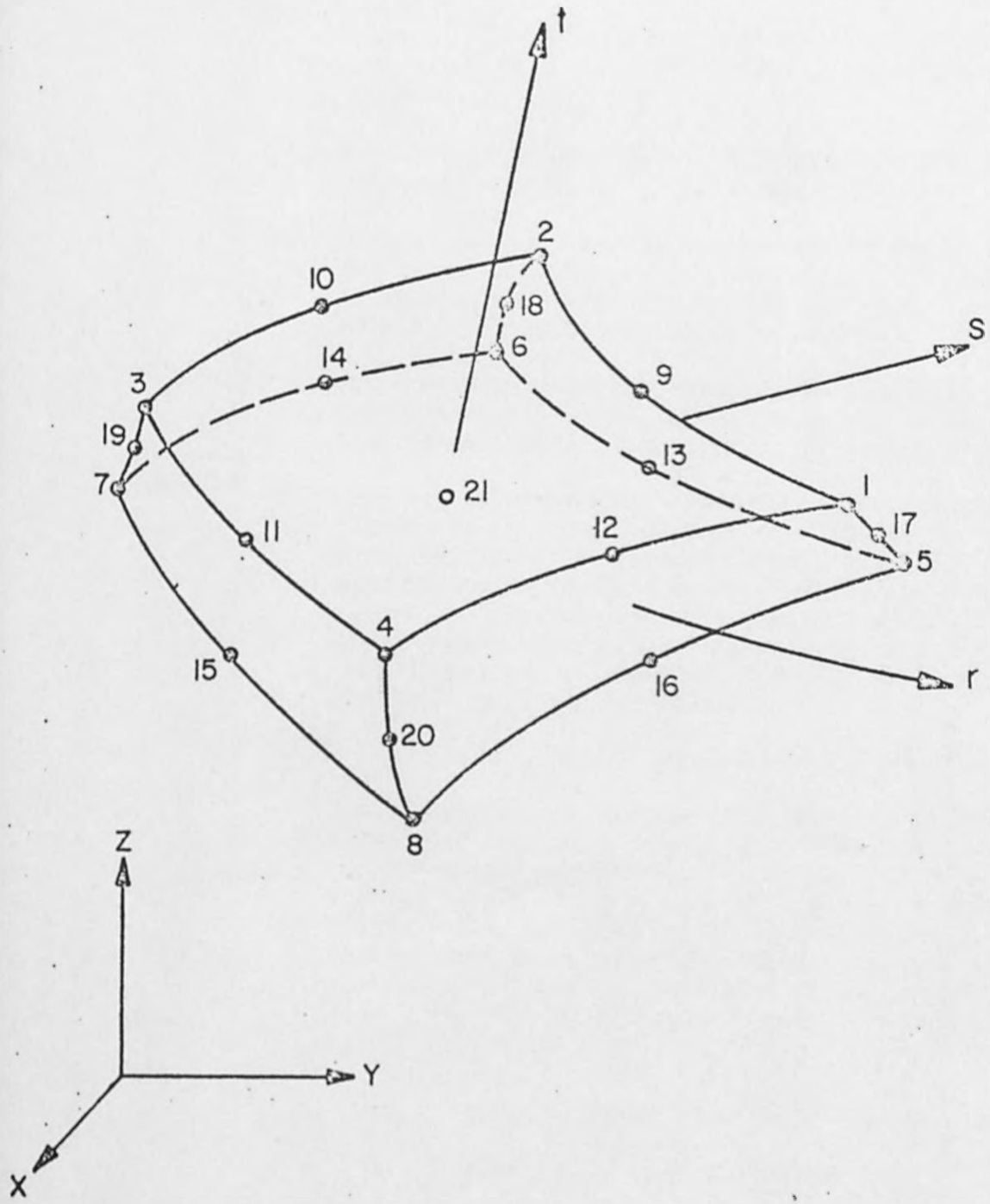
(2) NPAR(3) is applicable for nonlinear element groups only and determines the type of geometrical nonlinearities to be considered in the analysis. In the present version of NONSAP only material nonlinearities are included in 3/D CONTINUUM elements; hence, NPAR(3) must be specified as one (1).

(3) NPAR(7) is the maximum number of nodes that can be used to describe any one of the elements in this group, i.e. elements in this group must have less than or equal to NPAR(7) nodes. A minimum of 8 and a maximum of 21 nodes are used to describe 3/D CONTINUUM elements as indicated in Fig. XI.3. Constant strain tetrahedra can be formed from 8-node elements by having nodes 1,2,3, and 4 coincide and nodes 7 and 8 coincide.

(4) The selection of appropriate integration orders depends on the element shape and strain state being considered. When the quantities being integrated vary irregularly a higher order is needed. An integration order of "2" is sufficient for most problems. Shell or plate problems often use thin elements in which strain varies more or less linearly through the thickness, but more irregularly in the plane of the surface of the shell. In these cases it is advantageous to specify a lower order for integration through the thickness, i.e., NPAR(11) < NPAR(10).

The expense of stiffness formation is dependent on the integration order, i.e.,

$$n = NPAR(11) * NPAR(10) * NPAR(10)$$



JRE XI.3 3/D CONTINUUM ELEMENT NODE NUMBERING CONVENTION

NOTES/(4) where n is the number of points at which $B^T DB$ must be calculated in order to find the element stiffness by Gauss integration.

The consistent mass matrix is always calculated using an integration order of three in each direction.

(5) Element stresses are calculated at the points defined in the stress output location table (see Section XI.3) assigned to an element. NPAR(13) defines the total number of stress output location tables input in Section XI.3.

For nonlinear material models (NPAR(15).GE.3) stress output tables cannot be used, and stresses are printed at integration points; NPAR(13) is set to "0" in these cases.

(6) Only one material model (defined by the value of NPAR(15)) is allowed in an element group. If NPAR(15) is 1 or 3, the model exists in the current library for 3/D CONTINUUM elements, and the entries for NPAR(17) and NPAR(18) are ignored by the program. If, however, NPAR(15) is "4" or larger, material constants must be read into a storage array, "PROP", which has dimensions NPAR(17), NPAR(16); i.e., property constants are stored as,

(PROP(I,J),I=1,NPAR(17),J=1,NPAR(16)

Non-library models require that NPAR(17) be given (input) as a positive integer. In addition, NPAR(18) must be specified for non-standard nonlinear models for use in allocating storage in a working array, "WA", which contains element history. The NPAR(18) parameters which characterize element history must be retained for materially nonlinear elements so that properties can be chosen in the current solution step. The working storage array (WA) is dimensioned IDWA by NPAR(2), where

IDWA = NPAR(18) * (NPAR(10)**2) * NPAR(11)

NPAR(2) = number of elements

i.e., for each integration point of each element NPAR(18) storage locations are preserved during the solution. See Appendix B for the procedure to incorporate a new material model.

The model defined for the element group must be consistent with the nonlinear formulation used (NPAR(3)) and the requirement of equilibrium iteration as defined on Card 4 of the Master Control Cards (Section II). Equilibrium

NOTES/(6)

iteration can be performed only if the material model allows for it and if there is at least one nonlinear element group. Table XI.1 summarizes the possible analyses with the 3/D CONTINUUM material models implemented in the current version of NONSAP.

FORMULATION	MATERIAL MODEL	POSSIBILITY OF EQUILIBRIUM ITERATION
LINEAR ANALYSIS	Linear isotropic elastic	NO
MATERIALLY NONLINEAR ONLY ANALYSIS	Curve description model	NO

TABLE XI.1 POSSIBLE ANALYSIS TYPES WITH 3/D CONTINUUM ELEMENTS

XI. 3/D CONTINUUM ELEMENTS (continued)

2. Material Property Data

NPAR(16) sets of cards must be input in this section. Card "a" ("material number card") is the same for all material models, but card(s) "b" ("material property card(s)") depend on the material model number NPAR(15).

a. material number card (I5,F10.0)

note	columns	variable	entry
	1 - 5	N	Material property set number; GE.1 and LE. NPAR(16)
(1)	6 - 15	DEN(N)	Mass density of the material used in the calculations of the mass matrix; GE. 0.0

NOTES/

(1) The mass density defined is used directly in the calculation of the element mass matrix, i.e. no acceleration constants are applied to the variable DEN(N).

b. material property card(s) (8F10.0)

For MODEL "1" (NPAR(15).EQ.1) → [linear isotropic elastic]

note	columns	variable	entry
(1)	1 - 10	PROP(1,N)	Young's modulus, E
	11 - 20	PROP(2,N)	Poisson's ratio, ν

NOTES/

(1) MODEL 1 is a linear library model defined by two (2) positive constants (E,ν); i.e., if NPAR(15).EQ.1, NPAR(17) is set to "2" by default. MODEL 1 can be used with linear or nonlinear element groups. Since the material constants are independent of history NPAR(18) is "0" by default.

FOR MODEL "3" (NPAR(15).EQ.3) → [curve description model]

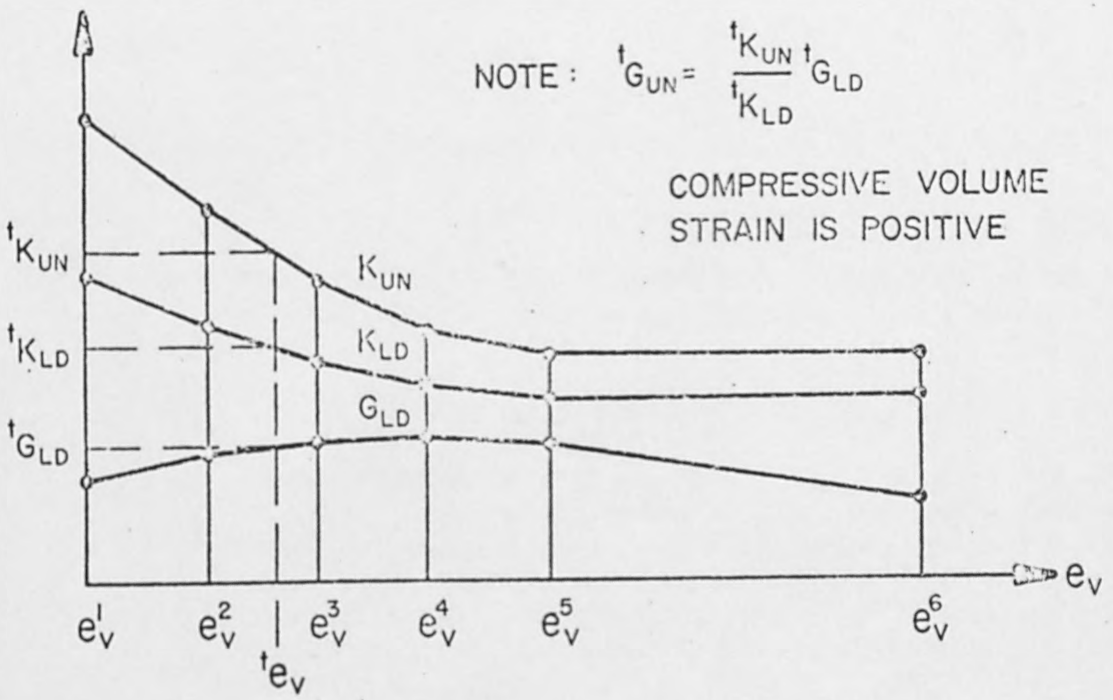
note	columns	variable	entry
first card			
(1)	1 - 10	PROP(1,N)	e_V^1 volume strain at point "1"
	11 - 20	PROP(2,N)	e_V^2....
	21 - 30	PROP(3,N)	e_V^3
	31 - 40	PROP(4,N)	e_V^4
	41 - 50	PROP(5,N)	e_V^5
	51 - 60	PROP(6,N)	e_V^6
	61 - 70	PROP(7,N)	K_{LD}^1 loading bulk modulus
	71 - 80	PROP(8,N)	at point "1" K_{LD}^2....
second card			
	1 - 10	PROP(9,N)	K_{LD}^3
	11 - 20	PROP(10,N)	K_{LD}^4
	21 - 30	PROP(11,N)	K_{LD}^5
	31 - 40	PROP(12,N)	K_{LD}^6
	41 - 50	PROP(13,N)	K_{UN}^1 unloading bulk modulus
	51 - 60	PROP(14,N)	at point "1" K_{UN}^2....
	61 - 70	PROP(15,N)	K_{UN}^3
	71 - 80	PROP(16,N)	K_{UN}^4
third card			
(2)	1 - 10	PROP(17,N)	K_{UN}^5
	11 - 20	PROP(18,N)	K_{UN}^6
	21 - 30	PROP(19,N)	G_{LD}^1 loading shear modulus
	31 - 40	PROP(20,N)	at point "1" G_{LD}^2....
	41 - 50	PROP(21,N)	G_{LD}^3
	51 - 60	PROP(22,N)	G_{LD}^4
	61 - 70	PROP(23,N)	G_{LD}^5
	71 - 80	PROP(24,N)	G_{LD}^6

NOTES/(1) MODEL 3 is a nonlinear library material model for which NPAR(17).EQ.24, by default. To characterize history, thirteen (13) variables--namely, the previously calculated

NOTES/(1) (continued)

stresses and strains, and the maximum volumetric strain ever reached in loading--need be saved for each integration point. Therefore, NPAR(18) is set to "13" by default.

All six input points must be defined, as indicated below:



Note that

$$e_V^J > e_V^{J-1}, \quad K_{UN}^J \geq K_{LD} \quad \text{and} \quad G_{LD}^J < 1.5\, K_{LD}^J.$$

where "J" is the J-th table input point. Linear interpolation is used to define the loading and unloading bulk moduli, ${}^tK_{LD}$, ${}^tK_{UN}$, and the loading shear modulus, ${}^tG_{LD}$, at time t between input points. If the volumetric strain during any time or load step exceeds e_V^6, a diagnostic message results.

(2) The unloading shear modulus at time t is calculated as ${}^tG_{UN} = ({}^tK_{UN}/\,{}^tK_{LD})\,{}^tG_{LD}$.

NOTATION

For the notation used in the description of the nonlinear element models refer to the report, "Static and Dynamic Geometric and Material Nonlinear Analysis," SESM Report No. 74-4 , Feb. 1974.

For MODEL "≥4" (NPAR(15).GE.4) → (user-supplied material model)

note	columns	variable	entry
(1)	1 - 10	PROP(1,N)	
	11 - 20	PROP(2,N)	
	21 - 30	PROP(3,N)	
	⋮	etc.	

NOTES/

(1) MODEL "≥4" is a non-standard model (i.e., is not in the material model library) and requires that property data be read

(PROP(J,N),J=1,NPAR(17),N=1,NPAR(16))

where NPAR(17) is user-supplied (i.e., no default value exists). Subroutines which allocate storage locations (in "WA") for element history (if the model is nonlinear) and which calculate the incremental stress-strain law and current stresses must be supplied. For a discussion on how to incorporate a user-supplied material model, see Appendix B.

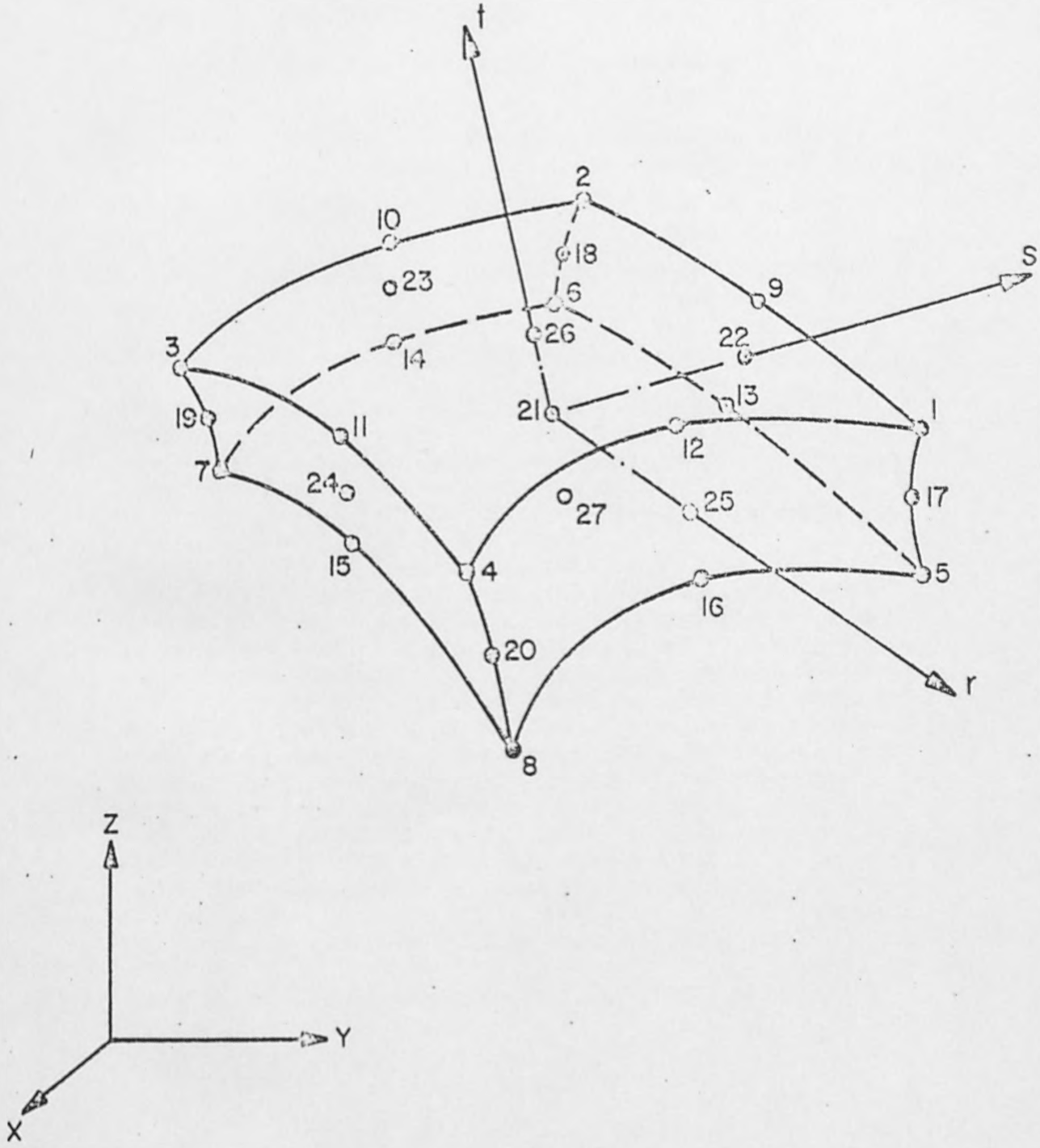


FIGURE XI.4 3/D CONTINUUM ELEMENT STRESS OUTPUT POINTS

XI. 3/D CONTINUUM ELEMENTS (continued)

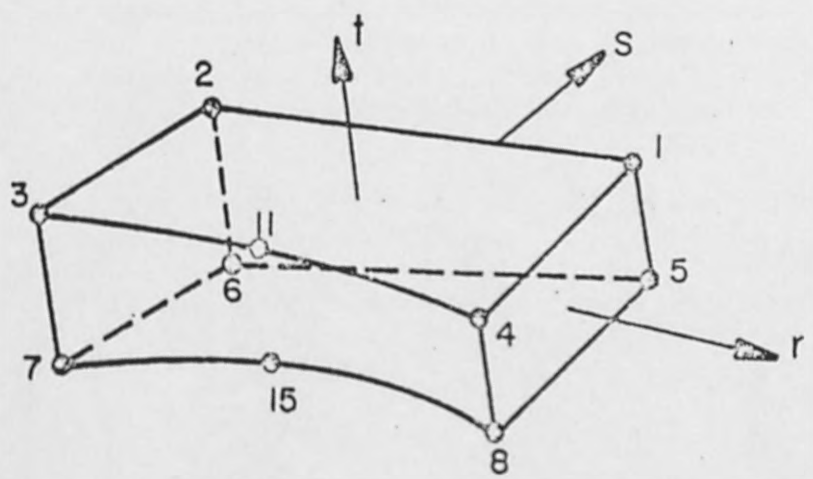
note	columns	variable	entry
	46 - 50	NOD(18)	Global node number of element nodal point 18
	51 - 55	NOD(19)	Global node number of element nodal point 19
	56 - 60	NOD(20)	Global node number of element nodal point 20
	61 - 65	NOD(21)	Global node number of element nodal point 21

NOTES/

(1) Element cards must be input in ascending element number order beginning with one (1) and ending with NPAR(2). Repetition of element numbers is illegal, but element cards may be omitted, and missing element data are generated according to the procedure described in note (6).

(2) IELD is a count of the node numbers actually posted on Cards 2 and 3 which must immediately follow Card 1. IELD must be at least eight (8), but must be less than or equal to the limit NPAR(7) which was given on the element group control card, Section XI.1. Element displacements are assigned at the IELD non-zero nodes, and thus the order of the element matrices is three (i.e., translations X,Y,Z) times IELD. The eight corner nodes of the hexahedron must be input, but nodes 9 to 21 are optional, and any or all of these optional nodes may be used to describe the element's displacement field. However, all 21 entries for NOD(I) are read from element data cards 2 and 3; if IELD .LT.21 the particular node locations not used in this element must be input as zero (0) in NOD(I). Figure XI.3 defines the input sequence that must be observed for element input.

For example, the 10-node element (IELD.EQ.10) shown below



NOTES/(2)

is defined by

NOD(I) = [X X X X X X X X 0 0 X 0 0 0 X 0 0 0 0 0 0],

where the nonzero entries (X) are the global mesh node numbers (Section III) of the 10 nodes.

(3) When element edges are straight it is unnecessary computationally to include side nodes in the numerical evaluation of coordinate derivatives, the Jacobian matrix, etc., and since regular element shapes are common, an option has been included to use fewer nodes in these geometric calculations than are used to describe element displacements. The first IELX nonzero nodes posted on Cards 2 and 3 are used to evaluate those parameters which pertain to element geometry only. IELX must be at least eight (8), and if omitted is re-set to IELD. A common application might be a 20 node element (i.e., IELD.EQ.20) with straight edges in which case IELX would be entered as "8".

(4) If IPS.EQ.0 no stress output will be provided for this element. A nonzero entry for the element stress printout flag "IPS" has a different meaning when used with nonlinear material models than it does when used with linear material models:

a) For a linear material model (MODEL.EQ.1) "IPS" specifies the stress output table (Section XI.3) to be used for this element. If no stress output tables were input (i.e., if NPAR(13).EQ.0) then IPS.EQ.1 specifies that stresses are to be printed at all integration points for this element.

b) For a nonlinear material model (MODEL.GE.3) IPS.EQ.1 specifies that stresses are to be printed at all integration points for this element.

(5) The flag IST allows the program to bypass stiffness and mass matrix calculations providing the current element is identical to the preceding element; i.e., the preceding and current elements are identical except for a rigid body translation. If IST.EQ.0, new matrices are computed for the current element.

(6) When element cards are omitted, element data are generated automatically as follows,

a) All data on Card 1 for generated elements is taken to be the same as that given on the first element card in the sequence;

NOTES/(6)

b) Nonzero node numbers (given on Cards 2 and 3 for the first element) are incremented by the value "KG" (on the first element's Card 1) as element generation progresses; zero (or blank) node number entries are generated as zeroes.

The last element cannot be generated.

XII. FREQUENCIES AND MODE SHAPES CALCULATION DATA

If the flag defining the eigensolution mode ("IEIG") given in Section II (Master Control Parameters, Card 3) is omitted, skip this block of data; i.e., if IEIG.EQ.0, no data is read in this section.

Frequency Solution Data Card (I5,F10.0,I5)

note	columns	variable	entry
(1)	1 - 5	NFREQ	Number of frequencies to be calculated; EQ.0, default set to "1"
(1)	6 - 15	COFQ	Cut-off circular frequency (units of radians per unit time); EQ.0, default set to "1.0E+08"
(2)	16 - 20	IFPR	Flag to suppress intermediate print-out of solution variables, interim informative messages, etc.; EQ.0, print-out suppressed EQ.1, print-out produced

NOTES/

(1) The lowest "NFREQ" vibration mode shapes and frequencies are evaluated for the structure linearized at time "TSTART". Frequencies are extracted in numerically ascending sequence, and unless the "N-th" frequency is > COFQ, the program will continue the eigensolution until all "NFREQ" modes have been determined. If the N-th frequency is > COFQ, where $1 \leq N \leq NFREQ$, the eigensolution is discontinued, and only the lowest "N" frequencies and mode shapes are printed.

At completion of solution, "physical" error bounds for each eigenvalue and eigenvector are calculated using $||(K-\omega^2 M)\phi||/||K\phi||$, where (ω,ϕ) is a calculated eigenpair.

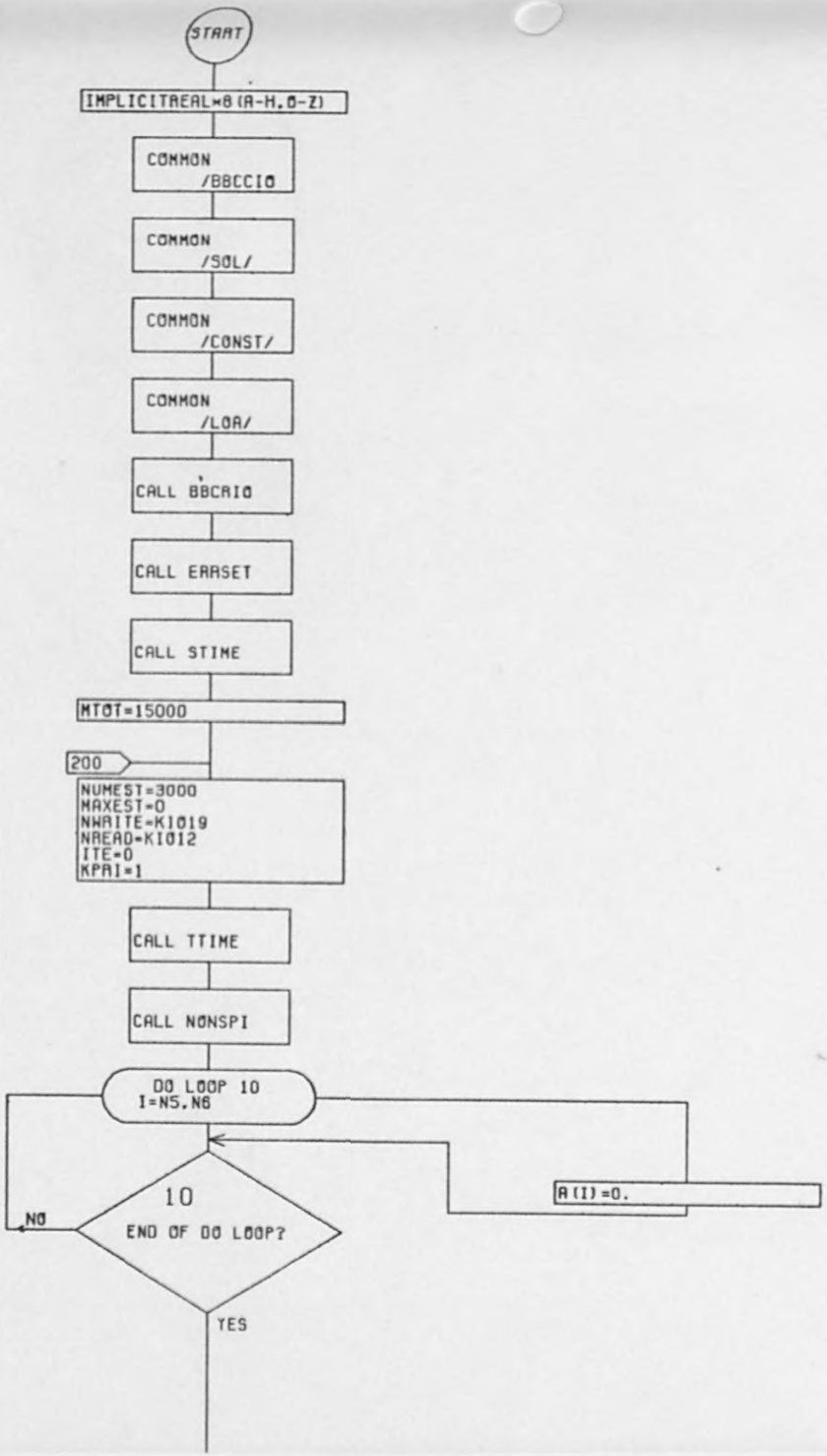
(2) During the course of eigenvalue extraction various solution variables, informative diagnostics, etc. are printed if IFPR.EQ.1. This print-out is useful in tracing solution behavior.

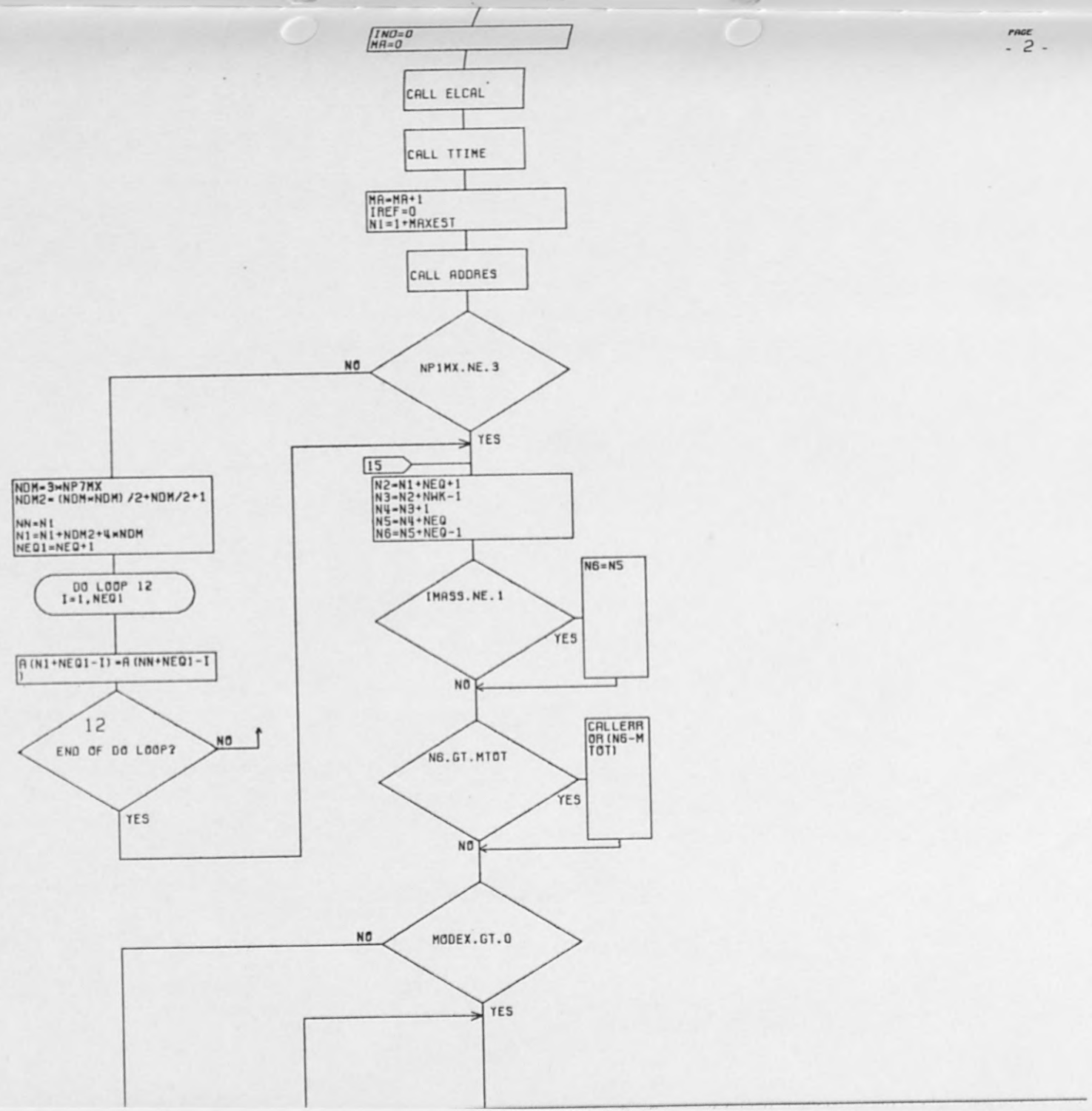
I. HEADING CARD (12A6)

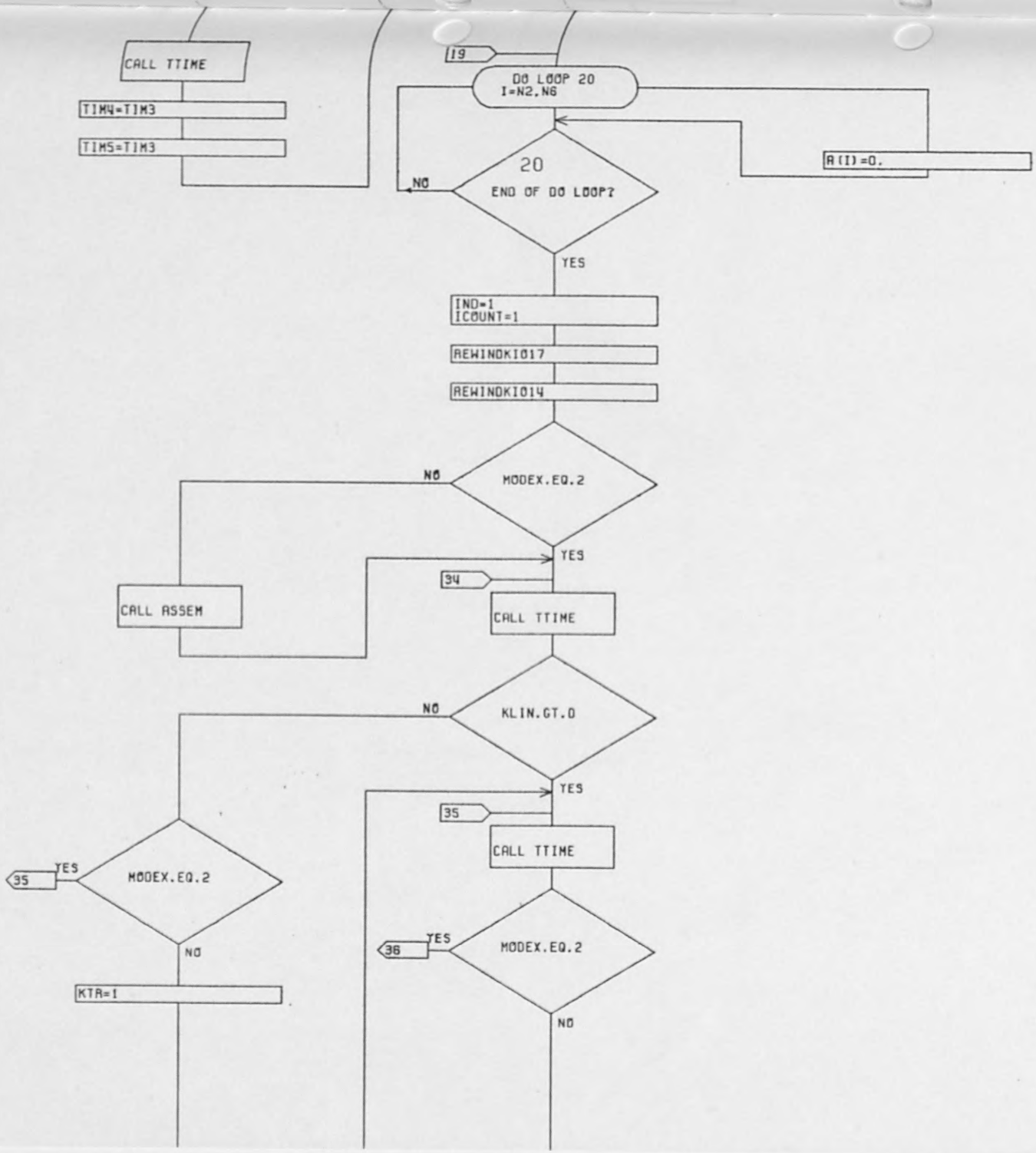
note	columns	variable	entry
(1)	1 - 72	HED(12)	Enter the master heading information for use in labeling the output

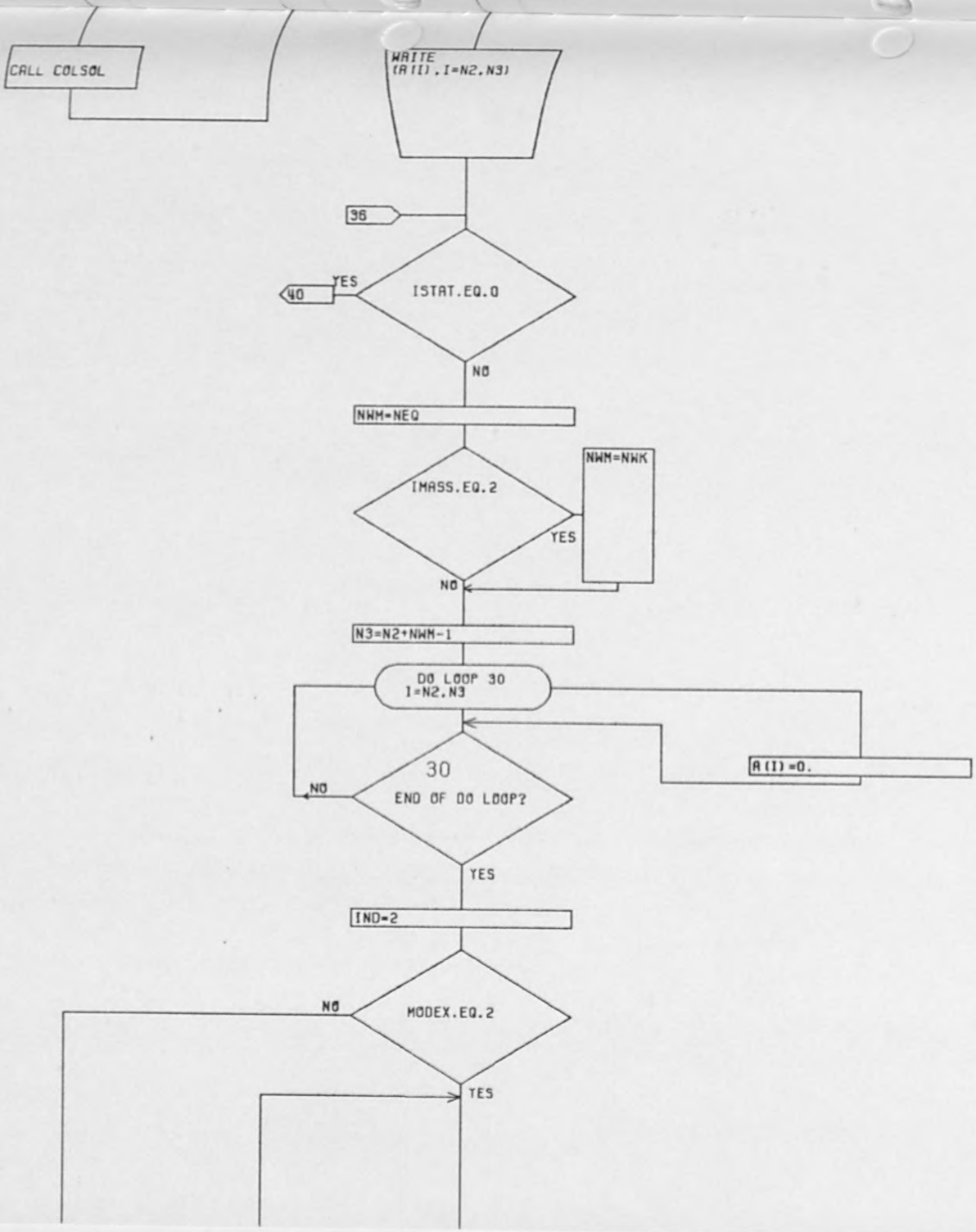
NOTES/

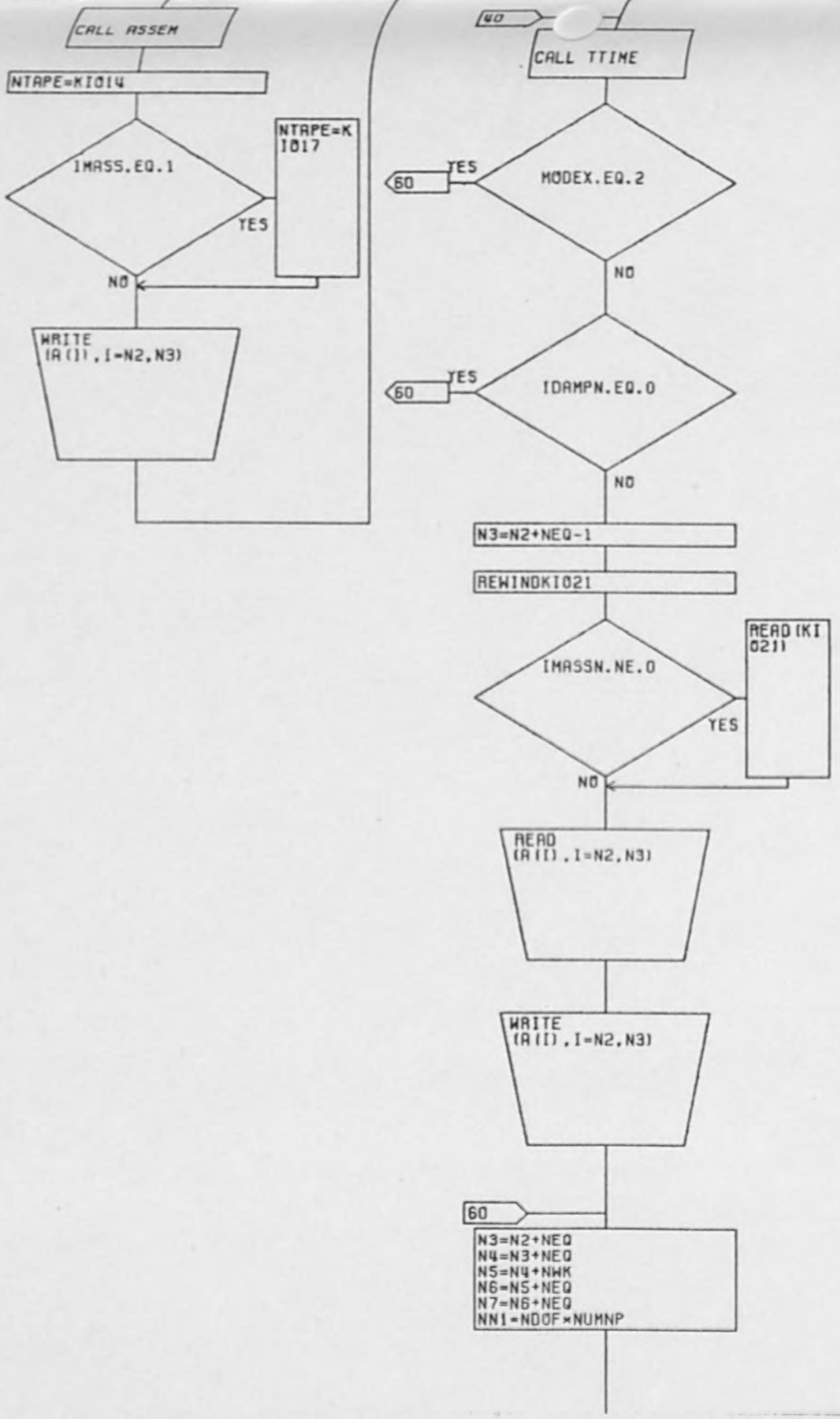
(1) Begin each new data case with a new heading card.

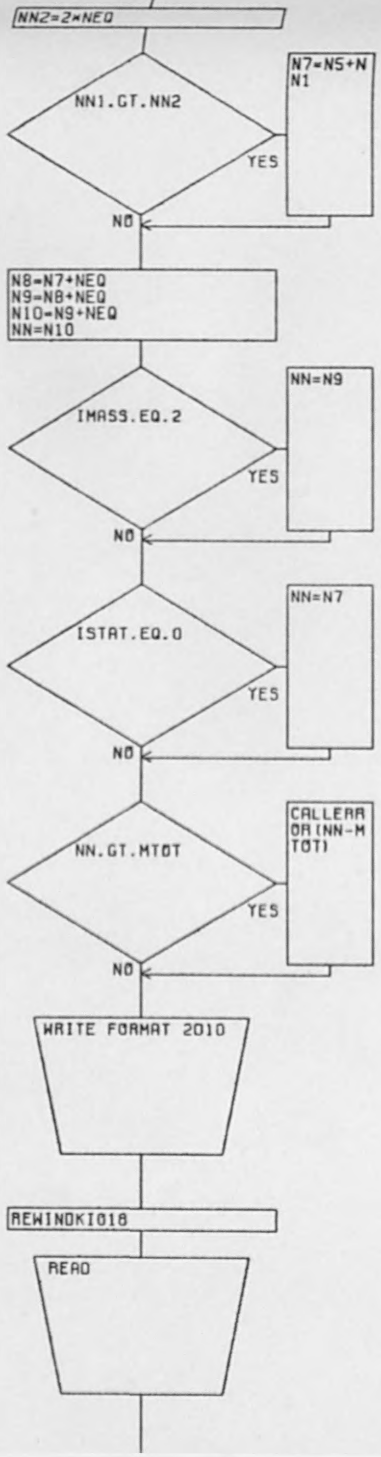

START
IMPLICITREAL*8(A-H,O-Z)
COMMON /BBCCIO
COMMON /SOL/
COMMON /CONST/
COMMON /LOA/
CALL BBCRIO
CALL ERRSET
CALL STIME
MTOT=15000
200
NUMEST=3000
MAXEST=0
NWRITE=KI019
NREAD=KI012
ITE=0
KPRI=1
CALL TTIME
CALL NONSPI
DO LOOP 10
I=N5,N6
10
END OF DO LOOP?
NO
YES
A(I)=0.

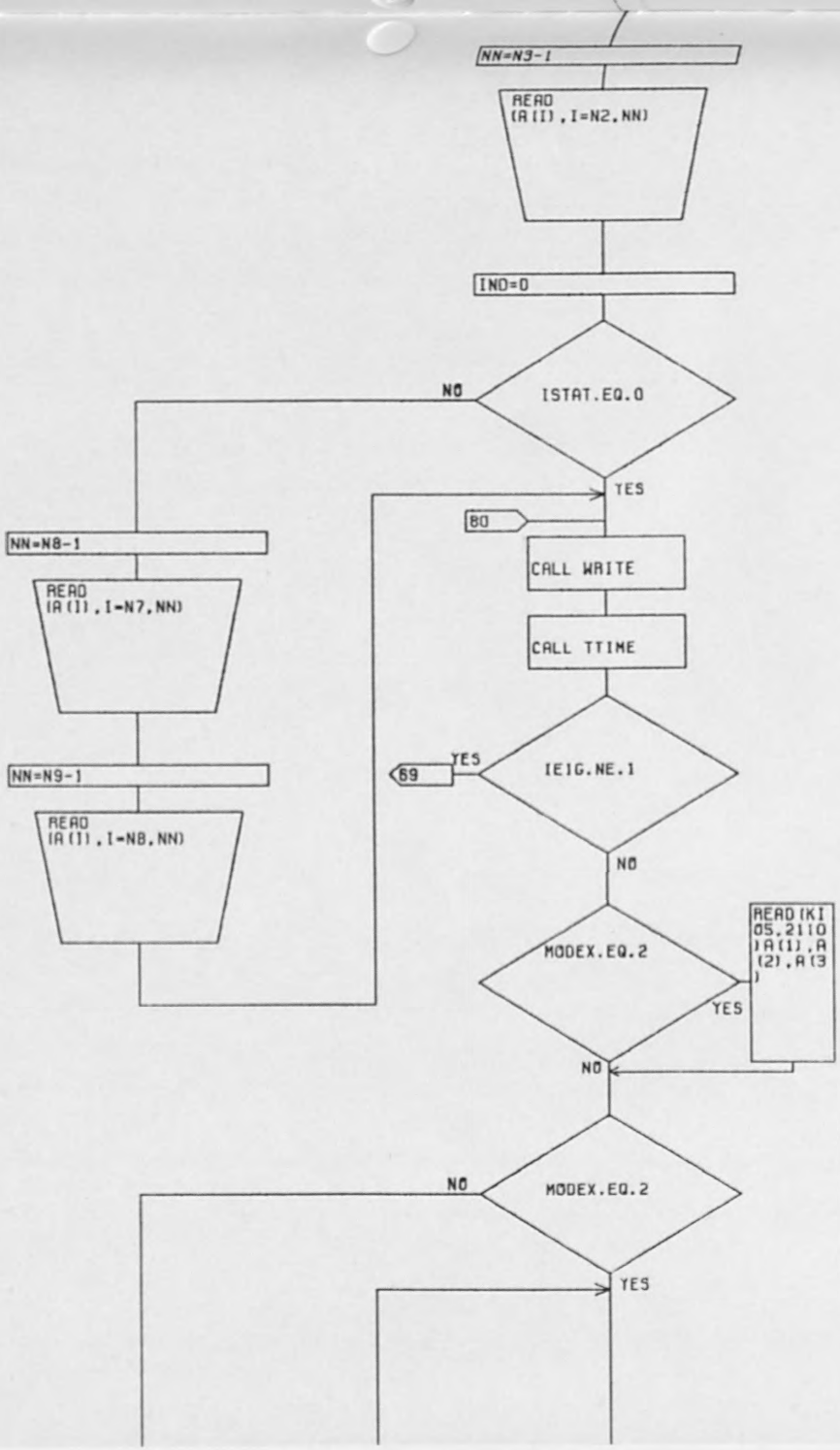

IND=0
MA=0
CALL ELCAL
CALL TTIME
MA=MA+1
IREF=0
N1=1+MAXEST
CALL ADDRES
NP1MX.NE.3
NO
YES
NDM=3*NP7MX
NDM2=(NDM*NDM)/2+NDM/2+1
NN=N1
N1=N1+NDM2+4*NDM
NEQ1=NEQ+1
DO LOOP 12
I=1,NEQ1
A(N1+NEQ1-I)=A(NN+NEQ1-I)
12
END OF DO LOOP?
NO
YES
15
N2=N1+NEQ+1
N3=N2+NWK-1
N4=N3+1
N5=N4+NEQ
N6=N5+NEQ-1
IMASS.NE.1
YES
N6=N5
NO
N6.GT.MTOT
YES
CALLERR
OR(N6-M
TOT)
NO
MODEX.GT.0
NO
YES

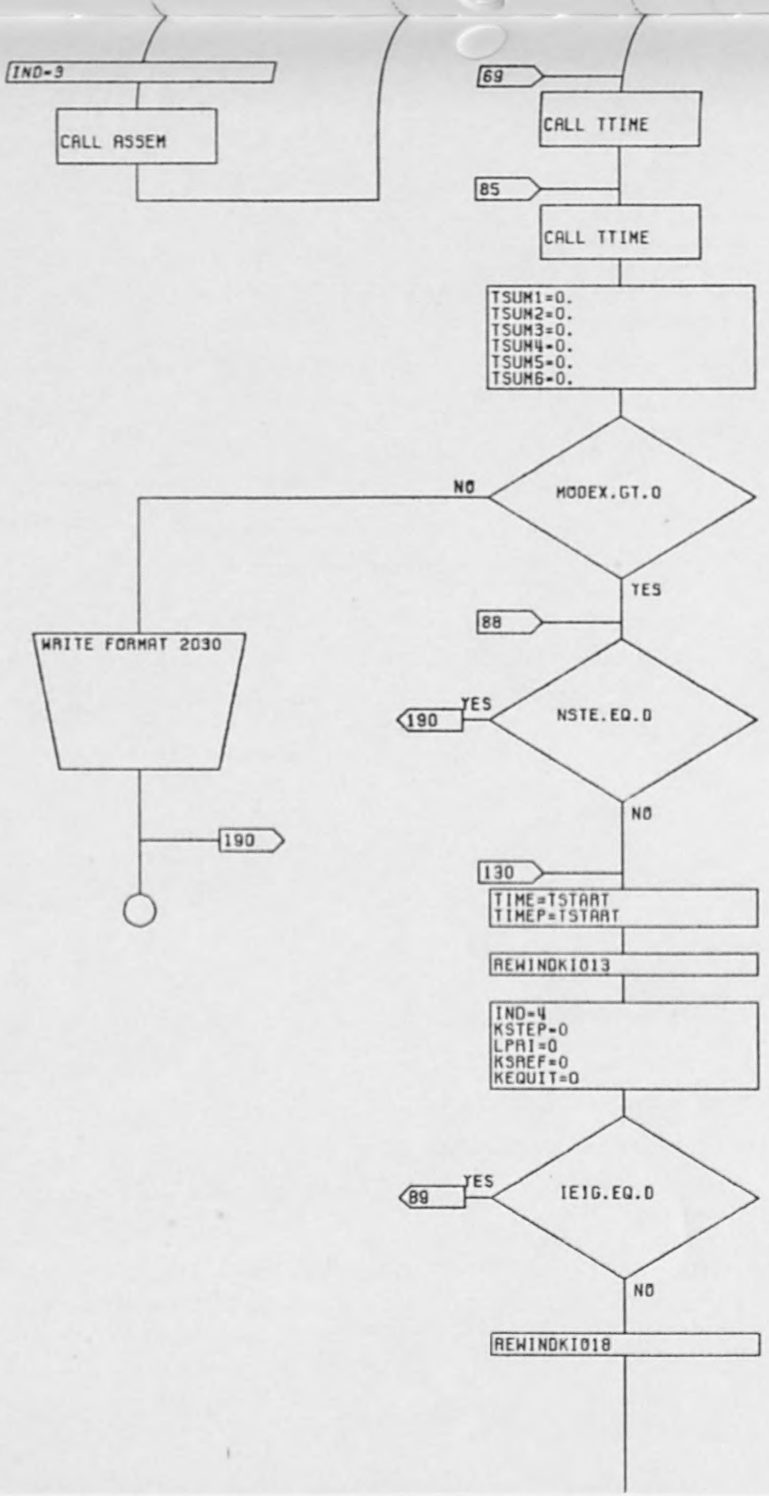

CALL TTIME
TIM4=TIM3
TIM5=TIM3
19
DO LOOP 20
I=N2,N6
A(I)=0.
20
END OF DO LOOP?
NO
YES
IND=1
ICOUNT=1
REWINDKI017
REWINDKI014
MODEX.EQ.2
NO
YES
34
CALL ASSEM
CALL TTIME
KLIN.GT.0
NO
YES
35
CALL TTIME
MODEX.EQ.2
YES
35
NO
KTA=1
MODEX.EQ.2
YES
36
NO

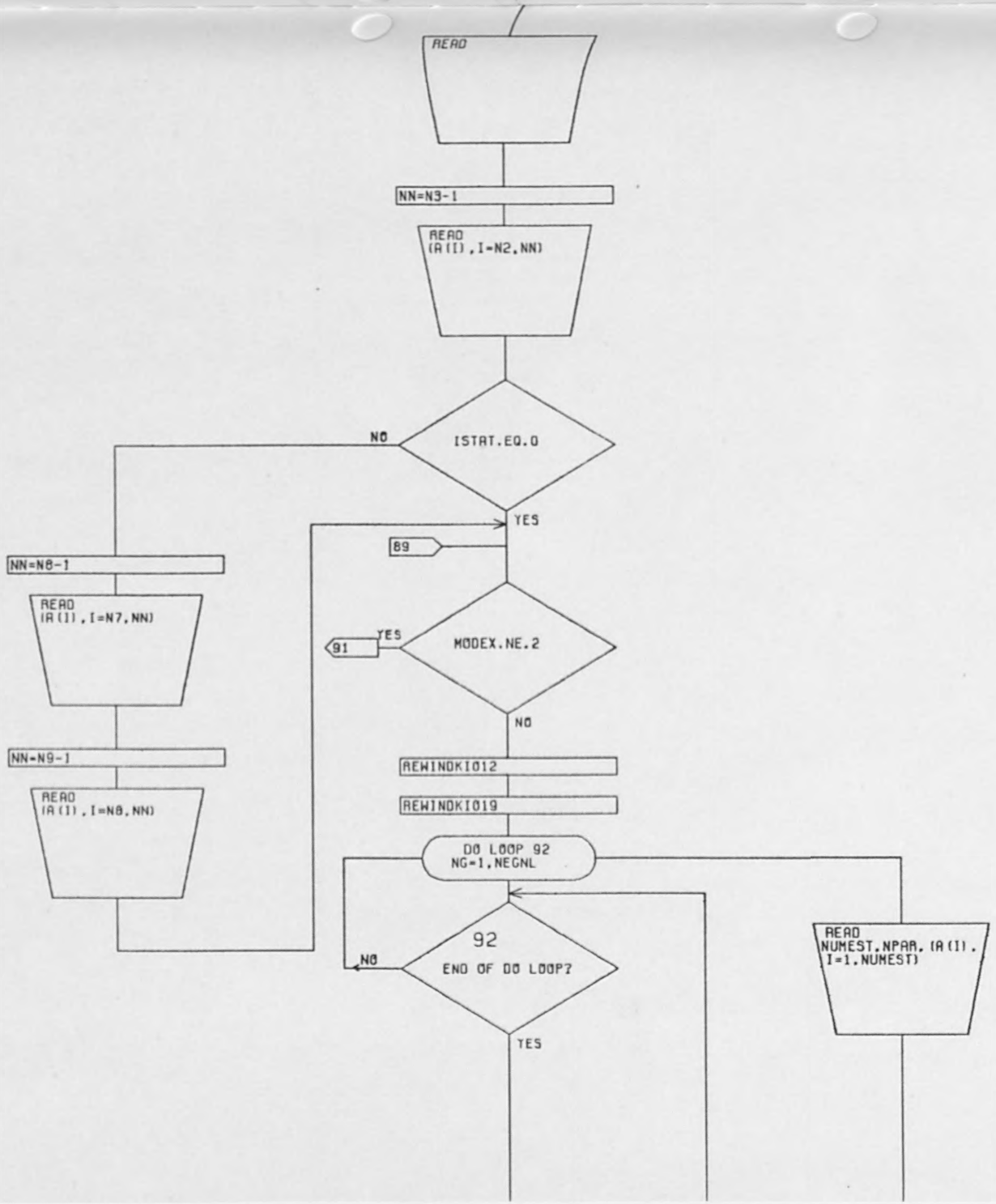

CALL COLSOL
WRITE
(A(I),I=N2,N3)
36
YES
40
ISTAT.EQ.0
NO
NWM=NEQ
IMASS.EQ.2
NWM=NWK
YES
NO
N3=N2+NWM-1
DO LOOP 30
I=N2,N3
A(I)=0.
30
END OF DO LOOP?
NO
YES
IND=2
NO
MODEX.EQ.2
YES

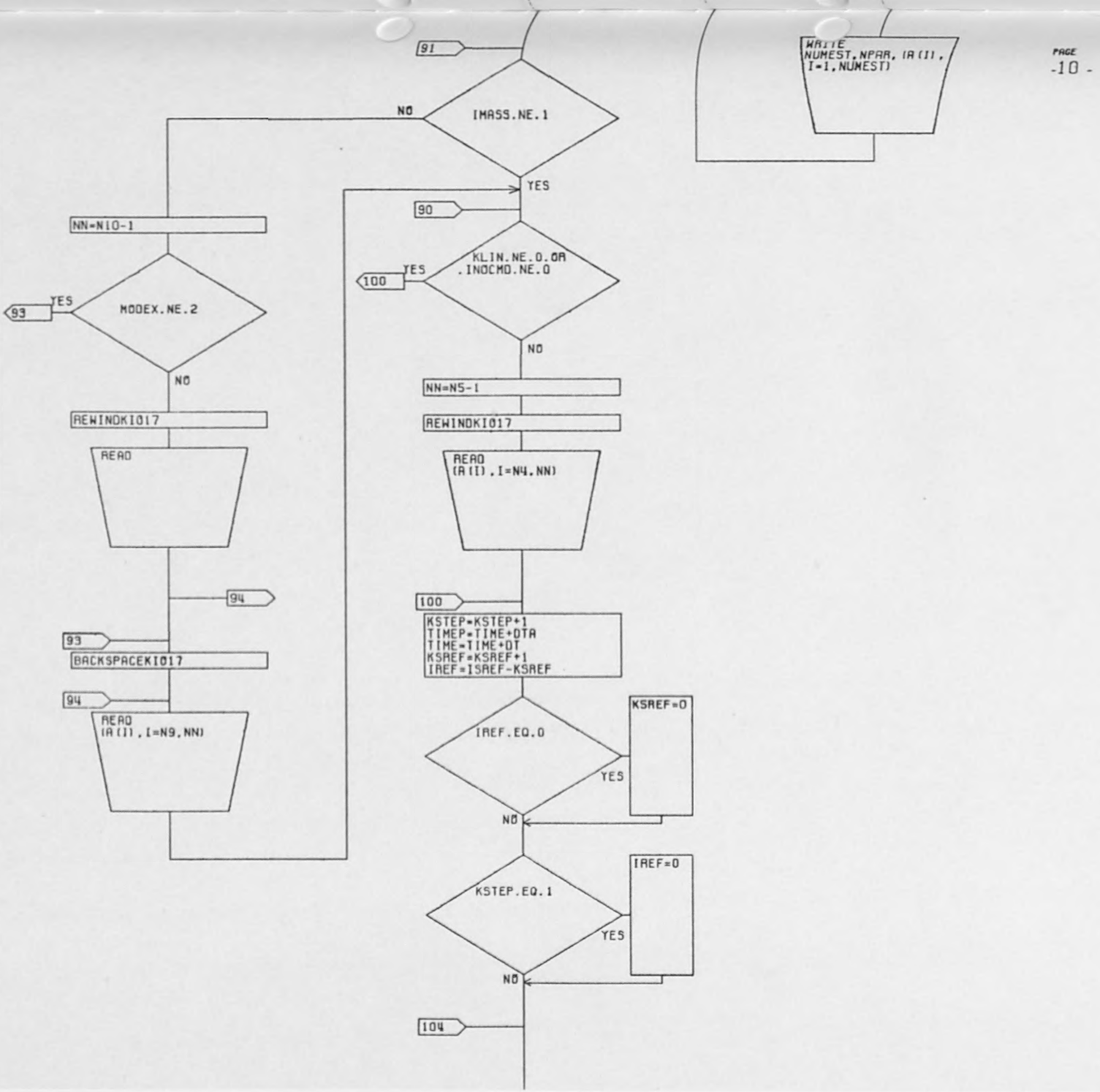

CALL ASSEM
NTAPE=KIO14
IMASS.EQ.1
YES
NTAPE=KIO17
NO
WRITE (A(I),I=N2,N3)
40
CALL TTIME
60
YES
MODEX.EQ.2
NO
60
YES
IDAMPN.EQ.0
NO
N3=N2+NEQ-1
REWINDKIO21
IMASSN.NE.0
YES
READ(KIO21)
NO
READ (A(I),I=N2,N3)
WRITE (A(I),I=N2,N3)
60
N3=N2+NEQ
N4=N3+NEQ
N5=N4+NHK
N6=N5+NEQ
N7=N6+NEQ
NN1=NDOF*NUMNP

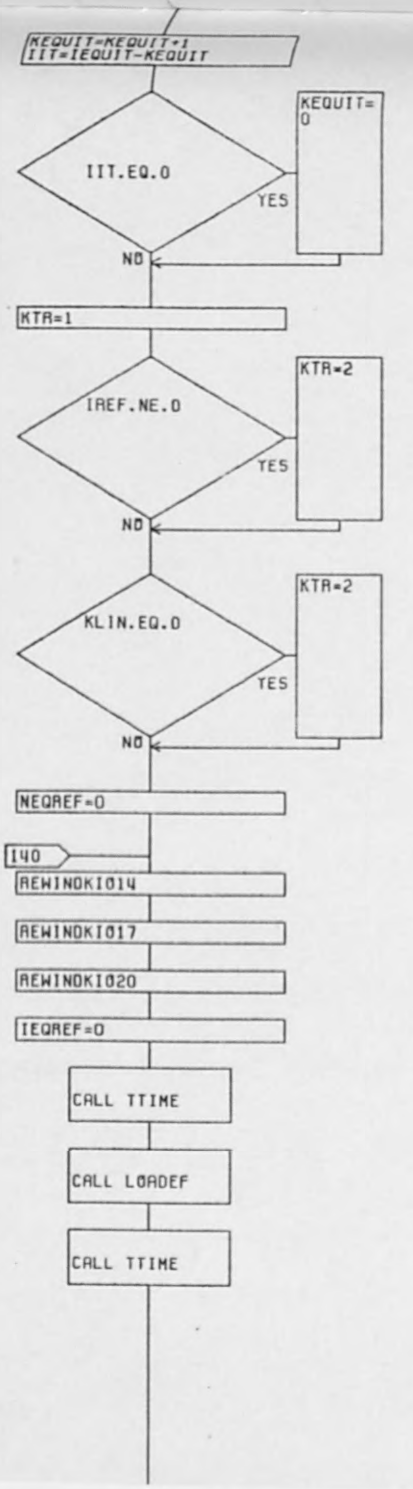

NN2=2*NEQ
NN1.GT.NN2
YES
N7=N5+N
N1
NO
N8=N7+NEQ
N9=N8+NEQ
N10=N9+NEQ
NN=N10
IMASS.EQ.2
YES
NN=N9
NO
ISTAT.EQ.0
YES
NN=N7
NO
NN.GT.MTOT
YES
CALLERR
OR(NN-M
TOT)
NO
WRITE FORMAT 2010
REWINDK1018
READ

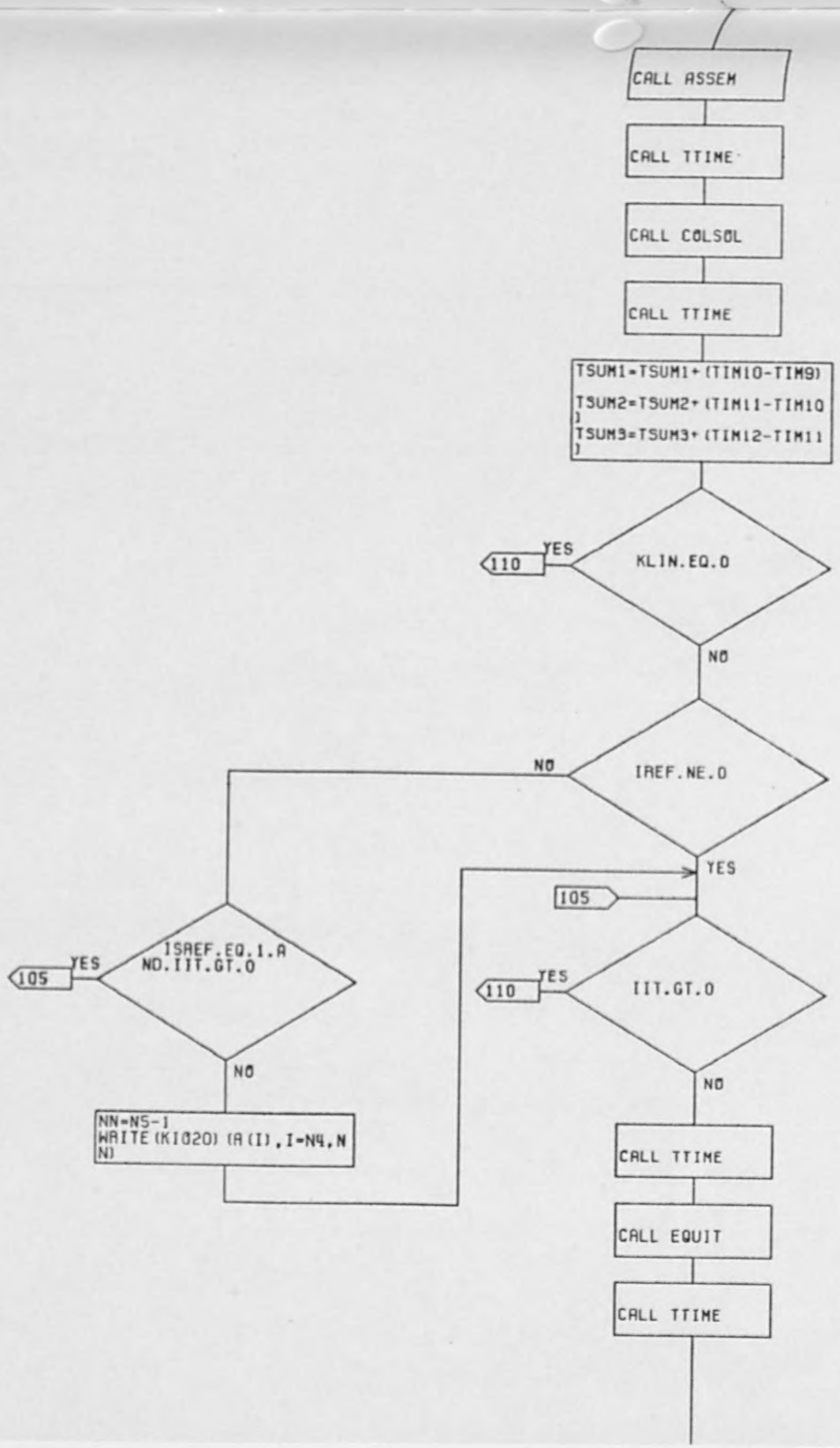

NN=N3-1
READ (A(I),I=N2,NN)
IND=0
ISTAT.EQ.0
NO
YES
80
CALL WRITE
CALL TTIME
IEIG.NE.1
YES
69
NO
MODEX.EQ.2
READ(KI05,2110)A(1),A(2),A(3)
YES
NO
NN=N8-1
READ (A(I),I=N7,NN)
NN=N9-1
READ (A(I),I=N8,NN)
MODEX.EQ.2
NO
YES

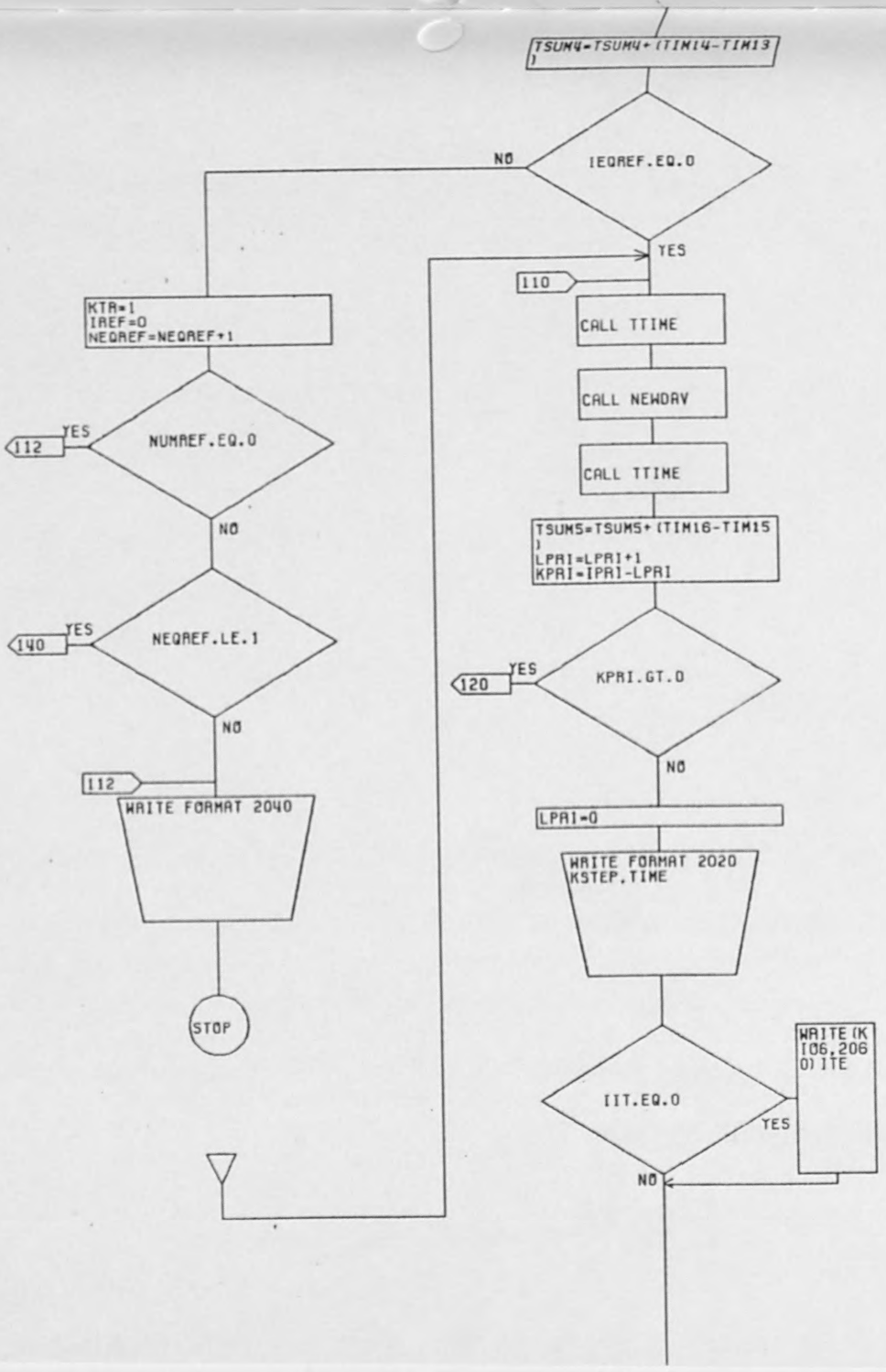

IND=3
CALL ASSEM
69
CALL TTIME
85
CALL TTIME
TSUM1=0.
TSUM2=0.
TSUM3=0.
TSUM4=0.
TSUM5=0.
TSUM6=0.
MODEX.GT.0
NO
YES
WRITE FORMAT 2030
88
190
YES
NSTE.EQ.0
NO
190
130
TIME=TSTART
TIMEP=TSTART
REWINDKI013
IND=4
KSTEP=0
LPRI=0
KSREF=0
KEQUIT=0
89
YES
IEIG.EQ.0
NO
REWINDKI018

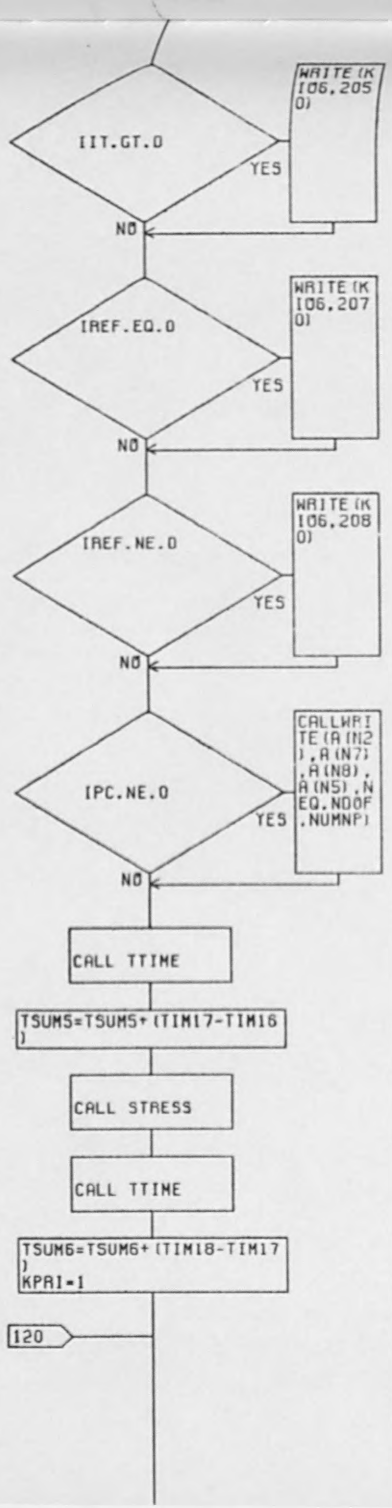

READ
NN=N3-1
READ
(A(I),I=N2,NN)
NO
ISTAT.EQ.0
YES
89
NN=N8-1
READ
(A(I),I=N7,NN)
91
YES
MODEX.NE.2
NO
NN=N9-1
REWINDKI012
READ
(A(I),I=N8,NN)
REWINDKI019
DO LOOP 92
NG=1,NEGNL
92
NO
END OF DO LOOP?
YES
READ
NUMEST,NPAR,(A(I),
I=1,NUMEST)

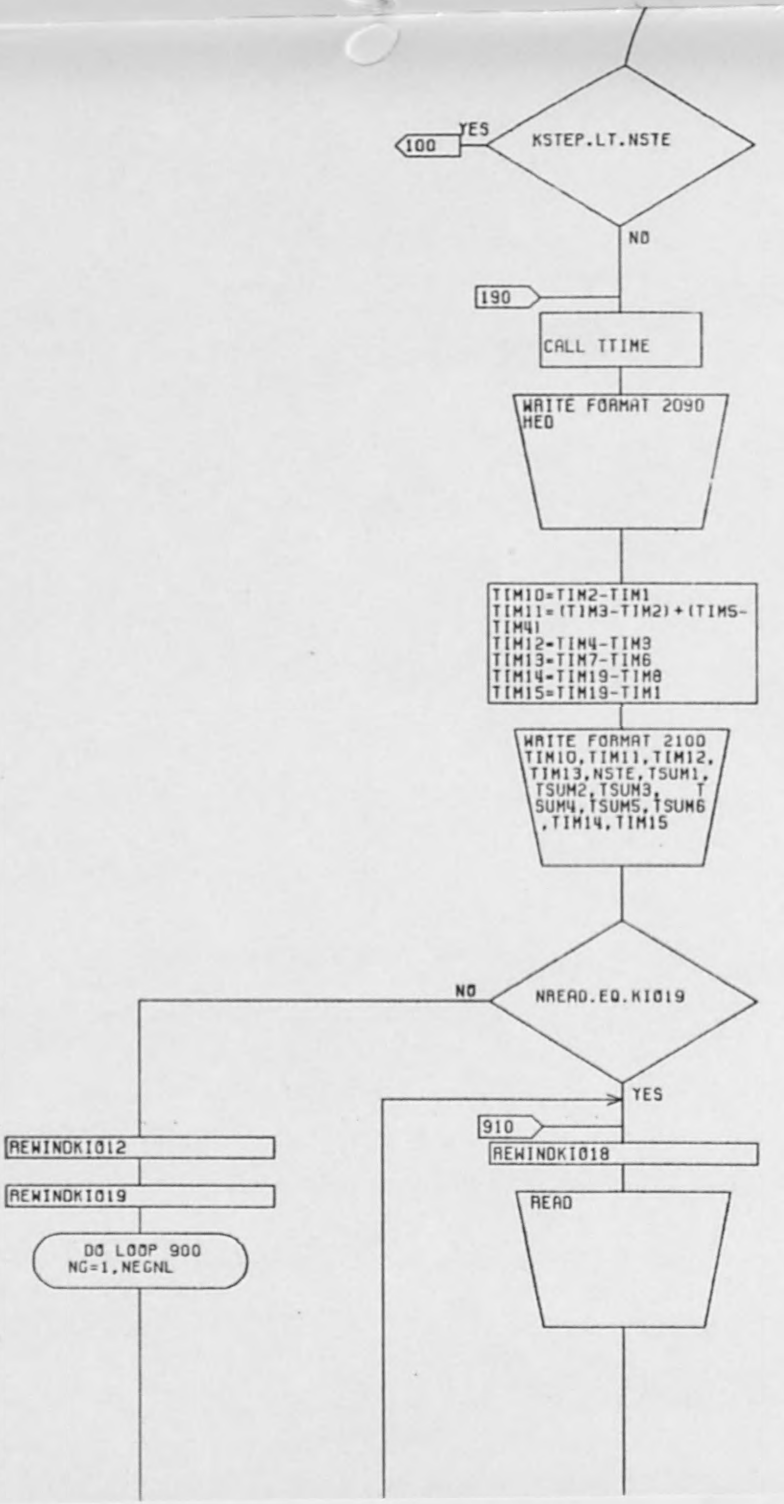

91
IMASS.NE.1
NO
YES
90
KLIN.NE.0.OR
.INOCMD.NE.0
100
YES
NO
NN=N10-1
MODEX.NE.2
93
YES
NO
REWINDK1017
READ
94
93
BACKSPACEK1017
94
READ
(A(I),I=N9,NN)
NN=N5-1
REWINDK1017
READ
(A(I),I=N4,NN)
100
KSTEP=KSTEP+1
TIMEP=TIME+DTA
TIME=TIME+DT
KSREF=KSREF+1
IREF=ISREF-KSREF
IREF.EQ.0
YES
KSREF=0
NO
KSTEP.EQ.1
YES
IREF=0
NO
104
WRITE
NUMEST,NPAR,(A(I),
I=1,NUMEST)

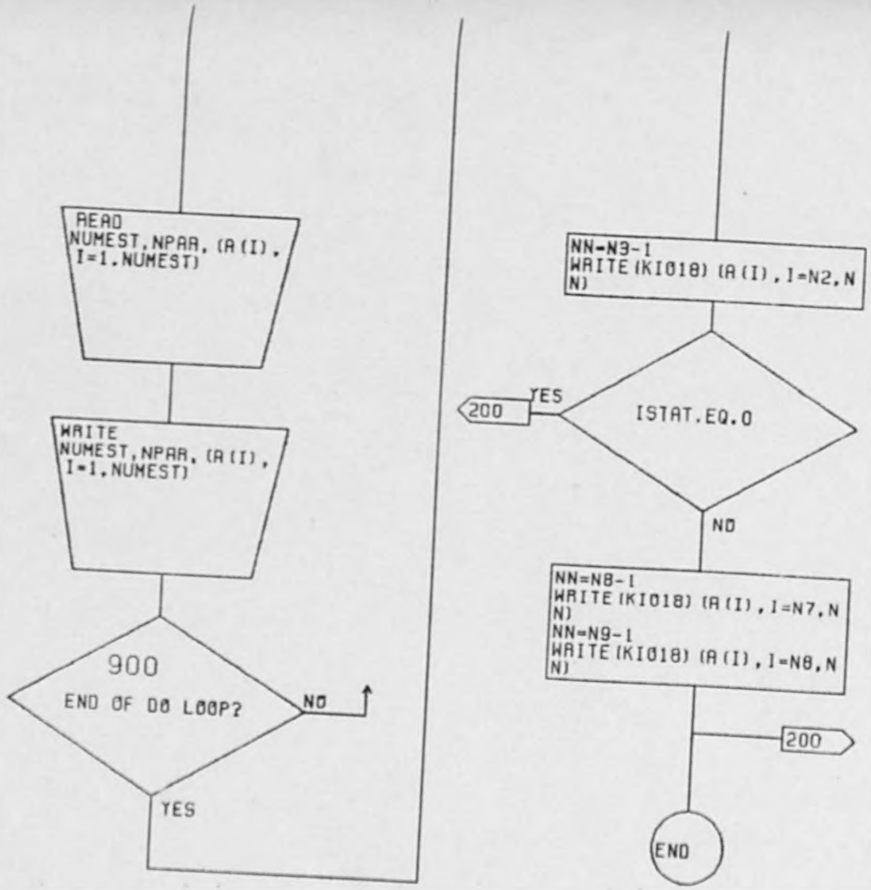

KEQUIT=KEQUIT+1
IIT=IEQUIT-KEQUIT
IIT.EQ.0
YES
KEQUIT=0
NO
KTR=1
IREF.NE.0
YES
KTR=2
NO
KLIN.EQ.0
YES
KTR=2
NO
NEQREF=0
I40
REWINDKI014
REWINDKI017
REWINDKI020
IEQREF=0
CALL TTIME
CALL LOADEF
CALL TTIME


CALL ASSEM
CALL TTIME
CALL COLSOL
CALL TTIME
TSUM1=TSUM1+(TIM10-TIM9)
TSUM2=TSUM2+(TIM11-TIM10)
TSUM3=TSUM3+(TIM12-TIM11)
110
YES
KLIN.EQ.0
NO
IREF.NE.0
NO
YES
105
ISREF.EQ.1.AND.IIT.GT.0
YES
105
NO
NN=NS-1
WRITE(K1020)(A(I),I=N4,NN)
110
YES
IIT.GT.0
NO
CALL TTIME
CALL EQUIT
CALL TTIME


TSUM4=TSUM4+(TIM14-TIM13)
IEQREF.EQ.0
NO
YES
110
CALL TTIME
CALL NEWDAY
CALL TTIME
TSUM5=TSUM5+(TIM16-TIM15)
LPRI=LPRI+1
KPRI=IPRI-LPRI
KPRI.GT.0
120
YES
NO
LPRI=0
WRITE FORMAT 2020
KSTEP,TIME
IIT.EQ.0
YES
NO
WRITE(K
106,206
0) ITE
KTR=1
IREF=0
NEQREF=NEQREF+1
NUMREF.EQ.0
112
YES
NO
NEQREF.LE.1
140
YES
NO
112
WRITE FORMAT 2040
STOP


IIT.GT.0
YES
WRITE(K
106,205
0)
NO
IREF.EQ.0
YES
WRITE(K
106,207
0)
NO
IREF.NE.0
YES
WRITE(K
106,208
0)
NO
IPC.NE.0
YES
CALLWRI
TE(A(N2
),A(N7)
,A(N8),
A(N5),N
EQ.NDOF
,NUMNP)
NO
CALL TTIME
TSUM5=TSUM5+(TIM17-TIM16
)
CALL STRESS
CALL TTIME
TSUM6=TSUM6+(TIM18-TIM17
)
KPRI=1
120


100
YES
KSTEP.LT.NSTE
NO
190
CALL TTIME
WRITE FORMAT 2090
HED
TIM10=TIM2-TIM1
TIM11=(TIM3-TIM2)+(TIM5-TIM4)
TIM12=TIM4-TIM3
TIM13=TIM7-TIM6
TIM14=TIM19-TIM8
TIM15=TIM19-TIM1
WRITE FORMAT 2100
TIM10,TIM11,TIM12,
TIM13,NSTE,TSUM1,
TSUM2,TSUM3, T
SUM4,TSUM5,TSUM6
,TIM14,TIM15
NO
NREAD.EQ.KI019
YES
910
REWINDKI012
REWINDKI019
DO LOOP 900
NG=1,NEGNL
REWINDKI018
READ


READ
NUMEST,NPAR,(A(I),
I=1,NUMEST)
WRITE
NUMEST,NPAR,(A(I),
I=1,NUMEST)
900
END OF DO LOOP?
NO
YES
NN=N3-1
WRITE(KI018)(A(I),I=N2,N
N)
200
YES
ISTAT.EQ.0
NO
NN=N8-1
WRITE(KI018)(A(I),I=N7,N
N)
NN=N9-1
WRITE(KI018)(A(I),I=N8,N
N)
200
END

```
TTTTTTTTTTTTT  TTTTTTTTTTTTT  DDDDDDDDDDD    AAAAAAAAAA       11        00000000    SSSSSSSSSS   MM         MM
TTTTTTTTTTTTT  TTTTTTTTTTTTT  DDDDDDDDDDDD  AAAAAAAAAAAA     111       0000000000   SSSSSSSSSSSS  MMMM       MMMM
     TT             TT       DD         DD  AA         AA   1111      00        00  SS          SS  MMMMM     MMMMM
     TT             TT       DD         DD  AA         AA     11      00        00  SS              MM MM   MM MM
     TT             TT       DD         DD  AA         AA     11      00        00  SSS             MM  MMMM   MM
     TT             TT       DD         DD  AAAAAAAAAAAA     11      00        00   SSSSSSSSS      MM   MM    MM
     TT             TT       DD         DD  AAAAAAAAAAAA     11      00        00    SSSSSSSSS    MM         MM
     TT             TT       DD         DD  AA         AA     11      00        00          SSS    MM         MM
     TT             TT       DD         DD  AA         AA     11      00        00           SS    MM         MM
     TT             TT       DD         DD  AA         AA     11      00        00  SS       SS    MM         MM
     TT             TT       DDDDDDDDDDDD  AA         AA  1111111111   0000000000   SSSSSSSSSSS   MM         MM
     TT             TT       DDDDDDDDDDD   AA         AA  1111111111    00000000     SSSSSSSSS    MM         MM


          JJ  000000000000  BBBBBBBBBB          2222222222  555555555555  777777777777   9999999999
          JJ  000000000000  BBBBBBBBBBB        2222222222  555555555555  77777777777   99999999999
          JJ  00        00  BB        BB      22        22  55            77      77    99         99
          JJ  00        00  BB        BB              22  55            77      77    99         99
          JJ  00        00  BB        BB              22  55            77      77    99         99
          JJ  00        00  BBBBBBBBBB              22    55555555555       77       999999999999
          JJ  00        00  BBBBBBBBBB            22      555555555555      77        99999999999
          JJ  00        00  BB        BB        22                55      77                  99
JJ        JJ  00        00  BB        BB      22                  55      77                  99
JJ        JJ  00        00  BB        BB    22          55        55      77        99        99
JJJJJJJJJJJ   000000000000  BBBBBBBBBBB     2222222222  555555555555      77        999999999999
 JJJJJJJJJ    000000000000  BBBBBBBBBB      2222222222   5555555555       77         9999999999
```

BBC BADEN TTDA10SM (2579) DATE: 82.237/82-08-25 TIME: 14.37.44 ABT: TDA NAME: MUKHERJEE

PROJECT: TSO
LIBRARY: TDA10
TYPE: FH2079

DATE: 82/08/25
TIME: 14:23
PAGE: 001

GENERAL DATA:
VOLUME SERIAL: 501004
DEVICE TYPE: 3350
ORGANIZATION: PO
CREATION DATE: 82/08/18

GENERAL DATA:
RECORD FORMAT: FB
RECORD LENGTH: 80
BLOCK SIZE: 6,160
1ST EXTENT SIZE: 65
SECONDARY QUAN: 0

CURRENT ALLOCATION:
65 TRACKS
1 EXTENT
30 DIRECTORY BLOCKS

CURRENT UTILIZATION:
63 TRACKS
1 EXTENT
8 DIRECTORY BLOCKS
90 MEMBERS

MEMBER NAME	VERS.MOD LEVEL	CREATION DATE	DATE AND TIME LAST MODIFIED	CURRENT NO. LINES	INITIAL NO. LINES	MODIFIED NO. LINES	USER ID
AAMAIN	01.01	82/08/24	82/08/25 09:34	575	575	1	TDA10
ADDBAN							
ADDCM	01.02	82/08/18	82/08/19 13:46	59	60	2	TDA10
ADDDI							
ADDMA							
ADDRES							
ASSEM							
BANDET							
CAUCHY							
CDMOD	01.03	82/08/18	82/08/20 13:57	334	340	4	TDA10
CMOD3D							
COLHT							
COLSOL							
DCRACK							
DERIQ							
DERIQ3							
DRUCK							
ELCAL							
ELEMNT							
ELPAL							
ELTTT6							
ELT2D3							
ELT2D4							
ELT2D6							
ELT2D7							
ELT2D8							
ELT2D9							
ELT3D3							
ELT3D4							
ELT3D5							
ELT3D6							
EL2D10							
EL2D11							

MEMBER NAME	VERS.MOD LEVEL	CREATION DATE	DATE AND TIME LAST MODIFIED	CURRENT NO. LINES	INITIAL NO. LINES	MODIFIED NO. LINES	USER ID
EL2D12							
EQUIT							
ERROR							
FREQS	01.02	82/08/18	82/08/18 17:51	130	134	1	TDA10
FUNCT							
FUNCT2							
ICDMOD	01.00	82/08/19	82/08/19 14:01	113	113	0	TDA10
ICMOD3							
IDRUCK							
IELPAL							
INITAL							
INITWA							
INPUT							
INTWA3							
IVTMOD							
LOADEF							
LOADS							
MATBAR							
MATRT2							
MATWRT							
MAXAEX	01.01	82/08/18	82/08/18 17:46	10	10	1	TDA10
MAXMIN							
MIDEP							
MLTPLY							
MOONEY							
MULT							
NEWDAV							
NODMAS							
NONSPI							
OPCOEF							
PNORM							
POSINV							
PRAGER							
QUADM							
QUADM3							
QUADS							
QUADS3							
READID	01.01	82/08/18	82/08/18 17:48	15	15	1	TDA10
RUSS							
SECANT	01.00	82/08/19	82/08/19 14:25	612	612	0	TDA10
STIME							

PROJECT: TSO
LIBRARY: TDA10
TYPE: FH2079

DATE: 82/08/25
TIME: 14:23
PAGE: 003

MEMBER NAME	VERS.MOD LEVEL	CREATION DATE	DATE AND TIME LAST MODIFIED	CURRENT NO. LINES	INITIAL NO. LINES	MODIFIED NO. LINES	USER ID
STRESS							
STSTL							
STSTN							
STST3L							
STST3N							
TDFE	01.00	82/08/19	82/08/19 14:43	532	532	0	TDA10
THDFE	01.01	82/08/19	82/08/19 15:13	540	539	3	TDA10
THREDM							
TODMFE							
TRUSS							
TTIME							
VTMOD							
WRITE							
WRTFRQ							
XYZ	01.00	82/08/19	82/08/19 16:39	35	35	0	TDA10
ZZ	01.00	82/08/19	82/08/19 16:37	7	7	0	TDA10
MAXIMUMS:	01.03	82/08/24	82/08/25 09:34	612	612	4	
TOTALS:				2,962	2,972	13	

END OF MEMBER LIST

```
START                                                                                MOD
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8  FLAGS
   1  C...PROGRAM-IDENTIFIKATION NUMBER     H2079                                    00000100
   1  C   TITEL  NONSAP                                                              00000200
   1  C                                                                             00000300
   1  C  IMPLEMENTED AND MODIFIED IN I B M 370 BY S N MUKHERJEE. 16.12.77.          00000400
   1  C   PROGRAMER,AUTHOR  MUKHERJEE  TKS                                         00000501   **
   1  C   PROGRAMED OR LAST CORRECTION   1.10.78                                   00000600
   1  C   SUBROUTINES AND FUNCTIONS OF COMMON-LIBRARIES     -                       00000700
   1  C                                                                             00000800
   1  C   INPUT/OUTPUT DEFINITIONS                                                 00000900
   1  C                          CARD -INPUT -REFERENCE     5                       00001000
   1  C                          PRINT-OUTPUT-REF.          3                       00001100
   1  C                          PUNCH-OUTPUT-REF.                                  00001200
   1  C                          WORKFILE(S)       11,12,13,14,19                   00001300
   1  C                          OTHER I/O-UNITS USED                               00001400
   1  C   SIZE PROG.LENGTH WITHOUT INPUT/OUTPUT BUFFER IN KILOBYTES   160          00001500
   1  C   COMPILER/CODE                EBCDIC                                      00001600
   1  C   IMPLEMENTED ON COMPUTERMODEL    IBM 370/168                             00001700
   1  C   PROG. STRUCTURE (IF OVERLAY)    KEINE                                   00001800
   1  C                                N O N S A P                                 00001900
   1  C                                                                            00002000
   1  C   **   **   **   **   **   **   **   **   **   **   **   **   **   **     00002100
   1  C                     A STRUCTURAL ANALYSIS PROGRAM FOR                      00002200
   1  C               STATIC AND DYNAMIC RESPONSE OF NONLINEAR SYSTEMS             00002300
   1  C                                                                            00002400
   1  C                  K. J. BATHE,  E. L. WILSON,  R. H. IDING                  00002500
   1  C                                                                            00002600
   1  C                    UNIVERSITY OF CALIFORNIA,  BERKELEY                     00002700
   1  C                                                                            00002800
   1  C                                                                            00002900
   1  C   **   **   **   **   **   **   **   **   **   **   **   **   **   **     00003000
   1  C                                                                            00003100
   1  C                                                                            00003200
   1  C                                                                            00003300
   1  C . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 00003400
   1  C .                                                                       .  00003500
   1  C .  T A P E  A L L O C A T I O N                                         .  00003600
   1  C .                                                                       .  00003700
   1  C .      . TAPE11  STORES LINEAR ELEMEST GROUP DATA                       .  00003800
   1  C .                ( WHEN NEGL.NE.0 ONLY )                                .  00003900
   1  C .                                                                       .  00004000
   1  C .      . TAPE12, TAPE19  STORE NONLINEAR ELEMENT GROUP DATA             .  00004100
   1  C .                                                                       .  00004200
   1  C .      . TAPE13  STORES EXTERNALLY APPLIED L6AD2                           00004300
   1  C .                                                                       .  00004400
   1  C .      . TAPE14  STORES SEQUENTIALLY                                       00004500
   1  C .        (1) LINEAR STIFFNESS MATRIX (IN NONLINEAR ANALYSIS              .  00004600
   1  C .            AND/OR WHEN RAYLEIGH DAMPING IS SPECIFIED)                  .  00004700
   1  C .            (IN CASE NEGL.NE.0 ONLY)                                    .  00004800
   1  C .        (2) MASS MATRIX (IN CASE IMASS.EQ.2 ONLY )                      .  00004900
```

```
START                                                                    MOD
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8  FLAGS

   1  C .         (3) CONCENTRATED NODAL DAMPING VECTOR                  . 00005000
   1  C .               THE MASS AND/OR DAMPING MATRICES ARE STORED ONLY  . 00005100
   1  C .               WHEN THERE ARE MASS AND/OR DAMPING EFFECTS.       . 00005200
   1  C .                                                                 . 00005300
   1  C .       . TAPE17  STORES SEQUENTIALLY                            . 00005400
   1  C .         (1) EFFECTIVE LINEAR STIFFNESS MATRIX  (IN LINEAR      . 00005500
   1  C .             ANALYSIS STORED IN FACTORED FORM)                  . 00005600
   1  C .         (2) LUMPED MASS MATRIX (IN CASE IMASS.EQ.1 ONLY)       . 00005700
   1  C .                                                                . 00005800
   1  C .       . TAPE18  STORES SEQUENT&ALLY                            . 00005900
   1  C .         (1) ID ARRAY                                           . 00006000
   1  C .         (2) DISPL, VELOCITY AND ACCELERATION FOR RESTART       . 00006100
   1  C .             OR FOR INITIAL CONDITIONS                          . 00006200
   1  C .                                                                . 00006300
   1  C .       . TAPE 20 STORES THE EFFECTI5E N6N-I5EAR STIFFNESS IN    . 00006400
   1  C .               FACTORED FORM                                    . 00006500
   1  C .                                                                . 00006600
   1  C .       . TAPE 21 STORES EITHER                                  . 00006700
   1  C .         (1) NODAL MASSES AND DAMPERS (DURING MATRIX ASSEMLAGE) . 00006800
   1  C .         (2) EIGENVALUES AND CORRESPONDING EIGENVECTORS         . 00006900
   1  C .             (DURING FREQUENCY ANALYSIS)                        . 00007000
   1  C .                                                                . 00007100
   1  C . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 00007200
   1  C                                                                  00007300
   1  C                                                                  00007400
   1  C                                                                  00007500
   7        IMPLICIT REAL*8(A-H,O-Z)                                    00007600
   7        COMMON/BBCCIO/KIO1,KIO2,KIO3,KIO4,KIO5,KIO6,KIO7,KIO8,KIO9,  00007700
   6       1KIO10,KIO11,KIO12,KIO13,KIO14,KIO15,KIO16,KIO17,KIO18,KIO19,KIO20,00007800
   6       2KIO21,KIO22,KIO23,KIO24,KIO25,KIO26,KIO27,KIO28,KIO29,KIO30  00007900
   7        COMMON /SOL/ NUMNP,NEQ,NWK,NWM,NWC,NUMEST,MIDEST,MAXEST,NSTE,MA  00008000
   7        COMMON /DIM/ N1,N2,N3,N4,N5,N6,N7,N8,N9,N10,N11,N12,N13,N14,N15  00008100
   7        COMMON /EL/ IND,ICOUNT,NPAR(20),NUMEG,NEGL,NEGNL,IMASS,IDAMP,ISTAT00008200
   6       1           ,NDOF,KLIN,IEIG,IMASSN,IDAMPN                      00008300
   7        COMMON /CONST/ DT,DTA,A0,A1,A2,A3,A4,A5,A6,A7,A8,A9,A10,A11    00008400
   6       1              ,A12,A13,A14,A15,A16,A17,A18,A19,A20,IOPE        00008500
   7        COMMON /LOA/ NLOAD,NPTM                                       00008600
   7        COMMON /JUNK/ HED(12),MTOT                                    00008700
   7        COMMON /VAR/ NG,KPRI,MODEX,KSTEP,ITE,ITEMAX,IREF,IEQREF,INOCMD  00008800
   7        COMMON /TAPES/ NREAD,NWRITE                                   00008900
   7        COMMON /NORMS/ RNORM,RENORM,RTOL                              00009000
   7        COMMON /CRACK/ GAMMA,STIFAC,SHEFAC                            00009100
   7        COMMON /PRCON/ NPB,IDC,IVC,IAC,IPC,IPNODE(2,8)                00009200
   7        COMMON /DAMPNG/ ADAMP,BDAMP                                   00009300
   7        COMMON /ELGFLG/ NP1MX,NP7MX                                   00009400
   7        COMMON /NSAPI/ OPVAR(7),TSTART,IEQUIT,NUMREF,IPRI,ISREF       00009500
   7        COMMON A(15001)                                               00009600
   1  C                                                                  00009700
   1  C                                                                  00009800
```

```
START                                                                                                  MOD
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8  FLAGS
   1  C                                                                                  00009900
   1  C  ERRSET IS CALLED TO SUPPRESS THE UNDERFLOW.                                     00010000
   1  C                                                                                  00010100
   7        CALL BBCRIO                                                                 00010200
   7        CALL ERRSET(208,0,-1,0,0,0)                                                 00010300
   7        CALL STIME                                                                  00010400
   7        MTOT=15000                                                                  00010500
   3      200 NUMEST=3000                                                               00010600
   7        MAXEST=0                                                                    00010700
   7        NWRITE =KI019                                                               00010800
   7        NREAD =KI012                                                                00010900
   7        ITE=0                                                                       00011000
   7        KPRI=1                                                                      00011100
   1  C                                                                                  00011200
   1  C                                                                                  00011300
   1  C     I N P U T   P H A S E                                                        00011400
   1  C                                                                                  00011500
   1  C                                                                                  00011600
   7        CALL TTIME (TIM1)                                                           00011700
   1  C                                                                                  00011800
   7        CALL NONSPI                                                                 00011900
   1  C                                                                                  00012000
   1  C                                                                                  00012100
   1  C     O B T A I N   E L E M E N T   I N F O R M A T I O N S                        00012200
   1  C                                                                                  00012300
   1  C                                                                                  00012400
   7        DO 10 I=N5,N6                                                               00012500
   4     10 A(I)=0.                                                                     00012600
   7        IND=0                                                                       00012700
   7        MA=0                                                                        00012800
   1  C                                                                                  00012900
   7        CALL ELCAL                                                                  00013000
   1  C                                                                                  00013100
   7        CALL TTIME (TIM2)                                                           00013200
   1  C                                                                                  00013300
   1  C                                                                                  00013400
   1  C     S O L U T I O N   P H A S E                                                  00013500
   1  C                                                                                  00013600
   1  C                                                                                  00013700
   7        MA=MA + 1                                                                   00013800
   7        IREF=0                                                                      00013900
   1  C                                                                                  00014000
   1  C     ASSEMBLE EFFECTIVE LINEAR STIFFNESS MATRIX                                   00014100
   1  C                                                                                  00014200
   7        N1=1 + MAXEST                                                               00014300
   7        CALL ADDRES (A(N1),A(N5))                                                   00014400
   1  C                                                                                  00014500
   1  C     IF THREE-DIMENSIONAL ELEMENTS ARE USED                                       00014600
   1  C     MAKE ROOM FOR ELEMENT STIFFNESSES                                            00014700
```

```
START                                                                                                    MOD
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8                   FLAGS
   1  C                                                                                                 00014800
   7        IF (NP1MX.NE.3) GO TO 15                                                                    00014900
   7        NDM=3*NP7MX                                                                                 00015000
   7        NDM2=(NDM*NDM)/2 + NDM/2 + 1                                                                00015100
   7        NN=N1                                                                                       00015200
   7        N1=N1 + NDM2 + 4*NDM                                                                        00015300
   7        NEQ1=NEQ + 1                                                                                00015400
   7        DO 12 I=1,NEQ1                                                                              00015500
   4     12 A(N1+NEQ1-I)=A(NN+NEQ1-I)                                                                   00015600
   1  C                                                                                                 00015700
   4     15 N2=N1 + NEQ + 1                                                                            00015800
   7        N3=N2 + NWK - 1                                                                             00015900
   7        N4=N3 + 1                                                                                   00016000
   7        N5=N4 + NEQ                                                                                 00016100
   7        N6=N5 + NEQ - 1                                                                             00016200
   7        IF (IMASS.NE.1) N6=N5                                                                       00016300
   7        IF (N6.GT.MTOT) CALL ERROR (N6 - MTOT)                                                      00016400
   7        IF (MODEX.GT.0) GO TO 19                                                                    00016500
   7        CALL TTIME (TIM3)                                                                           00016600
   7        TIM4=TIM3                                                                                   00016700
   7        TIM5=TIM3                                                                                   00016800
   4     19 DO 20 I=N2,N6                                                                              00016900
   4     20 A(I)=0.                                                                                    00017000
   7        IND=1                                                                                       00017100
   7        ICOUNT=1                                                                                    00017200
   7        REWIND KIO17                                                                                00017300
   7        REWIND KIO14                                                                                00017400
   7        IF (MODEX.EQ.2) GO TO 34                                                                    00017500
   1  C                                                                                                 00017600
   7        CALL ASSEM                                                                                  00017700
   1  C                                                                                                 00017800
   1  34    CALL TTIME (TIM3)                                                                           00017900
   7        IF (KLIN.GT.0) GO TO 35                                                                     00018000
   7        IF (MODEX.EQ.2) GO TO 35                                                                    00018100
   1  C                                                                                                 00018200
   1  C     IN CASE OF LINEAR ANALYSIS TRIANGULARIZE EFFECTIVE STIFFNESS                                00018300
   1  C     MATRIX                                                                                      00018400
   1  C                                                                                                 00018500
   7        KTR=1                                                                                       00018600
   7        CALL COLSOL (A(N2),A(N4),A(N1),NEQ,MA,NWK,KTR)                                              00018700
   1  C                                                                                                 00018800
   1  35    CALL TTIME (TIM4)                                                                           00018900
   7        IF (MODEX.EQ.2) GO TO 36                                                                    00019000
   7        WRITE(KIO17) (A(I),I=N2,N3)                                                                 00019100
   1  C                                                                                                 00019200
   1  C     ASSEMBLE MASS MATRIX                                                                        00019300
   1  C                                                                                                 00019400
   4     36 IF (ISTAT.EQ.0) GO TO 40                                                                   00019500
   7        NWM=NEQ                                                                                     00019600
```

```
START                                                                                                  MOD
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8  FLAGS
   7        IF (IMASS.EQ.2) NWM=NWK                                                    00019700
   7        N3=N2 + NWM - 1                                                            00019800
   7        DO 30 I=N2,N3                                                              00019900
   4     30 A(I)=0.                                                                    00020000
   7        IND=2                                                                      00020100
   7        IF (MODEX.EQ.2) GO TO 40                                                   00020200
   1  C                                                                                00020300
   7        CALL ASSEM                                                                 00020400
   1  C                                                                                00020500
   7        NTAPE = KIO14                                                              00020600
   7        IF (IMASS.EQ.1) NTAPE=KIO17                                                00020700
   7        WRITE (NTAPE) (A(I),I=N2,N3)                                               00020800
   1  C                                                                                00020900
   1  C     STORE CONCENTRATED DAMPING VECTOR ON TAPE 4                                00021000
   1  C                                                                                00021100
   1  40    CALL TTIME (TIM5)                                                          00021200
   7        IF (MODEX.EQ.2) GO TO 60                                                   00021300
   7        IF (IDAMPN.EQ.0) GO TO 60                                                  00021400
   7        N3=N2 + NEQ - 1                                                            00021500
   7        REWIND KIO21                                                               00021600
   7        IF (IMASSN.NE.0) READ (KIO21)                                              00021700
   7        READ(KIO21) (A(I),I=N2,N3)                                                 00021800
   7        WRITE(KIO14) (A(I),I=N2,N3)                                                00021900
   1  C                                                                                00022000
   4     60 N3=N2 + NEQ                                                                00022100
   7        N4=N3 + NEQ                                                                00022200
   7        N5=N4 + NWK                                                                00022300
   7        N6=N5 + NEQ                                                                00022400
   7        N7=N6 + NEQ                                                                00022500
   7        NN1=NDOF*NUMNP                                                             00022600
   7        NN2=2*NEQ                                                                  00022700
   7        IF (NN1.GT.NN2) N7=N5 + NN1                                                00022800
   7        N8=N7 + NEQ                                                                00022900
   7        N9=N8 + NEQ                                                                00023000
   7        N10=N9 + NEQ                                                               00023100
   7        NN=N10                                                                     00023200
   7        IF (IMASS.EQ.2) NN=N9                                                      00023300
   7        IF (ISTAT.EQ.0) NN=N7                                                      00023400
   7        IF (NN.GT.MTOT) CALL ERROR (NN-MTOT)                                       00023500
   1  C                                                                                00023600
   1  C     WRITE INITIALIZED DISPLACEMENTS, VELOCITIES, AND ACCELERATIONS             00023700
   1  C        ( OR STARTING DISPL/VEL/ACC IF THIS IS A RESTART JOB )                  00023800
   1  C                                                                                00023900
   7        WRITE(KIO6,2010)                                                           00024000
   7        REWIND KIO18                                                               00024100
   7        READ (KIO18)                                                               00024200
   7        NN=N3 - 1                                                                  00024300
   7        READ(KIO18) (A(I),I=N2,NN)                                                 00024400
   7        IND=0                                                                      00024500
```

```
START                                                                                                MOD
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8  FLAGS

   7        IF (ISTAT.EQ.0) GO TO 80                                                    00024600
   7        NN=N8 - 1                                                                   00024700
   7        READ(KIO18) (A(I),I=N7,NN)                                                  00024800
   7        NN=N9 - 1                                                                   00024900
   7        READ(KIO18) (A(I),I=N8,NN)                                                  00025000
   4     80 CALL WRITE (A(N2),A(N7),A(N8),A(N5),NEQ,NDOF,NUMNP)                         00025100
   1  C                                                                                 00025200
   1  C                                                                                 00025300
   1  C     F R E Q U E N C Y   S O L U T I O N                                         00025400
   1  C                                                                                 00025500
   1  C                                                                                 00025600
   7        CALL TTIME (TIM6)                                                           00025700
   7        IF (IEIG.NE.1) GO TO 69                                                     00025800
   7        IF (MODEX.EQ.2) READ(KIO5,2110) A(1),A(2),A(3)                              00025900
   7        IF (MODEX.EQ.2) GO TO 69                                                    00026000
   7        IND=3                                                                       00026100
   1  C                                                                                 00026200
   7        CALL ASSEM                                                                  00026300
   1  C                                                                                 00026400
   1  69    CALL TTIME (TIM7)                                                           00026500
   1  C                                                                                 00026600
   1  C                                                                                 00026700
   1  C     T I M E   I N T E G R A T I O N                                             00026800
   1  C                                                                                 00026900
   1  C                                                                                 00027000
   1  85    CALL TTIME (TIM8)                                                           00027100
   7        TSUM1=0.                                                                    00027200
   7        TSUM2=0.                                                                    00027300
   7        TSUM3=0.                                                                    00027400
   7        TSUM4=0.                                                                    00027500
   7        TSUM5=0.                                                                    00027600
   7        TSUM6=0.                                                                    00027700
   7        IF (MODEX.GT.0) GO TO 88                                                    00027800
   7        WRITE(KIO6,2030)                                                            00027900
   7        GO TO 190                                                                   00028000
   4     88 IF (NSTE.EQ.0) GO TO 190                                                    00028100
   1  C                                                                                 00028200
   3    130 TIME=TSTART                                                               00028300
   7        TIMEP=TSTART                                                                00028400
   7        REWIND KIO13                                                                00028500
   7        IND=4                                                                       00028600
   7        KSTEP=0                                                                     00028700
   7        LPRI=0                                                                      00028800
   7        KSREF=0                                                                     00028900
   7        KEQUIT=0                                                                    00029000
   1  C                                                                                 00029100
   1  C     INITIAL CONDITIONS ARE TAKEN INTO CORE                                      00029200
   1  C                                                                                 00029300
   7        IF (IEIG.EQ.0) GO TO 89                                                     00029400
```

```
START                                                                                MOD
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8  FLAGS

   7        REWIND KIO18                                                          00029500
   7        READ(KIO18)                                                           00029600
   7        NN=N3 - 1                                                             00029700
   7        READ(KIO18) (A(I),I=N2,NN)                                            00029800
   7        IF (ISTAT.EQ.0) GO TO 89                                              00029900
   7        NN=N8 - 1                                                             00030000
   7        READ(KIO18) (A(I),I=N7,NN)                                            00030100
   7        NN=N9 - 1                                                             00030200
   7        READ(KIO18) (A(I),I=N8,NN)                                            00030300
   4     89 CONTINUE                                                              00030400
   1  C                                                                           00030500
   1  C     IF THIS IS A RESTARTED JOB TRANSFER NONLINEAR ELEMENT GROUP            00030600
   1  C     DATA TO TAPE 12                                                       00030700
   1  C                                                                           00030800
   7        IF (MODEX.NE.2) GO TO 91                                              00030900
   7        REWIND KIO12                                                          00031000
   7        REWIND KIO19                                                          00031100
   7        DO 92 NG=1,NEGNL                                                      00031200
   7        READ(KIO19) NUMEST,NPAR,(A(I),I=1,NUMEST)                             00031300
   4     92 WRITE(KIO12) NUMEST,NPAR,(A(I),I=1,NUMEST)                            00031400
   1  C     LUMPED MASS MATRIX IS TAKEN INTO CORE                                 00031500
   1  C                                                                           00031600
   4     91 IF (IMASS.NE.1) GO TO 90                                              00031700
   7        NN=N10 - 1                                                            00031800
   1  C                                                                           00031900
   7        IF (MODEX.NE.2) GO TO 93                                              00032000
   7        REWIND KIO17                                                          00032100
   7        READ(KIO17)                                                           00032200
   7        GO TO 94                                                              00032300
   4     93 BACKSPACE KIO17                                                       00032400
   4     94 READ(KIO17) (A(I),I=N9,NN)                                            00032500
   1  C                                                                           00032600
   1  C     THE TRIANGULARIZED EFFECTIVE STIFFNESS MATRIX IS TAKEN INTO CORE      00032700
   1  C     PROVIDED THAT                                                         00032800
   1  C        1.   LINEAR ANALYSIS                                               00032900
   1  C        2.   NO CONSISTENT MASS MATRIX                                     00033000
   1  C        3.   NO RAYLEIGH DAMPING                                           00033100
   1  C                                                                           00033200
   4     90 IF (KLIN.NE.0 .OR. INOCMD.NE.0) GO TO 100                             00033300
   7        NN=N5-1                                                               00033400
   7        REWIND KIO17                                                          00033500
   7        READ(KIO17) (A(I),I=N4,NN)                                            00033600
   1  C                                                                           00033700
   1  C                                                                           00033800
   1  C     T I M E   S T E P   I N C R E M E N T A T I O N                       00033900
   1  C                                                                           00034000
   1  C        KSTEP .EQ. STEP COUNTER                                            00034100
   1  C        TIME .EQ. TIME AT WHICH SOLUTION IS REQUIRED                       00034200
   1  C                                                                           00034300
```

```
START                                                                                                  MOD
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8  FLAGS
   1  C                                                                                 00034400
   3    100 KSTEP=KSTEP + 1                                                           00034500
   7        TIMEP=TIME + DTA                                                          00034600
   7        TIME=TIME + DT                                                            00034700
   1  C                                                                                 00034800
   1  C     STIFFNESS REFORMATION FLAG                                                  00034900
   1  C        IREF.EQ.0 IF STIFFNESS IS TO BE REFORMED                                 00035000
   1  C                                                                                 00035100
   7        KSREF=KSREF + 1                                                           00035200
   7        IREF=ISREF - KSREF                                                        00035300
   7        IF (IREF.EQ.0) KSREF=0                                                    00035400
   7        IF (KSTEP.EQ.1) IREF=0                                                    00035500
   1  C                                                                                 00035600
   1  C     FLAG FOR EQUILIBRIUM ITERATION                                              00035700
   1  C        IIT.EQ.0 IF ITERATION IS TO BE PERFORMED                                 00035800
   1  C        IIT.GT.0 IF NO ITERATION IS TO BE PERFORMED                              00035900
   1  C                                                                                 00036000
   3    104 KEQUIT=KEQUIT + 1                                                         00036100
   7        IIT=IEQUIT - KEQUIT                                                       00036200
   7        IF (IIT.EQ.0) KEQUIT=0                                                    00036300
   1  C                                                                                 00036400
   1  C     FLAG FOR TRIANGULARIZATION AND/OR SIMPLE REDUCTION AND                      00036500
   1  C     BACKSUBSTITUTION IN COLSOL                                                  00036600
   1  C        KTR.EQ.1   FOR TRIANGULARIZATION PLUS SOLUTION                           00036700
   1  C        KTR.EQ.2   FOR VECTOR SOLUTION ONLY                                      00036800
   1  C                                                                                 00036900
   7        KTR=1                                                                     00037000
   7        IF (IREF.NE.0) KTR=2                                                      00037100
   7        IF (KLIN.EQ.0) KTR=2                                                      00037200
   1  C                                                                                 00037300
   1  C     NEQREF IS THE NUMBER OF TIMES THE NONLINEAR STIFFNESS MATRIX                00037400
   1  C     WAS REFORMED                                                                00037500
   1  C                                                                                 00037600
   7        NEQREF=0                                                                  00037700
   3    140 REWIND KIO14                                                              00037800
   7        REWIND KIO17                                                              00037900
   7        REWIND KIO20                                                              00038000
   1  C                                                                                 00038100
   1  C     FLAG TO INDICATE CONVERGENCE IN EQUILIBRIUM ITERATION                       00038200
   1  C        IEQREF.EQ.0   CONVERGENCE                                                00038300
   1  C        IEQREF.EQ.1   NORM OF OUT-OF-BALANCE LOADS IS LARGER THAN NORM           00038400
   1  C                      OF INCREMENTAL LOADS (SEE EQUIT)                           00038500
   1  C                                                                                 00038600
   7        IEQREF=0                                                                  00038700
   1  C                                                                                 00038800
   1  C                                                                                 00038900
   1  C     S O L U T I O N   O F   I N C R E M E N T A L   E Q U A T I O N S           00039000
   1  C                                                                                 00039100
   1  C                                                                                 00039200
```

```
START                                                                                MOD
 COL ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8  FLAGS

   1  C     CALCULATE LINEAR EFFECTIVE LOADS BALANCED IN CURRENT CONFIGURATION00039300
   1  C                                                                      00039400
   7        CALL TTIME (TIM9)                                              00039500
   1  C                                                                      00039600
   7        CALL LOADEF (A(N1),A(N2),A(N7),A(N8),A(N3),A(N6),A(N4),A(N9))  00039700
   1  C                                                                      00039800
   7        CALL TTIME (TIM10)                                             00039900
   1  C                                                                      00040000
   1  C     CALCULATE EFFECTIVE NONLINEAR MATRIX AND FINAL EFFECTIVE LOADS  00040100
   1  C                                                                      00040200
   7        CALL ASSEM                                                     00040300
   1  C                                                                      00040400
   7        CALL TTIME (TIM11)                                             00040500
   1  C                                                                      00040600
   1  C     SOLVE FOR INCREMENT IN DISPLACEMENT VECTOR, NO TRIANGULARIZATION 00040700
   1  C     IF SIMPLE EQUILIBRIUM ITERATION IS TO BE PERFORMED              00040800
   1  C                                                                      00040900
   7        CALL COLSOL (A(N4),A(N3),A(N1),NEQ,MA,NWK,KTR)                 00041000
   1  C                                                                      00041100
   7        CALL TTIME (TIM12)                                             00041200
   7        TSUM1=TSUM1 + (TIM10 - TIM9)                                   00041300
   7        TSUM2=TSUM2 + (TIM11 - TIM10)                                  00041400
   7        TSUM3=TSUM3 + (TIM12 - TIM11)                                  00041500
   1  C                                                                      00041600
   1  C     NO ITERATION IN LINEAR ANALYSIS                                00041700
   1  C                                                                      00041800
   7        IF (KLIN.EQ.0) GO TO 110                                       00041900
   1  C                                                                      00042000
   1  C     THE FACTORED NONLINEAR EFFECTIVE STIFFNESS MATRIX IS STORED ON  00042100
   1  C     TAPE 20 IF THE FOLLOWING CONDITIONS HOLD                       00042200
   1  C        1.   A NEW EFFECTIVE STIFFNESS MATRIX WAS TRIANGULARIZED    00042300
   1  C             (IREF.EQ.0)                                            00042400
   1  C        2.   THE EFFECTIVE STIFFNESS MATRIX IS NEEDED IN THE        00042500
   1  C             NEXT TIME STEP (ISREF.NE.1)                            00042600
   1  C        3.   THE EFFECTIVE STIFFNESS IS NEEDED IN THE EQUILIBRIUM   00042700
   1  C             ITERATION IN THIS TIME STEP (IIT.EQ.0)                 00042800
   1  C                                                                      00042900
   7        IF (IREF.NE.0) GO TO 105                                       00043000
   7        IF (ISREF.EQ.1 .AND. IIT.GT.0) GO TO 105                       00043100
   7        NN=N5 - 1                                                      00043200
   7        WRITE (KIO20) (A(I),I=N4,NN)                                   00043300
   1  C                                                                      00043400
   1  C                                                                      00043500
   1  C     I T E R A T I O N   F O R   D Y N A M I C   E Q U I L I B R I U M 00043600
   1  C                                                                      00043700
   1  C                                                                      00043800
   3    105 IF (IIT.GT.0) GO TO 110                                         00043900
   1  C                                                                      00044000
   7        CALL TTIME (TIM13)                                             00044100
```

```
START                                                                                                  MOD
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8  FLAGS

   1  C                                                                                 00044200
   7        CALL EQUIT (A(N4),A(N3),A(N5),A(N2),A(N7),A(N8),A(N1),A(N6),A(N9))        00044300
   1  C                                                                                 00044400
   7        CALL TTIME (TIM14)                                                        00044500
   7        TSUM4=TSUM4 + (TIM14 - TIM13)                                             00044600
   1  C                                                                                 00044700
   1  C     CHECK FOR NO CONVERGENCE IN EQUILIBRIUM ITERATION AND                       00044800
   1  C     POSSIBLE REFORMATION OF STIFFNESS                                           00044900
   1  C                                                                                 00045000
   7        IF (IEQREF.EQ.0) GO TO 110                                                00045100
   7        KTR=1                                                                     00045200
   7        IREF=0                                                                    00045300
   7        NEQREF=NEQREF + 1                                                         00045400
   7        IF (NUMREF.EQ.0) GO TO 112                                                00045500
   7        IF (NEQREF.LE.1) GO TO 140                                                00045600
   3    112 WRITE(KIO6,2040)                                                          00045700
   7        STOP                                                                      00045800
   1  C                                                                                 00045900
   1  C                                                                                 00046000
   1  C     C A L C U L A T I O N   O F   N E W   D I S P , V E L , A C C             00046100
   1  C                                                 V E C T O R S                 00046200
   1  C                                                                                 00046300
   1  C                                                                                 00046400
   1  110   CALL TTIME (TIM15)                                                         00046500
   1  C                                                                                 00046600
   7        CALL NEWDAV (A(N2),A(N7),A(N8),A(N3))                                     00046700
   1  C                                                                                 00046800
   7        CALL TTIME (TIM16)                                                        00046900
   7        TSUM5=TSUM5 + (TIM16 - TIM15)                                             00047000
   1  C                                                                                 00047100
   1  C     PRINT FLAG                                                                  00047200
   1  C        KPRI.EQ.0 FOR PRINT OUT OF DISP, VEL, ACC AND STRESSES                   00047300
   1  C                                                                                 00047400
   7        LPRI=LPRI + 1                                                             00047500
   7        KPRI=IPRI - LPRI                                                          00047600
   7        IF (KPRI.GT.0) GO TO 120                                                  00047700
   7        LPRI=0                                                                    00047800
   1  C                                                                                 00047900
   7        WRITE(KIO6,2020) KSTEP,TIME                                               00048000
   7        IF (IIT.EQ.0) WRITE (KIO6,2060)ITE                                        00048100
   7        IF (IIT.GT.0) WRITE(KIO6,2050)                                            00048200
   7        IF (IREF.EQ.0) WRITE(KIO6,2070)                                           00048300
   7        IF (IREF.NE.0) WRITE(KIO6,2080)                                           00048400
   1  C                                                                                 00048500
   1  C     PRINT DISPLACEMENTS,VELOCITIES AND ACCELERATIONS                            00048600
   1  C                                                                                 00048700
   7        IF (IPC.NE.0)                                                             00048800
   6       1CALL WRITE (A(N2),A(N7),A(N8),A(N5),NEQ,NDOF,NUMNP)                       00048900
   1  C                                                                                 00049000
```

```
START                                                                                                                MOD
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8  FLAGS

   7        CALL TTIME (TIM17)                                                                                    00049100
   7        TSUM5=TSUM5 + (TIM17 - TIM16)                                                                         00049200
   1  C                                                                                                           00049300
   1  C                                                                                                           00049400
   1  C     C A L C U L A T I O N   O F   S T R E S S E S                                                         00049500
   1  C                                                                                                           00049600
   1  C                                                                                                           00049700
   7        CALL STRESS                                                                                           00049800
   1  C                                                                                                           00049900
   7        CALL TTIME (TIM18)                                                                                    00050000
   7        TSUM6=TSUM6 + (TIM18 - TIM17)                                                                         00050100
   7        KPRI=1                                                                                                00050200
   3     120 IF (KSTEP.LT.NSTE) GO TO 100                                                                          00050300
   1  C                                                                                                           00050400
   1  C                                                                                                           00050500
   1  C     P R I N T   T I M E   L O G                                                                           00050600
   1  C                                                                                                           00050700
   1  C                                                                                                           00050800
   1  190   CALL TTIME (TIM19)                                                                                    00050900
   7        WRITE(KIO6,2090) HED                                                                                  00051000
   7        TIM10=TIM2 - TIM1                                                                                     00051100
   7        TIM11=(TIM3 - TIM2) + (TIM5 - TIM4)                                                                   00051200
   7        TIM12=TIM4 - TIM3                                                                                     00051300
   7        TIM13=TIM7 - TIM6                                                                                     00051400
   7        TIM14=TIM19 - TIM8                                                                                    00051500
   7        TIM15=TIM19 - TIM1                                                                                    00051600
   7        WRITE(KIO6,2100) TIM10,TIM11,TIM12,TIM13, NSTE,TSUM1,TSUM2,TSUM3,                                     00051700
   6       1               TSUM4,TSUM5,TSUM6,TIM14,TIM15                                                          00051800
   1  C                                                                                                           00051900
   1  C                                                                                                           00052000
   1  C     P R E P A R E   T A P E S   F O R   P O S S I B L E                                                   00052100
   1  C                                                                                                           00052200
   1  C                       F U T U R E   R E S T A R T   J O B S                                               00052300
   1  C                                                                                                           00052400
   1  C                                                                                                           00052500
   7        IF (NREAD.EQ.KIO19) GO TO 910                                                                         00052600
   7        REWIND KIO12                                                                                          00052700
   7        REWIND KIO19                                                                                          00052800
   7        DO 900 NG=1,NEGNL                                                                                     00052900
   7        READ(KIO12) NUMEST,NPAR,(A(I),I=1,NUMEST)                                                             00053000
   3     900 WRITE(KIO19) NUMEST,NPAR,(A(I),I=1,NUMEST)                                                           00053100
   3     910 REWIND KIO18                                                                                         00053200
   7        READ (KIO18)                                                                                          00053300
   7        NN=N3 - 1                                                                                             00053400
   7        WRITE(KIO18) (A(I),I=N2,NN)                                                                           00053500
   7        IF (ISTAT.EQ.0) GO TO 200                                                                             00053600
   7        NN=N8 - 1                                                                                             00053700
   7        WRITE(KIO18) (A(I),I=N7,NN)                                                                           00053800
   7        NN=N9 - 1                                                                                             00053900
```

```
START                                                                    MOD
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8  FLAGS
   7        WRITE(KIO18) (A(I),I=N8,NN)                                  00054000
   1  C                                                                  00054100
   7        GO TO 200                                                    00054200
   1  C                                                                  00054300
   1  C                                                                  00054400
   2   2010 FORMAT (1H1,38H I N I T I A L   C O N D I T I O N S     ///)   00054500
   2   2020 FORMAT (1H1,46H P R I N T   O U T   F O R   T I M E   S T E P ,I5,00054600
   6       1 40X,12H ( AT TIME  ,E10.4,2H ) )                           00054700
   2   2030 FORMAT(////41H  D A T A   C H E C K   C O M P L E T E D)     00054800
   2   2040 FORMAT (//// 64H STOP BECAUSE OUT-OF-BALANCE LOADS LARGER THAN INC00054900
   6       1REMENTAL LOADS )                                            00055000
   2   2050 FORMAT (/40X,44HNO EQUILIBRIUM ITERATION ON THIS TIME STEP       )00055100
   2   2060 FORMAT (/17X,I5,79H EQUILIBRIUM ITERATIONS PERFORMED ON THIS TIME 00055200
   6       1STEP TO REESTABLISH EQUILIBRIUM  )                          00055300
   2   2070 FORMAT (40X, 48HSTIFFNESS REFORMED FOR THIS TIME STEP           )00055400
   2   2080 FORMAT (40X,42HSTIFFNESS NOT REFORMED FOR THIS TIME STEP        )00055500
   2   2090 FORMAT (1H1,34H S O L U T I O N   T I M E   L O G//12X,         00055600
   6       1        11HFOR PROBLEM//1X,12A6////)                        00055700
   2   2100 FORMAT (49H INPUT PHASE . . . . . .F9.2//                      00055800
   6       1        49H ASSEMBLAGE OF LINEAR STIFFNESS,                  /00055900
   6       2        49H   EFFECTIVE STIFFNESS, AND MASS MATRICES . . . .F9.2//00056000
   6       3        49H TRIANGULARIZATION OF EFFECTIVE STIFFNESS MATRIX F9.2//00056100
   6       4        49H FREQUENCY ANALYSIS . . . . . .F9.2//            00056200
   6       5        24H STEP-BY-STEP SOLUTION (,I5,12H TIME STEPS)       //00056300
   6       6        43H    CALCULATION OF EFFECTIVE LOAD VECTORS .      ,F9.2/00056400
   6       7        43H    UPDATING EFFECTIVE STIFFNESS MATRICES          /00056500
   6       8        43H      AND LOAD VECTORS FOR NONLINEARITIES .      ,F9.2/00056600
   6       9        43H    SOLUTION OF EQUATIONS . . . . . .            ,F9.2/00056700
   6       A        43H    EQUILIBRIUM ITERATIONS . . . . . .           ,F9.2/00056800
   6       B        43H    CALCULATION AND PRINTING OF DISPLACE-          /00056900
   6       C        43H      MENTS, VELOCITIES, AND ACCELERATIONS       ,F9.2/00057000
   6       D        43H    CALCULATION AND PRINTING OF STRESSES  .      F9.2//00057100
   6       E        30X,19H STEP-BY-STEP TOTAL ,F9.2/////                00057200
   6       F        49H T O T A L   S O L U T I O N   T I M E . . . . .F9.2) 00057300
   2   2110 FORMAT (I5,E10.4,I5)                                        00057400
   7        END                                                         00057500
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7        SUBROUTINE ADDBAN (A,MAXA,S,LM,ND)                                  00000100
   7        IMPLICIT REAL*8(A-H,O-Z)                                            00000200
   1  C                                                                         00000300
   1  C                                                                         00000400
   1  C . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 00000500
   1  C .                                                                     . 00000600
   1  C .   ASSEMBLES UPPER TRIANGULAR ELEMENT STIFFNESS INTO                 . 00000700
   1  C .         COMPACTED GLOBAL STIFFNESS                                  . 00000800
   1  C .                                                                     . 00000900
   1  C .         A = GLOBAL STIFFNESS                                        . 00001000
   1  C .         S = ELEMENT STIFFNESS                                       . 00001100
   1  C .         ND = DEGREES OF FREEDOM IN ELEMENT STIFFNESS                . 00001200
   1  C .                                                                     . 00001300
   1  C .                     S(1)        S(2)        S(3)          . . .     . 00001400
   1  C .         S   =                   S(ND+1)     S(ND+2)       . . .     . 00001500
   1  C .                                             S(2*ND)       . . .     . 00001600
   1  C .                                                           . . .     . 00001700
   1  C .                                                                     . 00001800
   1  C .                                                                     . 00001900
   1  C .                     A(1)        A(3)        A(6)          . . .     . 00002000
   1  C .         A   =                   A(2)        A(5)          . . .     . 00002100
   1  C .                                             A(4)          . . .     . 00002200
   1  C .                                                           . . .     . 00002300
   1  C .                                                                     . 00002400
   1  C .                                                                     . 00002500
   1  C . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 00002600
   1  C                                                                         00002700
   1  C                                                                         00002800
   7        DIMENSION A(1),MAXA(1),S(1),LM(1)                                   00002900
   1  C                                                                         00003000
   7        NDI=0                                                               00003100
   7        DO 200 I=1,ND                                                       00003200
   7        II=LM(I)                                                            00003300
   7        IF (II) 200,200,100                                                 00003400
   2   100  MI=MAXA(II)                                                        00003500
   7        KS=I                                                                00003600
   7        DO 220 J=1,ND                                                       00003700
   7        JJ=LM(J)                                                            00003800
   7        IF (JJ) 220,220,110                                                 00003900
   2   110  IJ=II - JJ                                                         00004000
   7        IF (IJ) 220,210,210                                                 00004100
   2   210  KK=MI + IJ                                                         00004200
   7        KSS=KS                                                              00004300
   7        IF (J.GE.I) KSS=J + NDI                                             00004400
   7        A(KK)=A(KK) + S(KSS)                                                00004500
   3    220 KS=KS + ND - J                                                      00004600
   3    200 NDI=NDI + ND - I                                                    00004700
   1  C                                                                         00004800
   7        RETURN                                                              00004900
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7        END                                                          00005000
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8  MOD FLAGS

   7        SUBROUTINE ADDCM (MAXA,MATRIX,XMN,NEQ)                                  00000100
   7        IMPLICIT REAL*8(A-H,O-Z)                                                00000200
   1  C                                                                             00000300
   1  C                                                                             00000400
   1  C     ADDS NODAL MASSES AND DAMPERS TO GLOBAL MASS MATRIX                     00000500
   1  C     OR TO GLOBAL EFFECTIVE STIFFNESS MATRIX                                 00000600
   1  C     NODAL DAMPERS ARE RECOMMENDED FOR USE ON BOUNDARY ELEMENTS              00000700
   1  C     TO SIMULATE ENERGY ABSORBING BOUNDARY CONDITION.                        00000800
   1  C     TRANSFORMATION AND ADD. ROUTINE: BY S.N.MUKHERJEE.BBC,BADEN JUNE,       00000901   *
   1  C     1976.                                                                   00001001   *
   7        COMMON/BBCCIO/KIO1,KIO2,KIO3,KIO4,KIO5,KIO6,KIO7,KIO8,KIO9,            00001100
   6       1KIO10,KIO11,KIO12,KIO13,KIO14,KIO15,KIO16,KIO17,KIO18,KIO19,KIO20,     00001200
   6       2KIO21,KIO22,KIO23,KIO24,KIO25,KIO26,KIO27,KIO28,KIO29,KIO30            00001300
   1  C                                                                             00001400
   7        REAL*8 MATRIX(1)                                                        00001600
   7        DIMENSION MAXA(1),XMN(1)                                                00001700
   7        COMMON /EL/ IND,ICOUNT,NPAR(20),NUMEG,NEGL,NEGNL,IMASS,IDAMP,ISTAT      00001800
   6       1           ,NDOFDM,KLIN,IEIG,IMASSN,IDAMPN                             00001900
   7        COMMON /CONST/ DT,DTA,A0,A1,A2,A3,A4,A5,A6,A7,A8,A9,A10,A11             00002000
   6       1              ,A12,A13,A14,A15,A16,A17,A18,A19,A20,IOPE                 00002100
   7        COMMON /DAMPNG/ ADAMP,BDAMP                                             00002200
   1  C                                                                             00002400
   7        GO TO (1,100), IND                                                      00002500
   1  C                                                                             00002600
   1  C                                                                             00002700
   1  C     ADD NODAL MASSES AND DAMPERS TO EFFECTIVE STIFFNESS                     00002800
   1  C                                                                             00002900
   5      1 REWIND KIO21                                                          00003000
   7        IF (IMASSN.EQ.0) GO TO 10                                             00003100
   7        READ(KIO21) (XMN(I),I=1,NEQ)                                          00003200
   7        FAC=A0 + ADAMP*A1                                                     00003300
   7        DO 2 I=1,NEQ                                                          00003400
   7        IN=MAXA(I)                                                            00003500
   5      2 MATRIX(IN)=MATRIX(IN) + FAC*XMN(I)                                    00003600
   4     10 IF (IDAMPN.EQ.0) RETURN                                               00003700
   7        READ(KIO21) (XMN(I),I=1,NEQ)                                          00003800
   7        FAC=A1                                                                00003900
   7        DO 20 I=1,NEQ                                                         00004000
   7        IN=MAXA(I)                                                            00004100
   4     20 MATRIX(IN)=MATRIX(IN) + FAC*XMN(I)                                    00004200
   7        RETURN                                                                00004300
   1  C                                                                             00004400
   1  C                                                                             00004500
   1  C     ADD CONCENTRATED NODAL MASSES TO GLOBAL MASS MATRIX                     00004600
   1  C                                                                             00004700
   3    100 IF (IMASSN.EQ.0) RETURN                                               00004800
   7        REWIND KIO21                                                          00004900
   7        READ(KIO21) (XMN(I),I=1,NEQ)                                          00005000
   7        IF (IMASS.EQ.1) GO TO 120                                             00005100
```

```
START                                                                    MOD
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8  FLAGS

   7          DO 101 I=1,NEQ                                          00005200
   7          IN=MAXA(I)                                              00005300
   3    101   MATRIX(IN)=MATRIX(IN) + XMN(I)                          00005400
   7          RETURN                                                  00005500
   3    120   DO 121 I=1,NEQ                                          00005600
   3    121   MATRIX(I)=MATRIX(I) + XMN(I)                            00005700
   7          RETURN                                                  00005800
   1  C                                                               00005900
   1  C                                                               00006000
   7          END                                                     00007000
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7        SUBROUTINE ADDDI (A,MAXA,S,LM,ND)                                00000100
   7        IMPLICIT REAL*8(A-H,O-Z)                                         00000200
   1  C                                                                      00000300
   7        COMMON/BBCCIO/KIO1,KIO2,KIO3,KIO4,KIO5,KIO6,KIO7,KIO8,KIO9,      00000400
   6       1KIO10,KIO11,KIO12,KIO13,KIO14,KIO15,KIO16,KIO17,KIO18,KIO19,KIO20,00000500
   6       2KIO21,KIO22,KIO23,KIO24,KIO25,KIO26,KIO27,KIO28,KIO29,KIO30      00000600
   7        DIMENSION A(1),MAXA(1),S(1),LM(1)                                00000700
   1  C                                                                      00000800
   7        DO 100 I=1,ND                                                    00000900
   7        II=LM(I)                                                         00001000
   7        IF (II) 100,100,110                                              00001100
   2   110  MI=MAXA(II)                                                     00001200
   7        WRITE(KIO6,1)A(MI),S(I)                                          00001300
   7        A(MI)=A(MI) + S(I)                                               00001400
   5      1 FORMAT(' ADDDI',2E14.4)                                          00001500
   2   100  CONTINUE                                                        00001600
   1  C                                                                      00001700
   7        RETURN                                                           00001800
   7        END                                                              00001900
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7          SUBROUTINE ADDMA (A,S,LM,ND,FAC,IST)                          00000100
   7          IMPLICIT REAL*8(A-H,O-Z)                                      00000200
   1  C                                                                     00000300
   7          DIMENSION A(1),S(1),LM(1)                                     00000400
   1  C                                                                     00000500
   7          IF (IST.GT.0) GO TO 200                                       00000600
   1  C                                                                     00000700
   1  C       ADD ELEMENT MASS TO GLOBAL LUMPED MASS VECTOR                 00000800
   1  C                                                                     00000900
   7          DO 100 I=1,ND                                                 00001000
   7          II=LM(I)                                                      00001100
   7          IF (II) 100,100,110                                           00001200
   3      110 A(II)=A(II) + S(I)                                            00001300
   3      100 CONTINUE                                                      00001400
   7          RETURN                                                        00001500
   1  C                                                                     00001600
   1  C                                                                     00001700
   1  C       IF REUSE OPTION ACTIVATED FOR THIS ELEMENT,                   00001800
   1  C       DIVIDE BY *FAC* TO RECOVER ELEMENT MASS BEFORE ADDING IT      00001900
   1  C       TO GLOBAL LUMPED MASS VECTOR                                  00002000
   1  C                                                                     00002100
   3      200 DFAC=1./FAC                                                   00002200
   7          DO 300 I=1,ND                                                 00002300
   7          II=LM(I)                                                      00002400
   7          IF (II) 300,300,310                                           00002500
   3      310 A(II)=A(II) + S(I)*DFAC                                       00002600
   3      300 CONTINUE                                                      00002700
   7          RETURN                                                        00002800
   1  C                                                                     00002900
   1  C                                                                     00003000
   7          END                                                           00003100
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7        SUBROUTINE ADDRES(MAXA,MHT)                                          00000100
   7        IMPLICIT REAL*8(A-H,O-Z)                                             00000200
   1  C                                                                          00000300
   1  C . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 00000400
   1  C .                                                                      . 00000500
   1  C .   P R O G R A M                                                      . 00000600
   1  C .      . TO CALCULATE ADDRESSES OF DIAGONAL ELEMENTS IN BANDED         . 00000700
   1  C .        MATRIX WHOSE COLUMN HEIGHTS ARE KNOWN                         . 00000800
   1  C .                                                                      . 00000900
   1  C .        MHT  = ACTIVE COLUMN HEIGHTS                                  . 00001000
   1  C.         MAXA = ADDRESSES OF DIAGONAL ELEMENTS                         . 00001100
   1  C .                                                                      . 00001200
   1  C . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 00001300
   1  C                                                                          00001400
   7        COMMON /SOL/ NUMNP,NEQ,NWK,NWM,NWC,NUMEST,MIDEST,MAXEST,NSTE,MA      00001500
   7        DIMENSION MAXA(1),MHT(1)                                             00001600
   1  C                                                                          00001700
   7        MAXA(1)=1                                                            00001800
   7        MAXA(2)=2                                                            00001900
   7        MA=0                                                                 00002000
   7        IF (NEQ.EQ.1) GO TO 100                                              00002100
   7        DO 10 I=2,NEQ                                                        00002200
   7        IF (MHT(I).GT.MA) MA=MHT(I)                                          00002300
   4     10 MAXA(I+1)=MAXA(I) + MHT(I) + 1                                       00002400
   3    100 MA=MA + 1                                                           00002500
   7        NWK=MAXA(NEQ+1) - MAXA(1)                                            00002600
   1  C                                                                          00002700
   7        RETURN                                                               00002800
   7        END                                                                  00002900
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7        SUBROUTINE ASSEM                                                    00000100
   7        IMPLICIT REAL*8(A-H,O-Z)                                            00000200
   1  C                                                                         00000300
   1  C . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 00000400
   1  C .                                                                     . 00000500
   1  C .   P R O G R A M                                                   . 00000600
   1  C .      . TO ASSEMBLE THE MATRICES NEEDED IN SOLUTION                . 00000700
   1  C .                                                                     . 00000800
   1  C .   S T O R A G E                                                     . 00000900
   1  C .     1. TAPE14 STORES SEQUENTIALLY                                   . 00001000
   1  C .          (1) LINEAR STIFFNESS MATRIX                                . 00001100
   1  C .          (2) LINEAR MASS MATRIX                                     . 00001200
   1  C .          (3) NODAL DAMPING VECTOR                                   . 00001300
   1  C .        IF FREQUENCY ANALYSIS SPECIFIED                              . 00001400
   1  C .     2. TAPE17 STORES THE EFFECTIVE LINEAR STIFFNESS MATRIX          . 00001500
   1  C       3. TAPE 20 STORES LINEAR STIFFNESS MATRIX                       . 00001600
   1  C .     4. TAPE 21 STORES CONCENTRATED MASSES AND DAMPERS               . 00001700
   1  C .                                                                     . 00001800
   1  C .   E X E C U T I O N   M O D E                                       . 00001900
   1  C .        IND=1   EFFECTIVE LINEAR STIFFNESS MATRIX IS ASSEMBLES       . 00002000
   1  C .          ICOUNT=1   LINEAR STIFFNESS IS ASSEMBLED                  . 00002100
   1  C .          ICOUNT=2   MASS AND DAMPING EFFECTS ARE ADDED              . 00002200
   1  C .        IND=2   MASS MATRIX IS ASSEMBLED                             . 00002300
   1  C .        IND=3   NONLINEAR STIFFNESS MATRIX IS ASSEMBLED              . 00002400
   1  C .                FOR FREQUENCY ANALYSIS                               . 00002500
   1  C .        IND=4   EFFECTIVE NONLINEAR STIFFNESS MATRIX IS ASSEMBLED    . 00002600
   1  C .                AND EFFECTIVE LOAD VECTOR IS UPDATED                 . 00002700
   1  C .                                                                     . 00002800
   1  C . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 00002900
   7        COMMON/BBCCIO/KIO1,KIO2,KIO3,KIO4,KIO5,KIO6,KIO7,KIO8,KIO9,       00003000
   6       1KIO10,KIO11,KIO12,KIO13,KIO14,KIO15,KIO16,KIO17,KIO18,KIO19,KIO20,00003100
   6       2KIO21,KIO22,KIO23,KIO24,KIO25,KIO26,KIO27,KIO28,KIO29,KIO30       00003200
   1  C                                                                        00003300
   7        COMMON /SOL/ NUMNP,NEQ,NWK,NWM,NWC,NUMEST,MIDEST,MAXEST,NSTE,MA   00003400
   7        COMMON /DIM/ N1,N2,N3,N4,N5,N6,N7,N8,N9,N10,N11,N12,N13,N14,N15   00003500
   7        COMMON /EL/ IND,ICOUNT,NPAR(20),NUMEG,NEGL,NEGNL,IMASS,IDAMP,ISTAT00003600
   6       1        ,NDOF,KLIN,IEIG,IMASSN,IDAMPN                             00003700
   7        COMMON /VAR/ NG,KPRI,MODEX,KSTEP,ITE,ITEMAX,IREF,IEQREF,INOCMD    00003800
   7        COMMON /CONST/ DT,DTA,A0,A1,A2,A3,A4,A5,A6,A7,A8,A9,A10,A11       00003900
   6       1              ,A12,A13,A14,A15,A16,A17,A18,A19,A20,IOPE           00004000
   7        COMMON /TAPES/ NREAD,NWRITE                                       00004100
   7        COMMON /NORMS/ RNORM,RENORM,RTOL                                  00004200
   7        COMMON /DAMPNG/ ADAMP,BDAMP                                       00004300
   7        COMMON A(1)                                                       00004400
   1  C                                                                        00004500
   1  C                                                                        00004600
   1  C                                                                        00004700
   1  C     B E F O R E   T I M E   I N T E G R A T I O N                      00004800
   1  C                                                                        00004900
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8
   1  C                                                                      00005000
   1  C     ASSEMBLAGE AND STORAGE OF REQUIRED LINEAR MATRICES               00005100
   1  C                                                                      00005200
   7        GO TO (10,100,600,300),IND                                       00005300
   1  C                                                                      00005400
   1  C     LINEAR STIFFNESS MATRIX IS ASSEMBLED                             00005500
   1  C                                                                      00005600
   4     10 IF (NEGL.EQ.0) GO TO 100                                         00005700
   1  C                                                                      00005800
   7        ICOUNT=1                                                         00005900
   7        REWIND KIO11                                                     00006000
   7        DO 200 NG=1,NEGL                                                 00006100
   1  C                                                                      00006200
   7        READ (KIO11) NUMEST,NPAR,(A(I),I=1,NUMEST)                       00006300
   1  C                                                                      00006400
   7        CALL ELEMNT                                                      00006500
   1  C                                                                      00006600
   3    200 CONTINUE                                                        00006700
   1  C                                                                      00006800
   1  C     FOR CALCULATION OF FREQUENCIES AND MODE SHAPE VECTORS STORE THE  00006900
   1  C LINEAR STIFFNESS MATRIX ON TAPE 20                                   00007000
   1  C                                                                      00007100
   7        IF (IEIG.NE.1) GO TO 80                                          00007200
   7        REWIND KIO20                                                     00007300
   7        WRITE(KIO20) (A(I),I=N2,N3)                                      00007400
   1  C                                                                      00007500
   1  C     FOR NONLINEAR ANALYSIS OR FOR RAYLEIGH DAMPING                   00007600
   1  C    STORE LINEAR STIFFNESS ON TAPE 14                                 00007700
   1  C                                                                      00007800
   4     80 IF (KLIN.EQ.0 .AND. BDAMP.EQ.0.0) GO TO 90                       00007900
   7        WRITE (KIO14) (A(I),I=N2,N3)                                     00008000
   4     90 IF (BDAMP.EQ.0.0) GO TO 100                                      00008100
   1  C                                                                      00008200
   1  C     ADD STIFFNESS CONTRIBUTION OF RAYLEIGH DAMPING                   00008300
   1  C     TO THE EFFECTIVE STIFFNESS MATRIX                                00008400
   1  C                                                                      00008500
   7        FACT=1. + A1*BDAMP                                               00008600
   7        DO 95 I=N2,N3                                                    00008700
   4     95 A(I)=FACT*A(I)                                                   00008800
   1  C                                                                      00008900
   3    100 IF (ISTAT.EQ.0) RETURN                                          00009000
   1  C                                                                      00009100
   1  C     MASS AND DAMPING EFFECTS ARE ADDED TO STIFFNESS MATRIX           00009200
   1  C     OR MASS MATRIX IS ASSEMBLED                                      00009300
   1  C                                                                      00009400
   1  C     CONTRIBUTION FROM CONCENTRATED MASSES AND DAMPERS                00009500
   1  C                                                                      00009600
   7        IF (IMASSN.EQ.0 .AND. IDAMPN.EQ.0) GO TO 12                      00009700
   7        CALL ADDCM (A(N1),A(N2),A(N4),NEQ)                               00009800
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   1  C                                                                      00009900
   1  C     CONTRIBUTION FROM DISTRIBUTED MASSES (CONSISTENT MASS CASE)      00010000
   1  C                                                                      00010100
   4     12 IF (IND.GT.1 .AND. IMASS.EQ.1) GO TO 230                         00010200
   7        REWIND KIO11                                                    00010300
   7        REWIND KIO12                                                    00010400
   7        ICOUNT=2                                                        00010500
   7        NN=KIO11                                                        00010600
   1  C                                                                      00010700
   7        DO 220 NG=1,NUMEG                                               00010800
   7        IF (NG.GT.NEGL) NN=KIO12                                        00010900
   7        READ (NN) NUMEST,NPAR,(A(I),I=1,NUMEST)                         00011000
   1  C                                                                      00011100
   7        CALL ELEMNT                                                     00011200
   1  C                                                                      00011300
   3    220 CONTINUE                                                        00011400
   7        RETURN                                                          00011500
   1  C                                                                      00011600
   1  C     CONTRIBUTION FROM DISTRIBUTED MASSES (LUMPED MASS CASE)          00011700
   1  C                                                                      00011800
   3    230 DO 232 I=1,NEQ                                                  00011900
   3    232 A(N2+I-1)=A(N2+I-1) + A(N5+I-1)                                 00012000
   7        RETURN                                                          00012100
   1  C                                                                      00012200
   1  C                                                                      00012300
   1  C     THE LINEAR EFFECTIVE STIFFNESS MATRIX, THE MASS MATRIX, AND THE  00012400
   1  C     CONCENTRATED NODAL DAMPING VECTOR ARE STORED ON TAPE IN THE       00012500
   1  C     MAIN PROGRAM NONSAP                                              00012600
   1  C                                                                      00012700
   1  C                                                                      00012800
   1  C                                                                      00012900
   1  C                                                                      00013000
   1  C     D U R I N G   T I M E   I N T E G R A T I O N                    00013100
   1  C                                                                      00013200
   1  C                                                                      00013300
   1  C     EFFECTIVE LINEAR STIFFNESS MATRIX IS READ AND ELEMENT ROUTINES   00013400
   1  C     ARE CALLED TO UPDATE THE MATRIX AND THE EFFECTIVE LOAD VECTOR    00013500
   1  C     FOR NONLINEARITIES                                               00013600
   1  C                                                                      00013700
   3    300 IF (KLIN.EQ.0 .AND. INOCMD.EQ.0) RETURN                         00013800
   7        NN=N5 - 1                                                       00013900
   7        IF (KLIN.NE.0) GO TO 306                                        00014000
   7        READ(KIO17) (A(I),I=N4,NN)                                      00014100
   7        RETURN                                                          00014200
   3    306 NTAPE =KIO20                                                    00014300
   7        IF (IREF.EQ.0) NTAPE=KIO17                                      00014400
   7        READ (NTAPE) (A(I),I=N4,NN)                                     00014500
   1  C                                                                      00014600
   1  C     LOOP OVER ALL NONLINEAR ELEMENT GROUPS                           00014700
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8
   1  C                                                                      00014800
   7        REWIND NREAD                                                    00014900
   7        REWIND NWRITE                                                   00015000
   7        DO 320 NG=1,NEGNL                                               00015100
   7        READ (NREAD) NUMEST,NPAR,(A(I),I=1,NUMEST)                      00015200
   1  C                                                                     00015300
   7        CALL ELEMNT                                                     00015400
   1  C                                                                     00015500
   3     310 WRITE (NWRITE) NUMEST,NPAR,(A(I),I=1,NUMEST)                   00015600
   1  C                                                                     00015700
   3     320 CONTINUE                                                      00015800
   3     330 M=NWRITE                                                      00015900
   7        NWRITE=NREAD                                                   00016000
   7        NREAD=M                                                        00016100
   1  C                                                                     00016200
   1  C     CALCULATE NORM OF INCREMENTAL LOAD                              00016300
   1  C                                                                     00016400
   7        NN=N3 + NEQ - 1                                                00016500
   7        RNORM=0.                                                       00016600
   7        DO 380 I=N3,NN                                                 00016700
   3     380 RNORM=RNORM + A(I)*A(I)                                       00016800
   1  C                                                                     00016900
   7        RETURN                                                         00017000
   1  C                                                                     00017100
   1  C                                                                     00017200
   1  C     F R E Q U E N C Y   A N A L Y S I S                             00017300
   1  C                                                                     00017400
   1  C                                                                     00017500
   1  C     LINEAR STIFFNESS MATRIX IS READ FROM TAPE (NSTIF)               00017600
   1  C     AND IS UPDATED FOR NONLINEARITIES                               00017700
   1  C                                                                     00017800
   3     600 NSTIF = KIO20                                                 00017900
   7        IND=4                                                          00018000
   7        NN=N5 - 1                                                      00018100
   7        IF (NEGL.GT.0) GO TO 620                                       00018200
   7        DO 610 I=N4,NN                                                 00018300
   3     610 A(I)=0.0                                                      00018400
   7        GO TO 630                                                      00018500
   3     620 REWIND NSTIF                                                  00018600
   7        READ (NSTIF) (A(I),I=N4,NN)                                    00018700
   1  C                                                                     00018800
   1  C     LOOP OVER ALL NONLINEAR ELEMENTS                                00018900
   1  C                                                                     00019000
   3     630 IF (NEGNL.EQ.0) GO TO 650                                     00019100
   7        DO 635 I=1,NEQ                                                 00019200
   7        A(N2+I-1)=0.0                                                  00019300
   3     635 A(N3+I-1)=0.0                                                 00019400
   7        REWIND KIO18                                                   00019500
   7        READ (KIO18)                                                   00019600
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7        MM=N2 + NEQ - 1                                                  00019700
   7        READ(KIO18) (A(I),I=N2,MM)                                       00019800
   7        REWIND KIO12                                                     00019900
   7        DO 640 NG=1,NEGNL                                                00020000
   7        READ(KIO12) NUMEST,NPAR,(A(I),I=1,NUMEST)                        00020100
   1  C                                                                      00020200
   7        CALL ELEMNT                                                      00020300
   1  C                                                                      00020400
   3    640 CONTINUE                                                         00020500
   1  C                                                                      00020600
   7        REWIND NSTIF                                                     00020700
   7        WRITE (NSTIF) (A(I),I=N4,NN)                                     00020800
   1  C                                                                      00020900
   1  C     FIND FREQUENCIES AND MODE SHAPES                                 00021000
   1  C                                                                      00021100
   1  650   CALL FREQS                                                      00021200
   1  C                                                                      00021300
   7        RETURN                                                           00021400
   1  C                                                                      00021500
   1  C                                                                      00021600
   1  C                                                                      00021700
   7        END                                                              00021800
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7          SUBROUTINE BANDET(A,B,V,MAXA,NN,MA,NWA,NWB,RA,NSCH,DET,ISCALE,KK) 00000100
   7          IMPLICIT REAL*8(A-H,O-Z)                                          00000200
   7          COMMON/BBCCIO/KI01,KI02,KI03,KI04,KI05,KI06,KI07,KI08,KI09,       00000300
   6         1KI010,KI011,KI012,KI013,KI014,KI015,KI016,KI017,KI018,KI019,KI020,00000400
   6         2KI021,KI022,KI023,KI024,KI025,KI026,KI027,KI028,KI029,KI030       00000500
   1  C                                                                         00000600
   7          DIMENSION A(1),B(1),V(1),MAXA(1)                                  00000700
   7          COMMON /TAPEIG/ NSTIF,NT                                          00000800
   1  C                                                                         00000900
   7          MA1=MA - 1                                                        00001000
   7          NR=NN-1                                                           00001100
   7          IF (KK-2) 100,700,800                                             00001200
   1  C                                                                         00001300
   3      100 TOL=1.0E+07                                                       00001400
   7          RTOL=1.0E-10                                                      00001500
   1  C       SCALE=2.0**200                                                    00001600
   7          SCALE=2.D0**166                                                   00001700
   7          NTF=3                                                             00001800
   7          IS=1                                                              00001900
   3      120 REWIND NSTIF                                                      00002000
   7          READ(NSTIF) (A(I),I=1,NWA)                                        00002100
   7          IF(NWB.EQ.NN) GO TO 139                                           00002200
   7          DO 138 I=1,NWB                                                    00002300
   3      138 A(I)=A(I)-RA*B(I)                                                 00002400
   7          GO TO 160                                                         00002500
   3      139 DO 140 I=1,NN                                                     00002600
   7          II=MAXA(I)                                                        00002700
   3      140 A(II)=A(II)-RA*B(I)                                               00002800
   3      160 IF(NWA.EQ.NN) GO TO 231                                           00002900
   7          IF (A(1).EQ.0.0) A(1)=1.E-08                                      00003000
   7          DO 200 N=2,NN                                                     00003100
   7          KL=MAXA(N) + 1                                                    00003200
   7          KU=MAXA(N+1) - 1                                                  00003300
   7          IF (KU-KL) 200,210,210                                            00003400
   3      210 BBB=0.                                                            00003500
   1  C                                                                         00003600
   7          KN=MAXA(N)                                                        00003700
   7          K=N                                                               00003800
   7          DO 220 KKK=KL,KU                                                  00003900
   7          K=K - 1                                                           00004000
   7          KI=MAXA(K)                                                        00004100
   7          C=A(KKK)/A(KI)                                                    00004200
   7          IF (DABS(C).LT.TOL)GO TO 218                                      00004300
   7          GO TO 224                                                         00004400
   3      218 BBB=BBB + C*A(KKK)                                                00004500
   3      220 A(KKK)=C                                                          00004600
   7          A(KN)=A(KN)-BBB                                                   00004700
   1  C                                                                         00004800
   7          IF (A(KN)) 226,224,226                                            00004900
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   3    224 IS=IS + 1                                                    00005000
   7        IF (IS.LE.NTF) GO TO 260                                     00005100
   7        WRITE(KIO6,1000) NTF,RA                                      00005200
   7        STOP                                                         00005300
   3    260 RA=RA + (1.0 - RTOL)                                         00005400
   7        GO TO 120                                                    00005500
   3    226 MR=MIN0(MA1,NN-N)                                            00005600
   7        IF (MR) 200,200,228                                          00005700
   3    228 MN=KU - KL + 1                                               00005800
   1  C                                                                  00005900
   7        DO 240 J=1,MR                                                00006000
   7        MJ=MAXA(N+J) + J                                             00006100
   7        MNJ=MAXA(N+J+1) - MJ - 1                                     00006200
   7        IF (MNJ) 240,240,230                                         00006300
   3    230 ND=MIN0(MN,MNJ)                                              00006400
   7        C=0.                                                         00006500
   7        KU=KN + ND                                                   00006600
   7        IC=MJ - KN                                                   00006700
   7        DO 300 KKK=KL,KU                                             00006800
   3    300 C=C + A(KKK)*A(KKK+IC)                                       00006900
   7        A(KN+IC)=A(KN+IC) - C                                        00007000
   3    240 CONTINUE                                                     00007100
   1  C                                                                  00007200
   3    200 CONTINUE                                                     00007300
   3    231 N1=MAXA(NN)                                                  00007400
   7        IF(A(N1).NE.0.0) GO TO 280                                   00007500
   7        AA=DABS(A(1))                                                00007600
   7        DO 290 I=2,NR                                                00007700
   7        I1=MAXA(I)                                                   00007800
   3    290 AA=AA+DABS(A(I1))                                            00007900
   7        N1=MAXA(NN)                                                  00008000
   7        A(N1)=-(AA/NR)*1.0E-16                                       00008100
   1  C                                                                  00008200
   3    280 NSCH=0                                                       00008300
   7        ISC=0                                                        00008400
   7        DET=1.0                                                      00008500
   7        DO 301 I=1,NN                                                00008600
   7        IF (DABS(DET).LT.SCALE)GO TO 320                             00008700
   7        DET=DET/SCALE                                                00008800
   7        ISC=ISC+1                                                    00008900
   3    320 I1 = MAXA(I)                                                 00009000
   7        DET = DET*A(I1)                                              00009100
   3    301 IF(A(I1).LT.0.) NSCH=NSCH+1                                  00009200
   1  C                                                                  00009300
   7        IF (ISCALE.LT.1000) GO TO 340                                00009400
   7        ISCALE=ISC                                                   00009500
   7        GO TO 900                                                    00009600
   3    340 IF (ISC-ISCALE) 350,900,370                                  00009700
   3    350 DET=DET/SCALE                                                00009800
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7         GO TO 900                                                        00009900
   3     370 DET=DET*SCALE                                                    00010000
   7         GO TO 900                                                        00010100
   1   C                                                                      00010200
   3     700 DO 400 N=2,NN                                                    00010300
   7         KL=MAXA(N) + 1                                                   00010400
   7         KU=MAXA(N+1) - 1                                                 00010500
   7         IF (KU-KL) 400,410,410                                           00010600
   3     410 K=N                                                              00010700
   7         C=0.                                                             00010800
   7         DO 420 KKK=KL,KU                                                 00010900
   7         K=K - 1                                                          00011000
   3     420 C=C + A(KKK)*V(K)                                                00011100
   7         V(N)=V(N) - C                                                    00011200
   3     400 CONTINUE                                                         00011300
   1   C                                                                      00011400
   3     800 DO 480 N=1,NN                                                    00011500
   7         K=MAXA(N)                                                        00011600
   3     480 V(N)=V(N)/A(K)                                                   00011700
   7         IF (NN.EQ.1) RETURN                                              00011800
   7         N=NN                                                             00011900
   7         DO 500 L=2,NN                                                    00012000
   7         KL=MAXA(N) + 1                                                   00012100
   7         KU=MAXA(N+1) - 1                                                 00012200
   7         IF (KU-KL) 500,510,510                                           00012300
   3     510 K=N                                                              00012400
   7         DO 520 KKK=KL,KU                                                 00012500
   7         K=K - 1                                                          00012600
   3     520 V(K)=V(K) - A(KKK)*V(N)                                          00012700
   3     500 N=N - 1                                                          00012800
   3     900 RETURN                                                           00012900
   1   C                                                                      00013000
   2    1000 FORMAT (37H0***ERROR   SOLUTION STOP IN *BANDET*, / 12X,         00013100
   6       1        1H(,I3,37H) TRIANGULAR FACTORIZATIONS ATTEMPTED, / 12X,   00013200
   6       2        16HCURRENT SHIFT = ,E20.14 / 1X)                          00013300
   7         END                                                              00013400
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7        SUBROUTINE CAUCHY                                                      00000100
   7        IMPLICIT REAL*8(A-H,O-Z)                                               00000200
   1  C                                                                            00000300
   1  C                                                                            00000400
   1  C . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 00000500
   1  C .                                                                        . 00000600
   1  C .     CONVERTS PIOLA-KIRKHHOFF STRESSES                                  . 00000700
   1  C .          TO CAUCHY STRESSES                                            . 00000800
   1  C .                                                                        . 00000900
   1  C .     CS   =   (1./DET(F)) * (F * PK * F(TRANSPOSED))                    . 00001000
   1  C .                                                                        . 00001100
   1  C . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 00001200
   1  C                                                                            00001300
   7        COMMON/BBCCIO/KIO1,KIO2,KIO3,KIO4,KIO5,KIO6,KIO7,KIO8,KIO9,          00001400
   6       1KIO10,KIO11,KIO12,KIO13,KIO14,KIO15,KIO16,KIO17,KIO18,KIO19,KIO20,  00001500
   6       2KIO21,KIO22,KIO23,KIO24,KIO25,KIO26,KIO27,KIO28,KIO29,KIO30         00001600
   1  C                                                                            00001700
   7        COMMON /EL/ IND,ICOUNT,NPAR(20),NUMEG,NEGL,NEGNL,IMASS,IDAMP,ISTAT   00001800
   6       1           ,NDOF,KLIN,IEIG,IMASSN,IDAMPN                            00001900
   7        COMMON /VAR/ NG,KPRI,MODEX,KSTEP,ITE,ITEMAX,IREF,IEQREF,INOCMD       00002000
   7        COMMON /DISDER/ DISD(5)                                              00002100
   7        COMMON /MATMOD/ STRESS(4),STRAIN(4),D(4,4),IPT,NEL                   00002200
   7        EQUIVALENCE (NPAR(5),ITYP2D)                                         00002300
   1  C                                                                            00002400
   1  C                                                                            00002500
   7        IF (ITYP2D-1) 710,720,730                                            00002600
   3    710 X33=1.+DISD(5)                                                       00002700
   7        GO TO 750                                                            00002800
   3    720 X33=1.                                                               00002900
   7        GO TO 750                                                            00003000
   3    730 X33=DSQRT(1.+2.*STRAIN(4))                                           00003100
   1  C                                                                            00003200
   3    750 X11=1.+DISD(1)                                                       00003300
   7        X12=   DISD(3)                                                       00003400
   7        X22=1.+DISD(2)                                                       00003500
   7        X21=   DISD(4)                                                       00003600
   1  C                                                                            00003700
   7        DET=X33*(X11*X22 - X12*X21)                                          00003800
   7        IF (DET .GT. 0.) GO TO 760                                           00003900
   7        WRITE(KIO6,2100) NEL,KSTEP,DET                                       00004000
   7        STOP                                                                 00004100
   1  C                                                                            00004200
   3    760 DET=1./DET                                                           00004300
   7        S1=STRESS(1)                                                         00004400
   7        S2=STRESS(2)                                                         00004500
   7        SS=STRESS(3)                                                         00004600
   1  C                                                                            00004700
   7        STRESS(1)=DET * (S1*X11*X11 + 2.*SS*X11*X12 + S2*X12*X12)            00004800
   7        STRESS(2)=DET * (S1*X21*X21 + 2.*SS*X21*X22 + S2*X22*X22)            00004900
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7        STRESS(3)=DET * (S1*X11*X21 + SS*(X11*X22+X12*X21) + S2*X12*X22)  00005000
   7        IF (ITYP2D.EQ.2) RETURN                                           00005100
   7        STRESS(4)=DET * (STRESS(4)*X33*X33)                               00005200
   1  C                                                                       00005300
   7        RETURN                                                            00005400
   1  C                                                                       00005500
   1  C                                                                       00005600
   2   2100 FORMAT (40H DETERMINATE NOT POSITIVE                          /   00005700
   6       1        12H ELEMENT   =,I5/ 12H TIME STEP =,I5/ 8H DET    =,     00005800
   6       2        E14.6//38H STOP                                    )     00005900
   1  C                                                                       00006000
   7        END                                                               00006100
```

```
START                                                                                                                  MOD
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8                                FLAGS
   2   C     EKK=STRAIN ABCISSAE                                                                                    00000100
   2   C     KLD=LOADING BULK MODULUS                                                                               00000200
   2   C     KUN=UNLOADING BULK MODULUS                                                                             00000300
   2   C     GLD= LOADING SHEAR MODULUS                                                                             00000400
   2   C     SIG= STRESSES FROM THE PREVIOUS TIME STEP                                                              00000502   *
   2   C     EPS= STRAINS FROM THE PREVIOUS TIME STEP                                                               00000600
   2   C     EVMAX=MAXIMUM VOLUMETRIC STRAIN EVER REACHED(+COMPRESSIVE)                                             00000703   **
   2   C*    EVGRAV= VOLUMETRIC STRAIN DUE TO GRAVITY LOAD (S.N.MUKHERJEE,BBC)                                      00000800
   2   C*    PGRAV= PRESSURE DUE TO GRAVITY LOAD(S.N.MUKHERJEE,BBC,BADEN,AUG,                                       00000901   *
   2   C     1977.                                                                                                  00001001   *
   2   C     MODE. EQ. 5  CDMOD WITH TENSION CUT-OFF                                                                00001700
   7         SUBROUTINE CDMOD (NEL,EKK,KLD,KUN,GLD,SIG,EPS,EVMAX,EVGRAV,                                            00001800
   6        1                  PGRAV,ANGLE,STRESS,STRAIN,C,IPT)                                                    00001900
   7         IMPLICIT REAL*8(A-H,O-Z)                                                                               00002000
   7         COMMON/BBCCIO/KI01,KI02,KI03,KI04,KI05,KI06,KI07,KI08,KI09,                                            00002100
   6        1KI010,KI011,KI012,KI013,KI014,KI015,KI016,KI017,KI018,KI019,KI020,                                    00002200
   6        2KI021,KI022,KI023,KI024,KI025,KI026,KI027,KI028,KI029,KI030                                           00002300
   7         COMMON /EL/ IND,ICOUNT,NPAR(20),NUMEG,NEGL,NEGNL,IMASS,IDAMP,ISTAT                                     00002400
   6        1           ,NDOF,KLIN,IEIG,IMASSN,IDAMPN                                                              00002500
   7         COMMON /VAR/ NG,KPRI,MODEX,KSTEP,ITE,ITEMAX,IREF,IEQREF,INOCMD                                         00002600
   1   C                                                                                                            00002700
   7         DIMENSION EKK(1),GLD(1),SIG(1),EPS(1),STRESS(1),STRAIN(1),C(4,1)                                       00002800
   1   C                                                                                                            00002900
   7         REAL*8 KLD(1),KUN(1),K,KUNLO                                                                           00003000
   1   C                                                                                                            00003100
   7         EQUIVALENCE (NPAR(5),ITYP2D),(NPAR(17),NCON),(NPAR(10),NINT)                                           00003200
   6        1           ,(NPAR(15),MODEL)                                                                          00003300
   1   C                                                                                                            00003400
   1   C                                                                                                            00003500
   7         IF (ITYP2D.LT.2) GO TO 1                                                                               00003600
   7         WRITE(KI06,2005)                                                                                       00003700
   7         STOP                                                                                                   00003800
   5       1 NPT=NINT*NINT                                                                                          00003900
   7         IPOINT=NCON/4                                                                                          00004000
   1   C                                                                                                            00004100
   1   C                                                                                                            00004200
   1   C        1. CALCULATE STRESS AND STRAIN DEVIATORS                                                            00004300
   1   C           OF THE PREVIOUS TIME STEP                                                                        00004400
   1   C                                                                                                            00004500
   1   C                                                                                                            00004600
   7         TMM=(SIG(1)+SIG(2)+SIG(4))/3.                                                                          00004700
   7         T11=SIG(1) - TMM                                                                                       00004800
   7         T22=SIG(2) - TMM                                                                                       00004900
   7         T33=SIG(4) - TMM                                                                                       00005000
   7         T12=SIG(3)                                                                                             00005100
   1   C                                                                                                            00005200
   7         EVV=-(EPS(1)+EPS(2)+EPS(4))                                                                            00005300
   7         EMM=-EVV/3.                                                                                            00005400
   7         E11=EPS(1) - EMM                                                                                       00005500
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8  MOD FLAGS
```

```
   7         E22=EPS(2) - EMM                                                          00005600
   7         E33=EPS(4) - EMM                                                          00005700
   7         E12=EPS(3)                                                                00005800
   1  C                                                                                00005900
   1  C                                                                                00006000
   1  C        2. CALCULATE PARAMETERS BASED ON STRESSES AND STRAINS                   00006100
   1  C           OF THE PREVIOUS TIME STEP                                            00006200
   1  C                                                                                00006300
   1  C                                                                                00006400
   7         IKAS=1                                                                    00006500
   7         IF (EVV.LT.EVMAX) IKAS=-1                                                 00006600
   7         EVTOT=EVMAX + EVGRAV                                                      00006700
   7         IK=1                                                                      00006800
   7         ITEM=1                                                                    00006900
   1  C                                                                                00007000
   1  C                                                                                00007100
   4     10  DO 15 L=IK,IPOINT                                                        00007200
   7         J=L                                                                       00007300
   7         IF (EVTOT.LT.EKK(J)) GO TO 28                                             00007400
   4     15  CONTINUE                                                                 00007500
   7         WRITE(KIO6,2015) EVTOT,NEL,EKK(IPOINT)                                    00007600
   7         STOP                                                                      00007700
   1  C                                                                                00007800
   1  C                                                                                00007900
   4     28  I=J-1                                                                    00008000
   1  C                                                                                00008100
   7         DELEV=EKK(J) - EKK(I)                                                     00008200
   7         DELEI=EVTOT - EKK(I)                                                      00008300
   7         RATIO=DELEI / DELEV                                                       00008400
   1  C                                                                                00008500
   7         IF (IKAS) 30,35,35                                                        00008600
   1  C                                                                                00008700
   4     30  KUNLO=KUN(I) + RATIO * ( KUN(J) - KUN(I) )                               00008800
   4     35  K    =KLD(I) + RATIO * ( KLD(J) - KLD(I) )                               00008900
   7         G    =GLD(I) + RATIO * ( GLD(J) - GLD(I) )                               00009000
   1  C                                                                                00009100
   7         IF (IKAS) 40,45,45                                                        00009200
   1  C                                                                                00009300
   4     40  G=G * (KUNLO/K)                                                          00009400
   7         K=KUNLO                                                                   00009500
   1  C                                                                                00009600
   4     45  IF (ITEM.EQ.2) GO TO 100                                                 00009700
   1  C                                                                                00009800
   1  C                                                                                00009900
   1  C        3. IF CRACKING HAS OCCURRED, USE PARAMETERS CALCULATED IN (2.)          00010000
   1  C           WITH CRACKED MODEL TO FIND CURRENT STRESSES BASED ON                 00010100
   1  C           GIVEN STRAINS                                                        00010200
   1  C                                                                                00010300
   1  C                                                                                00010400
```

```
START                                                                    MOD
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8  FLAGS

   7        IF (MODEL.EQ.4) GO TO 50                                                  00010500
   7        IF (ANGLE.GT.361.) GO TO 50                                               00010600
   1  C                                                                               00010700
   1  C     FIND STRESSES PERPENDICULAR AND PARALLEL TO CRACK                         00010800
   1  C                                                                               00010900
   7        PI=4.*DATAN(1.0D0)                                                        00011000
   7        GAM=-2.*PI*(ANGLE/360.)                                                   00011100
   7        SG=DSIN(GAM)                                                              00011200
   7        CG=DCOS(GAM)                                                              00011300
   7        R11=CG*SIG(1) - SG*SIG(3)                                                 00011400
   7        R12=CG*SIG(3) - SG*SIG(2)                                                 00011500
   7        R21=SG*SIG(1) + CG*SIG(3)                                                 00011600
   7        R22=SG*SIG(3) + CG*SIG(2)                                                 00011700
   7        S11=R11*CG - R12*SG                                                       00011800
   7        S22=R21*SG + R22*CG                                                       00011900
   1  C                                                                               00012000
   1  C     TEST TO SEE WHICH DIRECTIONS HAVE OPEN CRACKS AT CURRENT STRESSES         00012100
   1  C                                                                               00012200
   7        IF (S11.LT.PGRAV .AND. S22.LT.PGRAV) NUMCRK=0                             00012300
   7        IF (S11.GE.PGRAV .AND. S22.LT.PGRAV) NUMCRK=1                             00012400
   7        IF (S11.LT.PGRAV .AND. S22.GE.PGRAV) NUMCRK=2                             00012500
   7        IF (S11.GE.PGRAV .AND. S22.GE.PGRAV) NUMCRK=3                             00012600
   1  C                                                                               00012700
   7        CALL DCRACK (C,ANGLE,K,G,NUMCRK)                                          00012800
   7        DO 56 I=1,4                                                               00012900
   7        STRESS(I)=0.                                                              00013000
   7        DO 56 J=1,4                                                               00013100
   4     56 STRESS(I)=STRESS(I) + C(I,J)*(STRAIN(J) - EPS(J))                         00013200
   7        DO 57 I=1,4                                                               00013300
   4     57 STRESS(I)=STRESS(I) + SIG(I)                                              00013400
   7        EVI=-(STRAIN(1) + STRAIN(2) + STRAIN(4))                                  00013500
   7        GO TO 65                                                                  00013600
   1  C                                                                               00013700
   1  C                                                                               00013800
   1  C        4. IF THERE IS NO CRACKING ON PREVIOUS TIME STEP, USE ISOTROPIC        00013900
   1  C           LAW WITH PARAMETERS CALCULATED IN (2.) TO FIND CURRENT             00014000
   1  C           STRESSES BASED ON GIVEN STRAINS                                    00014100
   1  C                                                                               00014200
   1  C                                                                               00014300
   4     50 EVI=-(STRAIN(1)+STRAIN(2)+STRAIN(4))                                      00014400
   7        EPSMM=-EVI/3.                                                             00014500
   7        EPS11=STRAIN(1)-EPSMM                                                     00014600
   7        EPS22=STRAIN(2)-EPSMM                                                     00014700
   7        EPS33=STRAIN(4)-EPSMM                                                     00014800
   7        EPS12=STRAIN(3)                                                           00014900
   1  C                                                                               00015000
   7        PEE = 3. * K * (EPSMM-EMM) + TMM                                          00015100
   1  C                                                                               00015200
   7        S11=2.*G*(EPS11-E11)+T11                                                  00015300
```

```
START                                                                                                  MOD
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8               FLAGS
    7         S22=2.*G*(EPS22-E22)+T22                                                    00015400
    7         S33=2.*G*(EPS33-E33)+T33                                                    00015500
    7         S12=   G*(EPS12-E12)+T12                                                    00015600
    1  C                                                                                  00015700
    1  C                                                                                  00015800
    7         STRESS(1)=S11+PEE                                                           00015900
    7         STRESS(2)=S22+PEE                                                           00016000
    7         STRESS(4)=S33+PEE                                                           00016100
    7         STRESS(3)=S12                                                               00016200
    1  C                                                                                  00016300
    1  C                                                                                  00016400
    1  C        5. DETERMINE WHETHER CURRENT STRESSES HAVE CAUSED CRACKING                00016500
    1  C           TO OCCUR                                                               00016600
    1  C                                                                                  00016700
    1  C                                                                                  00016800
    7         IF (MODEL.EQ.4) GO TO 65                                                    00016900
    7         IF (ANGLE.LT.361. .OR. KPRI.EQ.0) GO TO 65                                  00017000
    7         CALL MAXMIN (STRESS,P1,P2,ANG)                                              00017100
    7         PTOT=-PGRAV + P1                                                            00017200
    7         IF (PTOT.GT.0.0) ANGLE=ANG                                                  00017300
    1  C                                                                                  00017400
    1  C                                                                                  00017500
    1  C        6. PRINT STRESSES                                                         00017600
    1  C                                                                                  00017700
    1  C                                                                                  00017800
    4     65 IF (KPRI.NE.0) GO TO 70                                                     00017900
    7        IF (IPT.NE.1) GO TO 66                                                      00018000
    7        WRITE(KIO6,2000)                                                            00018100
    7        WRITE(KIO6,2035) NEL                                                        00018200
    4     66 CALL MAXMIN (STRESS,P1,P2,AG)                                               00018300
    7        IF (MODEL.EQ.4) GO TO 67                                                    00018400
    1  C                                                                                  00018500
    1  C     FOR PRINTING PURPOSES ONLY, FIND WHAT TYPE OF CRACKING THESE                 00018600
    1  C     STRESSES GIVE RISE TO                                                       00018700
    1  C                                                                                  00018800
    7         IF (ANGLE.LT.361.0) GO TO 165                                               00018900
    1  C                                                                                  00019000
    1  C     TEST TO SEE IF CRACKING OCCURS FOR THE FIRST TIME                           00019100
    1  C     AT THIS PRINTING STRESS LEVEL                                               00019200
    1  C                                                                                  00019300
    7         ANGPRI=1000.                                                                00019400
    7         CALL MAXMIN (STRESS,P1,P2,ANG)                                              00019500
    7         PTOT=-PGRAV + P1                                                            00019600
    7         IF (PTOT.GT.0.0) ANGPRI=ANG                                                 00019700
    7         IF (ANGPRI.GT.361.0) GO TO 67                                               00019800
    7         GO TO 166                                                                   00019900
    1  C                                                                                  00020000
    1  C     TEST FOR SECOND CRACK OR CRACK CLOSING                                      00020100
    1  C     AT THIS PRINTING STRESS LEVEL                                               00020200
```

```
START                                                                                MOD
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8  FLAGS

   1  C                                                                              00020300
   3    165 ANGPRI=ANGLE                                                            00020400
   3    166 PI=4.0*DATAN(1.0D0)                                                     00020500
   7        GAM=-2.*PI*(ANGPRI/360.)                                                00020600
   7        SG=DSIN(GAM)                                                            00020700
   7        CG=DCOS(GAM)                                                            00020800
   7        R11=CG*STRESS(1) - SG*STRESS(3)                                         00020900
   7        R12=CG*STRESS(3) - SG*STRESS(2)                                         00021000
   7        R21=SG*STRESS(1) + CG*STRESS(3)                                         00021100
   7        R22=SG*STRESS(3) + CG*STRESS(2)                                         00021200
   7        S11=R11*CG - R12*SG                                                     00021300
   7        S22=R21*SG + R22*CG                                                     00021400
   7        IF (S11.LT.PGRAV .AND. S22.LT.PGRAV) NUMCRK=0                           00021500
   7        IF (S11.GE.PGRAV .AND. S22.LT.PGRAV) NUMCRK=1                           00021600
   7        IF (S11.LT.PGRAV .AND. S22.GE.PGRAV) NUMCRK=2                           00021700
   7        IF (S11.GE.PGRAV .AND. S22.GE.PGRAV) NUMCRK=3                           00021800
   1  C                                                                             00021900
   7        NF=NUMCRK + 1                                                           00022000
   7        GO TO (61,62,63,64) ,NF                                                 00022100
   4     61 WRITE(KIO6,2042)IPT,(STRESS(I),I=1,4),P1,P2,AG,PGRAV,ANGPRI             00022200
   7        GO TO 70                                                                00022300
   4     62 WRITE(KIO6,2040)IPT,(STRESS(I),I=1,4),P1,P2,AG,PGRAV,ANGPRI             00022400
   7        GO TO 70                                                                00022500
   4     63 WRITE(KIO6,2043)IPT,(STRESS(I),I=1,4),P1,P2,AG,PGRAV,ANGPRI             00022600
   7        GO TO 70                                                                00022700
   4     64 WRITE(KIO6,2044)IPT,(STRESS(I),I=1,4),P1,P2,AG,PGRAV,ANGPRI             00022800
   7        GO TO 70                                                                00022900
   4     67 WRITE(KIO6,2041)IPT,(STRESS(I),I=1,4),P1,P2,AG,PGRAV                    00023000
   1  C                                                                             00023100
   4     70 IF (ITE.GE.1 .OR. KPRI.EQ.0) RETURN                                     00023200
   1  C                                                                             00023300
   1  C                                                                             00023400
   1  C        7. UPDATE STRESSES AND STRAINS                                       00023500
   1  C                                                                             00023600
   1  C                                                                             00023700
   7        SIG(1)=STRESS(1)                                                        00023800
   7        SIG(2)=STRESS(2)                                                        00023900
   7        SIG(3)=STRESS(3)                                                        00024000
   7        SIG(4)=STRESS(4)                                                        00024100
   1  C                                                                             00024200
   7        EPS(1)=STRAIN(1)                                                        00024300
   7        EPS(2)=STRAIN(2)                                                        00024400
   7        EPS(3)=STRAIN(3)                                                        00024500
   7        EPS(4)=STRAIN(4)                                                        00024600
   1  C                                                                             00024700
   7        IF (EVI.GT.EVMAX) EVMAX=EVI                                             00024800
   1  C                                                                             00024900
   1  C                                                                             00025000
   1  C        8. FORM THE STRESS-STRAIN RELATIONSHIP                               00025100
```

```
START                                                                                MOD
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8  FLAGS
   1  C                                                                              00025200
   1  C                                                                              00025300
   7        IF (EVI.LT.EVMAX) GO TO 75                                               00025400
   1  C                                                                              00025500
   1  C     L O A D I N G : EFFECT OF GRAVITY LOAD IS ADDED (S.N.MUKHERJEE,           00025600
   1  C     BBC,BADEN 1979)                                                          00025700
   7        EVTOT=EVMAX + EVGRAV                                                     00025800
   7        IK=J                                                                     00025900
   7        IKAS=1                                                                   00026000
   7        ITEM=2                                                                   00026100
   7        GO TO 10                                                                 00026200
   1  C                                                                              00026300
   4     75 IF (IKAS) 100,85,85                                                      00026400
   1  C                                                                              00026500
   1  C     W A S   L O A D I N G,   B U T   N O W   U N L O A D I N G               00026600
   1  C                                                                              00026700
   4     85 KUNLO=KUN(I) + RATIO * ( KUN(J) - KUN(I) )                               00026800
   7        G=G * (KUNLO/K)                                                          00026900
   7        K=KUNLO                                                                  00027000
   1  C                                                                              00027100
   1  C     W A S   U N L O A D I N G,   A N D   N O W   U N L O A D I N G           00027200
   1  C                                                                              00027300
   3    100 IF (MODEL.EQ.4) GO TO 101                                               00027400
   7        IF (ANGLE.LT.361.) GO TO 110                                             00027500
   1  C                                                                              00027600
   1  C     ISOTROPIC STRESS-STRAIN LAW                                              00027700
   1  C                                                                              00027800
   3    101 DUM=0.6666667 * G                                                       00027900
   7        A1=K + 2.*DUM                                                            00028000
   7        B1=K - DUM                                                               00028100
   1  C                                                                              00028200
   7        C(1,1)=A1                                                                00028300
   7        C(1,2)=B1                                                                00028400
   7        C(1,3)=0.                                                                00028500
   7        C(2,1)=B1                                                                00028600
   7        C(2,2)=A1                                                                00028700
   7        C(2,3)=0.                                                                00028800
   7        C(3,1)=0.                                                                00028900
   7        C(3,2)=0.                                                                00029000
   7        C(3,3)=G                                                                 00029100
   1  C                                                                              00029200
   7        IF (ITYP2D.EQ.1) RETURN                                                  00029300
   1  C                                                                              00029400
   7        C(1,4)=B1                                                                00029500
   7        C(2,4)=B1                                                                00029600
   7        C(3,4)=0.                                                                00029700
   7        C(4,1)=B1                                                                00029800
   7        C(4,2)=B1                                                                00029900
   7        C(4,3)=0.                                                                00030000
```

```
START                                                                                MOD
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8  FLAGS

   7        C(4,4)=A1                                                                00030100
   1  C                                                                              00030200
   7        RETURN                                                                   00030300
   1  C                                                                              00030400
   1  C     CRACKED STRESS-STRAIN LAW                                                00030500
   1  C                                                                              00030600
   3    110 PI=4.0*DATAN(1.0D0)                                                      00030700
   7        GAM=-2.*PI*(ANGLE/360.)                                                  00030800
   7        SG=DSIN(GAM)                                                             00030900
   7        CG=DCOS(GAM)                                                             00031000
   7        R11=CG*STRESS(1) - SG*STRESS(3)                                          00031100
   7        R12=CG*STRESS(3) - SG*STRESS(2)                                          00031200
   7        R21=SG*STRESS(1) + CG*STRESS(3)                                          00031300
   7        R22=SG*STRESS(3) + CG*STRESS(2)                                          00031400
   7        S11=R11*CG - R12*SG                                                      00031500
   7        S22=R21*SG + R22*CG                                                      00031600
   7        IF (S11.LT.PGRAV .AND. S22.LT.PGRAV) NUMCRK=0                            00031700
   7        IF (S11.GE.PGRAV .AND. S22.LT.PGRAV) NUMCRK=1                            00031800
   7        IF (S11.LT.PGRAV .AND. S22.GE.PGRAV) NUMCRK=2                            00031900
   7        IF (S11.GE.PGRAV .AND. S22.GE.PGRAV) NUMCRK=3                            00032000
   7        CALL DCRACK (C,ANGLE,K,G,NUMCRK)                                         00032100
   1  C                                                                              00032200
   7        RETURN                                                                   00032300
   1  C                                                                              00032400
   1  C                                                                              00032500
   2   2000 FORMAT (///24H  ELEMENT   INTEGRATION /19H  NUMBER      POINT ,4X,      00032600
   6       1        50H  SIGMA-X1  SIGMA-X2   TAU-X12  SIGMA-X3  SIGMA-P+,         00032700
   6       2        40H  SIGMA-P-  ANGLE     PGRAV  CRACK ANGLE /)                  00032800
   2   2005 FORMAT (49H **STOP - PLANE STRESS CANNOT BE USED WITH CDMOD  )          00032900
   2   2015 FORMAT (///36H ***ERROR  CURRENT VOLUMETRIC STRAIN,E14.6,2X,            00033000
   6       1        13HFOR ELEMENT (,I2,1H)/11X,21HEXCEEDS TABLE MAXIMUM,          00033100
   6       2        E14.6//8H ***STOP)                                             00033200
   2   2035 FORMAT (I9)                                                           00033300
   2   2040 FORMAT (9X,I10,4X,6E10.3,F7.2,E10.3,F11.2,16H (ONLY 1ST OPEN)   )     00033400
   2   2041 FORMAT (9X,I10,4X,6E10.3,F7.2,E10.3,13H  NO CRACKING  )               00033500
   2   2042 FORMAT (9X,I10,4X,6E10.3,F7.2,E10.3,F11.2,10H  (CLOSED)  )            00033600
   2   2043 FORMAT (9X,I10,4X,6E10.3,F7.2,E10.3,F11.2,16H (ONLY 2ND OPEN)  )      00033700
   2   2044 FORMAT (9X,I10,4X,6E10.3,F7.2,E10.3,F11.2,12H  (2 CRACKS)  )          00033800
   1  C                                                                          00033900
   7        END                                                                  00034000
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7        SUBROUTINE CMOD3D (NEL,EKK,KLD,KUN,GLD,SIG,EPS,EVMAX,              00000100
   6       1                   STRESS,STRAIN,C,IPT)                            00000200
   7        IMPLICIT REAL*8(A-H,O-Z)                                          00000300
   1  C                                                                       00000400
   1  C                                                                       00000500
   1  C . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .  00000600
   1  C                                                                       00000700
   1  C        EKK       STRAIN ABCISSAE                                      00000800
   1  C        KLD       LOADING BULKS MODULUS                                00000900
   1  C        KUN       UNLOADING BULKS MODULUS                              00001000
   1  C        GLD       LOADING SHEAR MODULUS                                00001100
   1  C                                                                       00001200
   1  C        SIG       STRESSES FROM THE PREVIOUS TIME STEP                 00001300
   1  C        EPS       STRAINS  FROM THE PREVIOUS TIME STEP                 00001400
   1  C        EVMAX     MAXIMUM VOLUMETRIC STRAIN EVER REACHED (COMP +)      00001500
   1  C                                                                       00001600
   1  C        STRESS    CURRENT STRESSES                                     00001700
   1  C        STRAIN    CURRENT STRAINS                                      00001800
   1  C        C         CURRENT ELASTICITY MODULUS MATRIX                    00001900
   1  C        IPT       INTEGRATION POINT INDICATOR                          00002000
   1  C . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .  00002100
   1  C                                                                       00002200
   1  C                                                                       00002300
   1  C                                                                       00002400
   7        COMMON/BBCCIO/KIO1,KIO2,KIO3,KIO4,KIO5,KIO6,KIO7,KIO8,KIO9,       00002500
   6       1KIO10,KIO11,KIO12,KIO13,KIO14,KIO15,KIO16,KIO17,KIO18,KIO19,KIO20,00002600
   6       2KIO21,KIO22,KIO23,KIO24,KIO25,KIO26,KIO27,KIO28,KIO29,KIO30       00002700
   7        COMMON /EL/ IND,ICOUNT,NPAR(20),NUMEG,NEGL,NEGNL,IMASS,IDAMP,ISTAT00002800
   6       1           ,NDOF,KLIN,IEIG,IMASSN,IDAMPN                          00002900
   7        COMMON /VAR/ NG,KPRI,MODEX,KSTEP,ITE,ITEMAX,IREF,IEQREF,INOCMD    00003000
   1  C                                                                       00003100
   7        DIMENSION EKK(1),GLD(1),SIG(1),EPS(1),STRESS(1),STRAIN(1),C(6,1)  00003200
   1  C                                                                       00003300
   7        REAL*8 KLD(1),KUN(1),K,KUNLO                                      00003400
   1  C                                                                       00003500
   7        EQUIVALENCE (NPAR(10),NINT),(NPAR(11),NINTZ),(NPAR(17),NCON)      00003600
   1  C                                                                       00003700
   1  C                                                                       00003800
   7        NPT=NINT*NINT*NINTZ                                               00003900
   7        IPOINT=NCON/4                                                     00004000
   1  C                                                                       00004100
   1  C                                                                       00004200
   1  C        1. CALCULATE STRESS AND STRAIN DEVIATORS                       00004300
   1  C           OF THE PREVIOUS TIME STEP                                   00004400
   1  C                                                                       00004500
   1  C                                                                       00004600
   7        TMM=(SIG(1)+SIG(2)+SIG(3))/3.                                     00004700
   7        T11=SIG(1) - TMM                                                  00004800
   7        T22=SIG(2) - TMM                                                  00004900
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7        T33=SIG(3) - TMM                                                  00005000
   7        T12=SIG(4)                                                        00005100
   7        T13=SIG(5)                                                        00005200
   7        T23=SIG(6)                                                        00005300
   1  C                                                                       00005400
   7        EVV=-(EPS(1)+EPS(2)+EPS(3))                                       00005500
   7        EMM=-EVV/3.                                                       00005600
   7        E11=EPS(1) - EMM                                                  00005700
   7        E22=EPS(2) - EMM                                                  00005800
   7        E33=EPS(3) - EMM                                                  00005900
   7        E12=EPS(4)                                                        00006000
   7        E13=EPS(5)                                                        00006100
   7        E23=EPS(6)                                                        00006200
   1  C                                                                       00006300
   1  C                                                                       00006400
   1  C        2. CALCULATE PARAMETERS BASED ON STRESSES AND STRAINS          00006500
   1  C           OF THE PREVIOUS TIME STEP                                   00006600
   1  C                                                                       00006700
   1  C                                                                       00006800
   7        IKAS=1                                                            00006900
   7        IF (EVV.LT.EVMAX) IKAS=-1                                         00007000
   7        EVTOT=EVMAX                                                       00007100
   7        IK=1                                                              00007200
   7        ITEM=1                                                            00007300
   1  C                                                                       00007400
   1  C                                                                       00007500
   4     10 DO 15 L=IK,IPOINT                                                 00007600
   7        J=L                                                               00007700
   7        IF (EVTOT.LT.EKK(J)) GO TO 28                                     00007800
   4     15 CONTINUE                                                          00007900
   7        WRITE(KIO6,2015) EVTOT,NEL,EKK(IPOINT)                            00008000
   7        STOP                                                              00008100
   1  C                                                                       00008200
   1  C                                                                       00008300
   4     28 I=J-1                                                             00008400
   1  C                                                                       00008500
   7        DELEV=EKK(J) - EKK(I)                                             00008600
   7        DELEI=EVTOT - EKK(I)                                              00008700
   7        RATIO=DELEI / DELEV                                               00008800
   1  C                                                                       00008900
   7        IF (IKAS) 30,35,35                                                00009000
   1  C                                                                       00009100
   4     30 KUNLO=KUN(I) + RATIO * ( KUN(J) - KUN(I) )                        00009200
   4     35 K    =KLD(I) + RATIO * ( KLD(J) - KLD(I) )                        00009300
   7        G    =GLD(I) + RATIO * ( GLD(J) - GLD(I) )                        00009400
   1  C                                                                       00009500
   7        IF (IKAS) 40,45,45                                                00009600
   1  C                                                                       00009700
   4     40 G=G * (KUNLO/K)                                                   00009800
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7        K=KUNLO                                                          00009900
   1  C                                                                      00010000
   4     45 IF (ITEM.EQ.2) GO TO 100                                         00010100
   1  C                                                                      00010200
   1  C        3. USING PARAMETERS CALCULATED IN (2.), FIND CURRENT STRESSES 00010300
   1  C           BASED ON GIVEN STRAINS                                     00010400
   1  C                                                                      00010500
   1  C                                                                      00010600
   7        EVI=-(STRAIN(1)+STRAIN(2)+STRAIN(3))                             00010700
   7        EPSMM=-EVI/3.                                                    00010800
   7        EPS11=STRAIN(1)-EPSMM                                            00010900
   7        EPS22=STRAIN(2)-EPSMM                                            00011000
   7        EPS33=STRAIN(3)-EPSMM                                            00011100
   7        EPS12=STRAIN(4)                                                  00011200
   7        EPS13=STRAIN(5)                                                  00011300
   7        EPS23=STRAIN(6)                                                  00011400
   1  C                                                                      00011500
   7        PEE = 3. * K * (EPSMM-EMM) + TMM                                 00011600
   1  C                                                                      00011700
   7        S11=2.*G*(EPS11-E11)+T11                                         00011800
   7        S22=2.*G*(EPS22-E22)+T22                                         00011900
   7        S33=2.*G*(EPS33-E33)+T33                                         00012000
   7        S12=   G*(EPS12-E12)+T12                                         00012100
   7        S13=   G*(EPS13-E13)+T13                                         00012200
   7        S23=   G*(EPS23-E23)+T23                                         00012300
   1  C                                                                      00012400
   1  C                                                                      00012500
   7        STRESS(1)=S11+PEE                                                00012600
   7        STRESS(2)=S22+PEE                                                00012700
   7        STRESS(3)=S33+PEE                                                00012800
   7        STRESS(4)=S12                                                    00012900
   7        STRESS(5)=S13                                                    00013000
   7        STRESS(6)=S23                                                    00013100
   1  C                                                                      00013200
   1  C     PRINT STRESSES                                                   00013300
   1  C                                                                      00013400
   7        IF (KPRI.NE.0) GO TO 50                                          00013500
   7        IF (IPT.NE.1) GO TO 55                                           00013600
   7        WRITE(KIO6,2030)                                                 00013700
   7        WRITE(KIO6,2035) NEL                                             00013800
   4     55 WRITE(KIO6,2040) IPT,(STRESS(I),I=1,6)                           00013900
   1  C                                                                      00014000
   4     50 IF (ITE.GE.1 .OR. KPRI.EQ.0) RETURN                              00014100
   1  C                                                                      00014200
   1  C        4. UPDATE STRESSES AND STRAINS                                00014300
   1  C                                                                      00014400
   7        SIG(1)=STRESS(1)                                                 00014500
   7        SIG(2)=STRESS(2)                                                 00014600
   7        SIG(3)=STRESS(3)                                                 00014700
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7          SIG(4)=STRESS(4)                                              00014800
   7          SIG(5)=STRESS(5)                                              00014900
   7          SIG(6)=STRESS(6)                                              00015000
   1  C                                                                     00015100
   7          EPS(1)=STRAIN(1)                                              00015200
   7          EPS(2)=STRAIN(2)                                              00015300
   7          EPS(3)=STRAIN(3)                                              00015400
   7          EPS(4)=STRAIN(4)                                              00015500
   7          EPS(5)=STRAIN(5)                                              00015600
   7          EPS(6)=STRAIN(6)                                              00015700
   1  C                                                                     00015800
   7          IF (EVI.GT.EVMAX) EVMAX=EVI                                   00015900
   1  C                                                                     00016000
   1  C                                                                     00016100
   7          IF (IPT.EQ.NPT) RETURN                                        00016200
   1  C                                                                     00016300
   1  C                                                                     00016400
   1  C          5. FORM THE STRESS-STRAIN RELATIONSHIP                     00016500
   1  C                                                                     00016600
   1  C                                                                     00016700
   7          IF (EVI.LT.EVMAX) GO TO 75                                    00016800
   1  C                                                                     00016900
   1  C     L O A D I N G                                                   00017000
   1  C                                                                     00017100
   7          EVTOT=EVMAX                                                   00017200
   7          IK=J                                                          00017300
   7          IKAS=1                                                        00017400
   7          ITEM=2                                                        00017500
   7          GO TO 10                                                      00017600
   1  C                                                                     00017700
   4     75 IF (IKAS) 100,85,85                                             00017800
   1  C                                                                     00017900
   1  C     W A S   L O A D I N G,   B U T   N O W   U N L O A D I N G      00018000
   1  C                                                                     00018100
   4     85 KUNLO=KUN(I) + RATIO * ( KUN(J) - KUN(I) )                      00018200
   7        G=G * (KUNLO/K)                                                00018300
   7        K=KUNLO                                                        00018400
   1  C                                                                     00018500
   1  C     W A S   U N L O A D I N G,   A N D   N O W   U N L O A D I N G  00018600
   1  C                                                                     00018700
   3    100 CONTINUE                                                       00018800
   1  C                                                                     00018900
   1  C                                                                     00019000
   7          DUM=0.6666667 * G                                           00019100
   7          A1=K + 2.*DUM                                               00019200
   7          B1=K - DUM                                                  00019300
   1  C                                                                     00019400
   7          DO 110 I=1,6                                                00019500
   7          DO 110 J=1,6                                                00019600
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   3    110 C(I,J)=0.0                                                        00019700
   1  C                                                                       00019800
   7        C(1,1)=A1                                                       00019900
   7        C(1,2)=B1                                                       00020000
   7        C(1,3)=B1                                                       00020100
   7        C(2,1)=B1                                                       00020200
   7        C(2,2)=A1                                                       00020300
   7        C(2,3)=B1                                                       00020400
   7        C(3,1)=B1                                                       00020500
   7        C(3,2)=B1                                                       00020600
   7        C(3,3)=A1                                                       00020700
   7        DO 120 I=4,6                                                    00020800
   3    120 C(I,I)=G                                                        00020900
   1  C                                                                       00021000
   7        RETURN                                                          00021100
   1  C                                                                       00021200
   1  C                                                                       00021300
   1  C                                                                       00021400
   2   2015 FORMAT (///36H ***ERROR  CURRENT VOLUMETRIC STRAIN,E14.6,2X,      00021500
   6       1        13HFOR ELEMENT (,I2,1H)/11X,21HEXCEEDS TABLE MAXIMUM,    00021600
   6       2        E14.6//8H ***STOP)                                       00021700
   2   2030 FORMAT (/                                                        00021800
   6       1        8H ELEMENT,4X,6HOUTPUT,/ 2X,6HNUMBER,2X,8HLOCATION,7X,   00021900
   6       1        8HSIGMA-X1,7X,8HSIGMA-X2,7X,8HSIGMA-X3,8X,7HTAU-X12,     00022000
   6       2        8X,7HTAU-X13,8X,7HTAU-X23 / 1X)                          00022100
   2   2035 FORMAT (I8)                                                      00022200
   2   2040 FORMAT (13X,I5,6F15.4)                                           00022300
   1  C                                                                       00022400
   7        END                                                             00022500
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7        SUBROUTINE COLHT (MHT,ND,LM)                                          00000100
   7        IMPLICIT REAL*8(A-H,O-Z)                                              00000200
   1  C                                                                           00000300
   7        COMMON /SOL/ NUMNP,NEQ,NWK,NWM,NWC,NUMEST,MIDEST,MAXEST,NSTE,MA       00000400
   7        DIMENSION LM(1),MHT(1)                                                00000500
   1  C                                                                           00000600
   7        LS=100000                                                             00000700
   7        DO 100 I=1,ND                                                         00000800
   7        IF (LM(I)) 110,100,110                                                00000900
   3    110 IF (LM(I)-LS) 120,100,100                                             00001000
   3    120 LS=LM(I)                                                              00001100
   3    100 CONTINUE                                                              00001200
   1  C                                                                           00001300
   7        DO 200 I=1,ND                                                         00001400
   7        II=LM(I)                                                              00001500
   7        IF (II.EQ.0) GO TO 200                                                00001600
   7        ME=II - LS                                                            00001700
   7        IF (ME.GT.MHT(II)) MHT(II)=ME                                         00001800
   2    200 CONTINUE                                                              00001900
   1  C                                                                           00002000
   7        RETURN                                                                00002100
   7        END                                                                   00002200
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7        SUBROUTINE COLSOL (A,V,MAXA,NN,MA,NWA,KKK)                          00000100
   7        IMPLICIT REAL*8(A-H,O-Z)                                            00000200
   1  C                                                                         00000300
   1  C . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 00000400
   1  C .                                                                     . 00000500
   1  C .   P R O G R A M                                                     . 00000600
   1  C .      . TO SOLVE SIMULTANEOUS EQUATIONS IN CORE, USING COMPACTED     . 00000700
   1  C .        STORAGE AND COLUMN REDUCTION SCHEME                          . 00000800
   1  C .                                                                     . 00000900
   1  C .        A    = MATRIX STORED IN COMPACTED FORM                       . 00001000
   1  C .        V    = VECTOR TO BE REDUCED                                  . 00001100
   1  C .        MAXA = VECTOR CONTAINING ADDRESSES OF DIAGONAL ELEMENTS      . 00001200
   1  C .               OF A                                                  . 00001300
   1  C .                                                                     . 00001400
   1  C . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 00001500
   1  C                                                                         00001600
   7        COMMON/BBCCIO/KIO1,KIO2,KIO3,KIO4,KIO5,KIO6,KIO7,KIO8,KIO9,       00001700
   6       1KIO10,KIO11,KIO12,KIO13,KIO14,KIO15,KIO16,KIO17,KIO18,KIO19,KIO20,00001800
   6       2KIO21,KIO22,KIO23,KIO24,KIO25,KIO26,KIO27,KIO28,KIO29,KIO30       00001900
   7        COMMON /EL/ IND,ICOUNT,NPAR(20),NUMEG,NEGL,NEGNL,IMASS,IDAMP,ISTAT00002000
   6       1           ,NDOF,KLIN,IEIG,IMASSN,IDAMPN                          00002100
   7        DIMENSION A(NWA),V(1),MAXA(1)                                      00002200
   1  C                                                                       00002300
   7        IF (NN.EQ.1) GO TO 799                                             00002400
   7        MA1=MA - 1                                                         00002500
   7        IF (KKK-2) 100,700,800                                             00002600
   1  C                                                                       00002700
   3    100 N=1                                                              00002800
   7        IF (A(1)) 80,85,110                                                00002900
   4     80 WRITE(KIO6,1000) N                                                 00003000
   7        STOP                                                               00003100
   4     85 WRITE(KIO6,1000) N                                                 00003200
   7        STOP                                                               00003300
   1  C                                                                       00003400
   3    110 DO 200 N=2,NN                                                    00003500
   7        KL=MAXA(N) + 1                                                     00003600
   7        KU=MAXA(N+1) - 1                                                   00003700
   7        IF (KU-KL) 200,210,210                                             00003800
   2    210 B=0.                                                             00003900
   1  C                                                                       00004000
   7        KN=MAXA(N)                                                         00004100
   7        K=N                                                                00004200
   7        DO 220 KK=KL,KU                                                    00004300
   7        K=K - 1                                                            00004400
   7        KI=MAXA(K)                                                         00004500
   7        C=A(KK)/A(KI)                                                      00004600
   7        B=B + C*A(KK)                                                      00004700
   2    220 A(KK)=C                                                          00004800
   7        A(KN)=A(KN) - B                                                    00004900
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8
   1  C                                                                      00005000
   7        IF (A(KN)) 222,224,226                                          00005100
   3    222 WRITE(KIO6,1000) N                                              00005200
   7        STOP                                                            00005300
   3    224 WRITE(KIO6,1000) N                                              00005400
   7        STOP                                                            00005500
   2    226 MR=MIN0(MA1,NN-N)                                               00005600
   7        IF (MR) 200,200,228                                             00005700
   2    228 MN=KU - KL + 1                                                  00005800
   1  C                                                                     00005900
   7        DO 240 J=1,MR                                                   00006000
   7        MJ=MAXA(N+J) + J                                                00006100
   7        MNJ=MAXA(N+J+1) - MJ - 1                                        00006200
   7        IF (MNJ) 240,240,230                                            00006300
   2    230 ND=MIN0(MN,MNJ)                                                 00006400
   7        C=0.                                                            00006500
   7        KU=KN + ND                                                      00006600
   7        IC=MJ - KN                                                      00006700
   7        DO 300 KK=KL,KU                                                 00006800
   2    300 C=C + A(KK)*A(KK+IC)                                            00006900
   7        A(KN+IC)=A(KN+IC) - C                                           00007000
   2    240 CONTINUE                                                        00007100
   1  C                                                                     00007200
   2    200 CONTINUE                                                        00007300
   7        IF (NEGNL.EQ.0) RETURN                                          00007400
   1  C                                                                     00007500
   2    700 DO 400 N=2,NN                                                   00007600
   7        KL=MAXA(N) + 1                                                  00007700
   7        KU=MAXA(N+1) - 1                                                00007800
   7        IF (KU-KL) 400,410,410                                          00007900
   2    410 K=N                                                             00008000
   7        C=0.                                                            00008100
   7        DO 420 KK=KL,KU                                                 00008200
   7        K=K - 1                                                         00008300
   2    420 C=C + A(KK)*V(K)                                                00008400
   7        V(N)=V(N) - C                                                   00008500
   2    400 CONTINUE                                                        00008600
   7        GO TO 800                                                       00008700
   1  C                                                                     00008800
   3    799 IF (KKK.EQ.1 .AND. IND.EQ.1) RETURN                             00008900
   1  C                                                                     00009000
   2    800 DO 480 N=1,NN                                                   00009100
   7        K=MAXA(N)                                                       00009200
   2    480 V(N)=V(N)/A(K)                                                  00009300
   7        IF (NN.EQ.1) RETURN                                             00009400
   7        N=NN                                                            00009500
   7        DO 500 L=2,NN                                                   00009600
   7        KL=MAXA(N) + 1                                                  00009700
   7        KU=MAXA(N+1) - 1                                                00009800
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7        IF (KU-KL) 500,510,510                                          00009900
   2   510  K=N                                                            00010000
   7        DO 520 KK=KL,KU                                                 00010100
   7        K=K - 1                                                         00010200
   2   520  V(K)=V(K) - A(KK)*V(N)                                         00010300
   2   500  N=N - 1                                                        00010400
   1  C                                                                    00010500
   2   1000 FORMAT (// 78H STOP - STIFFNESS NOT POSITIVE DEFINITE          00010600
   6       1NEGATIVE PIVOT IN POSITION    I4)                             00010700
   2   1010 FORMAT (// 36H STOP BECAUSE ZERO PIVOT IN POSITION    I4 )     00010800
   7        RETURN                                                          00010900
   7        END                                                             00011000
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7          SUBROUTINE DCRACK (C,ANGLE,K,G,IK)                              00000100
   7          IMPLICIT REAL*8(A-H,O-Z)                                        00000200
   1  C                                                                       00000300
   7          DIMENSION C(4,1),D(4,4),T(4,4)                                  00000400
   7          COMMON /CRACK/ GAMMA,STIFAC,SHEFAC                              00000500
   1  C                                                                       00000600
   1  C      REAL K                                                           00000700
   1  C                                                                       00000800
   1  C                                                                       00000900
   1  C      COMPUTE ISOTROPIC STRESS-STRAIN LAW                              00001000
   1  C                                                                       00001100
   1  C                                                                       00001200
   7          DUM=0.666666667*G                                               00001300
   7          A1=K + 2.*DUM                                                   00001400
   7          B1=K - DUM                                                      00001500
   1  C                                                                       00001600
   7          C(1,1)=A1                                                       00001700
   7          C(1,2)=B1                                                       00001800
   7          C(1,3)=0.                                                       00001900
   7          C(1,4)=B1                                                       00002000
   7          C(2,2)=A1                                                       00002100
   7          C(2,3)=0.                                                       00002200
   7          C(2,4)=B1                                                       00002300
   7          C(3,3)=G                                                        00002400
   7          C(3,4)=0.                                                       00002500
   7          C(4,4)=A1                                                       00002600
   1  C                                                                       00002700
   1  C                                                                       00002800
   1  C      MODIFY STRESS-STRAIN LAW TO ACCOUNT FOR CRACKING                 00002900
   1  C                                                                       00003000
   1  C                                                                       00003100
   7          DFAC=STIFAC                                                     00003200
   7          IF (IK.EQ.0) RETURN                                             00003300
   7          C(3,3)=C(3,3)/SHEFAC                                            00003400
   7          IF (IK.EQ.1) GO TO 20                                           00003500
   7          DO 11 I=2,4                                                     00003600
   4       11 C(2,I)=C(2,I)/DFAC                                              00003700
   7          C(1,2)=C(1,2)/DFAC                                              00003800
   1  C                                                                       00003900
   7          IF (IK.EQ.2) GO TO 13                                           00004000
   7          C(1,1)=C(1,1)/DFAC                                              00004100
   7          C(1,3)=C(1,3)/DFAC                                              00004200
   7          C(1,4)=C(1,4)/DFAC                                              00004300
   1  C                                                                       00004400
   4       13 DO 12 I=1,4                                                     00004500
   7          DO 12 J=I,4                                                     00004600
   4       12 C(J,I)=C(I,J)                                                   00004700
   7          GO TO 102                                                       00004800
   1  C                                                                       00004900
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   4     20 DO 100 I=1,4                                                    00005000
   3    100 C(1,I)=C(1,I)/DFAC                                              00005100
   7        DO 101 I=1,4                                                    00005200
   7        DO 101 J=I,4                                                    00005300
   3    101 C(J,I)=C(I,J)                                                   00005400
   1  C                                                                     00005500
   1  C                                                                     00005600
   1  C     SET UP COORDINATE TRANSFORMATION FROM CRACKED ORIENTATION        00005700
   1  C                                                                     00005800
   1  C                                                                     00005900
   3    102 PI=4.0*DATAN(1.0D0)                                             00006000
   7        GAM=(ANGLE/360.)*2.*PI                                          00006100
   7        IF(DABS(GAM).LT.0D-8) GO TO 202                                 00006200
   7        SG=DSIN(GAM)                                                    00006300
   7        CG=DCOS(GAM)                                                    00006400
   7        T(1,1) =  CG**2                                                 00006500
   7        T(1,2) =  SG**2                                                 00006600
   7        T(1,3) =  CG* SG                                                00006700
   7        T(1,4) =  0.0                                                   00006800
   7        T(2,1) =  T(1,2)                                                00006900
   7        T(2,2) =  T(1,1)                                                00007000
   7        T(2,3) = -T(1,3)                                                00007100
   7        T(2,4) =  0.0                                                   00007200
   7        T(3,1) =  T(2,3)* 2.0                                           00007300
   7        T(3,2) = -T(3,1)                                                00007400
   7        T(3,3) =  T(1,1)- T(1,2)                                        00007500
   7        T(3,4) =  0.0                                                   00007600
   7        T(4,1) =  0.0                                                   00007700
   7        T(4,2) =  0.0                                                   00007800
   7        T(4,3) =  0.0                                                   00007900
   7        T(4,4) =  1.0                                                   00008000
   3    202 CONTINUE                                                        00008100
   1  C                                                                     00008200
   1  C                                                                     00008300
   1  C     ROTATE THE STRESS-STRAIN MATRIX TO GLOBAL COORDINATES            00008400
   1  C                                                                     00008500
   1  C                                                                     00008600
   7        IF(DABS(GAM).LT.0D-8) GO TO 90                                  00008700
   1  C                                                                     00008800
   1  C     T(TRANSPOSE) * C(MATERIAL)                                      00008900
   1  C                                                                     00009000
   7        DO 60 IR=1,4                                                    00009100
   7        DO 60 IC=1,4                                                    00009200
   7        D(IR,IC) = 0.0                                                  00009300
   7        DO 50 IN=1,4                                                    00009400
   4     50 D(IR,IC) = D(IR,IC) + T(IN,IR)* C(IN,IC)                        00009500
   4     60 CONTINUE                                                        00009600
   1  C                                                                     00009700
   1  C     T(TRANSPOSE) * C(MATERIAL) * T                                  00009800
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   1  C                                                                    00009900
   7        DO 80 IR=1,4                                                  00010000
   7        DO 80 IC=IR,4                                                 00010100
   7        C(IR,IC) = 0.0                                                00010200
   7        DO 70 IN=1,4                                                  00010300
   4     70 C(IR,IC) = C(IR,IC) + D(IR,IN)* T(IN,IC)                      00010400
   4     80 C(IC,IR)=C(IR,IC)                                             00010500
   1  C                                                                    00010600
   4     90 RETURN                                                        00010700
   1  C                                                                    00010800
   1  C                                                                    00010900
   7        END                                                           00011000
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7         SUBROUTINE DERIQ (NEL,XX,B,DET,R,S,X1BAR,NOD5)                        00000100
   7         IMPLICIT REAL*8(A-H,O-Z)                                              00000200
   1  C                                                                            00000300
   1  C                                                                            00000400
   1  C      EVALUATION OF THE STRAIN-DISPLACEMENT MATRIX AT POINT (R,S) FOR       00000500
   1  C      A QUADRILATERAL ELEMENT, AXISYMMETRIC GEOMETRY                        00000600
   1  C                                                                            00000700
   1  C                                                                            00000800
   7         COMMON /EL/ IND,ICOUNT,NPAR(20),NUMEG,NEGL,NEGNL,IMASS,IDAMP,ISTAT   00000900
   6        1           ,NDOF,KLIN,IEIG,IMASSN,IDAMPN                             00001000
   7         COMMON /TODIM/ BET,THIC,DE,IEL,NND5                                   00001100
   7         DIMENSION     XX(2,1),B(4,1),NOD5(1)                                  00001200
   7         DIMENSION     H(8),IPERM(4),P(2,8),XJ(2,2),XJI(2,2)                   00001300
   1  C                                                                            00001400
   7         EQUIVALENCE (NPAR(5),ITYP2D)                                          00001500
   1  C                                                                            00001600
   1  C                                                                            00001700
   1  C      FIND INTERPOLATION FUNCTIONS AND JACOBIAN                             00001800
   1  C                                                                            00001900
   7         CALL FUNCT2 (R,S,H,P,NOD5,XJ,DET,XX,NEL)                              00002000
   1  C                                                                            00002100
   1  C                                                                            00002200
   1  C      COMPUTE INVERSE OF THE JACOBIAN MATRIX                                00002300
   1  C                                                                            00002400
   7         DUM = 1.0/DET                                                         00002500
   7         XJI(1,1) = XJ(2,2)* DUM                                               00002600
   7         XJI(1,2) =-XJ(1,2)* DUM                                               00002700
   7         XJI(2,1) =-XJ(2,1)* DUM                                               00002800
   7         XJI(2,2) = XJ(1,1)* DUM                                               00002900
   1  C                                                                            00003000
   1  C      EVALUATE GLOBAL DERIVATIVE OPERATOR ( B-MATRIX )                      00003100
   1  C                                                                            00003200
   7         DO 130 K=1,IEL                                                        00003300
   7         K2=K*2                                                                00003400
   7         B(1,K2-1) = 0.                                                        00003500
   7         B(1,K2  ) = 0.                                                        00003600
   7         B(2,K2-1) = 0.                                                        00003700
   7         B(2,K2  ) = 0.                                                        00003800
   7         DO 120 I=1,2                                                          00003900
   7         B(1,K2-1) = B(1,K2-1) + XJI(1,I) * P(I,K)                             00004000
   3   120   B(2,K2  ) = B(2,K2  ) + XJI(2,I) * P(I,K)                             00004100
   7         B(3,K2  ) = B(1,K2-1)                                                 00004200
   3   130   B(3,K2-1) = B(2,K2  )                                                 00004300
   1  C                                                                            00004400
   1  C      IN CASE OF PLANE STRAIN OR PLANE STRESS ANALYSIS WE DO NOT INCLUDE    00004500
   1  C      THE NORMAL STRAIN COMPONENT                                           00004600
   1  C                                                                            00004700
   7         IF (ITYP2D.GT.0) RETURN                                               00004800
   1  C                                                                            00004900
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   1  C      COMPUTE THE RADIUS AT POINT (R,S)                                  00005000
   1  C                                                                         00005100
   7         X1BAR = 0.0                                                       00005200
   7         DO 50 K=1,IEL                                                     00005300
   4      50 X1BAR = X1BAR + H(K)* XX(1,K)                                     00005400
   1  C                                                                         00005500
   1  C      EVALUATE THE HOOP STRAIN-DISPLACEMENT RELATION                     00005600
   1  C                                                                         00005700
   7         IF(X1BAR.GT.1.0E-8) GO TO 150                                     00005800
   1  C                                                                         00005900
   1  C      FOR THE CASE OF ZERO RADIUS EQUATE RADIAL TO HOOP STRAIN           00006000
   1  C                                                                         00006100
   7         ND=2*IEL                                                          00006200
   7         DO 140 K=1,ND                                                     00006300
   3    140  B(4,K)=B(1,K)                                                     00006400
   7         RETURN                                                            00006500
   1  C                                                                         00006600
   1  C      NON-ZERO RADIUS                                                   00006700
   1  C                                                                         00006800
   3    150  DUM = 1.0/X1BAR                                                  00006900
   7         DO 160 K=1,IEL                                                    00007000
   7         K2=K*2                                                            00007100
   7         B(4,K2  ) = 0.                                                    00007200
   3    160  B(4,K2-1) = H(K) * DUM                                           00007300
   1  C                                                                         00007400
   7         RETURN                                                            00007500
   1  C                                                                         00007600
   1  C                                                                         00007700
   7         END                                                               00007800
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7        SUBROUTINE DERIQ3  (NEL,XX,B,DET,R,S,T,NOD9)                          00000100
   7        IMPLICIT REAL*8(A-H,O-Z)                                              00000200
   1  C                                                                           00000300
   1  C                                                                           00000400
   1  C . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 00000500
   1  C .                                                                       . 00000600
   1  C .   P R O G R A M                                                       . 00000700
   1  C .                                                                       . 00000800
   1  C .      EVALUATES STRAIN-DISPLACEMENT MATRIX B AT POINT (R,S,T)          . 00000900
   1  C .                                                                       . 00001000
   1  C .      CURVILINEAR HEXAHEDRON   8 TO 21 NODES                           . 00001100
   1  C .                                                                       . 00001200
   1  C . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 00001300
   1  C                                                                           00001400
   1  C                                                                           00001500
   1  C                                                                           00001600
   7        COMMON /EL/ IND,ICOUNT,NPAR(20),NUMEG,NEGL,NEGNL,IMASS,IDAMP,ISTAT00001700
   6       1           ,NDOFDM,KLIN,IEIG,IMASSN,IDAMPN                          00001800
   7        COMMON /THREED/ DE,IELD,IELX,IEL,NPT,IDW,NND9                       00001900
   7        DIMENSION XX(3,1),B(1),NOD9(1)                                      00002000
   7        DIMENSION H(21),P(3,21),XJ(3,3),XJI(3,3)                            00002100
   1  C                                                                           00002200
   1  C                                                                           00002300
   1  C     FIND INTERPOLATION FUNCTIONS AND THEIR DERIVATIVES                    00002400
   1  C     EVALUATE JACOBIAN MATRIX AT POINT (R,S,T)                             00002500
   1  C     COMPUTE DETERMINANT OF JACOBIAN MATRIX AT POINT (R,S,T)               00002600
   1  C                                                                           00002700
   1  C                                                                           00002800
   7        CALL FUNCT (R,S,T,H,P,NOD9,XJ,DET,XX)                               00002900
   1  C                                                                           00003000
   1  C                                                                           00003100
   1  C     COMPUTE INVERSE OF JACOBIAN MATRIX                                    00003200
   1  C                                                                           00003300
   1  C                                                                           00003400
   7        DUM=1.0/DET                                                         00003500
   7        XJI(1,1)=DUM*( XJ(2,2)*XJ(3,3) - XJ(2,3)*XJ(3,2))                   00003600
   7        XJI(2,1)=DUM*(-XJ(2,1)*XJ(3,3) + XJ(2,3)*XJ(3,1))                   00003700
   7        XJI(3,1)=DUM*( XJ(2,1)*XJ(3,2) - XJ(2,2)*XJ(3,1))                   00003800
   7        XJI(1,2)=DUM*(-XJ(1,2)*XJ(3,3) + XJ(1,3)*XJ(3,2))                   00003900
   7        XJI(2,2)=DUM*( XJ(1,1)*XJ(3,3) - XJ(1,3)*XJ(3,1))                   00004000
   7        XJI(3,2)=DUM*(-XJ(1,1)*XJ(3,2) + XJ(1,2)*XJ(3,1))                   00004100
   7        XJI(1,3)=DUM*( XJ(1,2)*XJ(2,3) - XJ(1,3)*XJ(2,2))                   00004200
   7        XJI(2,3)=DUM*(-XJ(1,1)*XJ(2,3) + XJ(1,3)*XJ(2,1))                   00004300
   7        XJI(3,3)=DUM*( XJ(1,1)*XJ(2,2) - XJ(1,2)*XJ(2,1))                   00004400
   1  C                                                                           00004500
   1  C                                                                           00004600
   1  C     EVALUATE B MATRIX IN GLOBAL (X,Y,Z) COORDINATES                       00004700
   1  C                                                                           00004800
   1  C                                                                           00004900
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7          DO 130 K=1,IEL                                                  00005000
   7          K2=K*3                                                          00005100
   7          DO 125 I=1,3                                                    00005200
   3    125   B(K2+1-I)=0.0                                                   00005300
   7          DO 120 I=1,3                                                    00005400
   7          B(K2-2)=B(K2-2) + XJI(1,I)*P(I,K)                               00005500
   7          B(K2-1)=B(K2-1) + XJI(2,I)*P(I,K)                               00005600
   3    120   B(K2)=B(K2) + XJI(3,I)*P(I,K)                                   00005700
   3    130   CONTINUE                                                        00005800
   1  C                                                                       00005900
   1  C                                                                       00006000
   7          RETURN                                                          00006100
   1  C                                                                       00006200
   7          END                                                             00006300
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7        SUBROUTINE DRUCK (PROP,SIG,EPS,ANGLE,COHES,IPEL)                    00000100
   7        IMPLICIT REAL*8(A-H,O-Z)                                            00000200
   1  C                                                                         00000300
   1  C                                                                         00000400
   1  C . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 00000500
   1  C .                                                                     . 00000600
   1  C .      IST     NUMBER OF STRESS COMPONENTS                            . 00000700
   1  C .      ISR     NUMBER OF STRAIN COMPONENTS                            . 00000800
   1  C .      SIG     STRESSES AT THE END OF THE PREVIOUS UPDATE             . 00000900
   1  C .      EPS     STRAINS  AT THE END OF THE PREVIOUS UPDATE             . 00001000
   1  C .      RATIO   PART OF STRAIN INCREMENT TAKEN ELASTICALLY             . 00001100
   1  C .      DELEPS  INCREMENT IN STRAINS                                   . 00001200
   1  C .      DELSIG  INCREMENT IN STRESSES, ASSUMING ELASTIC BEHAVIOR       . 00001300
   1  C .                                                                     . 00001400
   1  C .      PROP(1) YOUNG S  MODULUS                                       . 00001500
   1  C .      PROP(2) POISSON S  RATIO                                       . 00001600
   1  C .      PROP(3) ANGLE OF FRICTION (INPUT UNIT = DEGREES)               . 00001700
   1  C .      PROP(4) COHESION                                               . 00001800
   1  C .                                                                     . 00001900
   1  C .      IPEL    =  1, MATERIAL ELASTIC,                                  00002000
   1  C .              =  2, MATERIAL PLASTIC                                   00002100
   1  C .                                                                     . 00002200
   1  C . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 00002300
   1  C                                                                         00002400
   7        COMMON/BBCCIO/KIO1,KIO2,KIO3,KIO4,KIO5,KIO6,KIO7,KIO8,KIO9,        00002500
   6       1KIO10,KIO11,KIO12,KIO13,KIO14,KIO15,KIO16,KIO17,KIO18,KIO19,KIO20,00002600
   6       2KIO21,KIO22,KIO23,KIO24,KIO25,KIO26,KIO27,KIO28,KIO29,KIO30        00002700
   1  C                                                                         00002800
   7        COMMON /EL    / IND,ICOUNT,NPAR(20),NUMEG,NEGL,NEGNL,IMASS,IDAMP, 00002900
   6       1                ISTAT,NDOF,KLIN,IEIG,IMASSN,IDAMPN                 00003000
   7        COMMON /VAR   / NG,KPRI,MODEX,KSTEP,ITE,ITEMAX,IREF,IEQREF,INOCMD 00003100
   7        COMMON /DISDER/ DISD(5)                                           00003200
   7        COMMON /MATMOD/ STRESS(4),STRAIN(4),C(4,4),IPT,NEL                00003300
   1  C     COMMON /DRPRAG/ ISR,IST,A1,B1,C1,A2,B2,C2,D2,G,BM,ALFA,YLD        00003400
   7        COMMON/DRPRAG/ A1,B1,C1,A2,B2,C2,D2,G,BM,ALFA,YLD,ISR,ISI         00003500
   1  C                                                                         00003600
   7        DIMENSION PROP(1),SIG(1),EPS(1)                                   00003700
   7        DIMENSION TAU(4),DELSIG(4),DELEPS(4),DEPS(4),STATE(2)             00003800
   1  C                                                                         00003900
   7        EQUIVALENCE (NPAR(3),INDNL),(NPAR(5),ITYP2D),(DELEPS(1),DEPS(1))  00004000
   1  C                                                                         00004100
   7        DATA NGLAST/1000/, STATE/1HE,1HP/                                 00004200
   1  C                                                                         00004300
   1  C                                                                         00004400
   1  C                                                                         00004500
   7        IF (IPT.NE.1) GO TO 110                                           00004600
   1  C                                                                         00004700
   7        IST=4                                                             00004800
   7        IF (ITYP2D.EQ.2) IST=3                                            00004900
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7          ISR=3                                                          00005000
   7          IF (ITYP2D.EQ.0) ISR=4                                         00005100
   7          YM=PROP(1)                                                     00005200
   7          PV=PROP(2)                                                     00005300
   7          D2=PV/(PV-1.)                                                  00005400
   7          G =YM/(1.+PV)/2.                                               00005500
   7          BM=YM/(1.-2.*PV)/3.                                            00005600
   7          A2=BM+4.*G/3.                                                  00005700
   7          B2=BM-2.*G/3.                                                  00005800
   7          C1=G                                                           00005900
   7          C2=G                                                           00006000
   1  C                                                                      00006100
   7          IF (ITYP2D.EQ.2) GO TO 105                                     00006200
   1  C                                                                      00006300
   1  C      PLANE STRAIN  /  AXISYMMETRIC                                  00006400
   7          A1=A2                                                          00006500
   7          B1=B2                                                          00006600
   7          GO TO 110                                                      00006700
   1  C                                                                      00006800
   1  C      PLANE STRESS                                                   00006900
   3    105 A1=YM/(1.-PV*PV)                                                 00007000
   7          B1=A1*PV                                                       00007100
   1  C                                                                      00007200
   3    110 ANG=ANGLE                                                        00007300
   7          CEE=COHES                                                      00007400
   7          YM=DCOS(ANG)                                                   00007500
   7          PV=DSIN(ANG)                                                   00007600
   7          YLD=1.732050808*(3.-PV)                                        00007700
   1  C                                                                      00007800
   7          ALFA=2.*PV/YLD                                                 00007900
   7          YLD=6.*CEE*YM/YLD                                              00008000
   1  C                                                                      00008100
   1  C                                                                      00008200
   1  C         1. CALCULATE INCREMENTAL STRAINS                            00008300
   1  C                                                                      00008400
   7          DO 120 I=1,ISR                                                 00008500
   3    120 DELEPS(I) = STRAIN(I) - EPS(I)                                   00008600
   1  C                                                                      00008700
   1  C                                                                      00008800
   1  C         2. CALCULATE THE STRESS INCREMENT,                          00008900
   1  C            ASSUMING ELASTIC BEHAVIOR                                00009000
   1  C                                                                      00009100
   7          DELSIG(1) = A1*DELEPS(1) + B1*DELEPS(2)                        00009200
   7          DELSIG(2) = B1*DELEPS(1) + A1*DELEPS(2)                        00009300
   7          DELSIG(3) = C1*DELEPS(3)                                       00009400
   7          DELSIG(4) = 0.                                                 00009500
   7          IF (ITYP2D.EQ.2) GO TO 150                                     00009600
   7          DELSIG(4) = B1 * (DELEPS(1)+DELEPS(2))                         00009700
   7          IF (ITYP2D.EQ.1) GO TO 150                                     00009800
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7        DELSIG(1) = DELSIG(1) + B1*DELEPS(4)                                  00009900
   7        DELSIG(2) = DELSIG(2) + B1*DELEPS(4)                                  00010000
   7        DELSIG(4) = DELSIG(4) + A1*DELEPS(4)                                  00010100
   1  C                                                                           00010200
   1  C                                                                           00010300
   1  C        3. CALCULATE TOTAL STRESSES,                                       00010400
   1  C           ASSUMING ELASTIC BEHAVIOR                                       00010500
   1  C                                                                           00010600
   3    150 TAU(4) = 0.                                                         00010700
   7        DO 160 I=1,IST                                                      00010800
   3    160 TAU(I) = SIG(I) + DELSIG(I)                                         00010900
   1  C                                                                           00011000
   1  C        4. CHECK WHETHER *TAU* STATE OF STRESS FALLS                       00011100
   1  C           OUTSIDE THE LOADING SURFACE                                     00011200
   1  C                                                                           00011300
   7        SM = (TAU(1)+TAU(2)+TAU(4))/3.                                      00011400
   7        SX = TAU(1) - SM                                                    00011500
   7        SY = TAU(2) - SM                                                    00011600
   7        SS = TAU(3)                                                         00011700
   7        SZ = TAU(4) - SM                                                    00011800
   7        SBAR=DSQRT( .5*(SX*SX+SY*SY+SZ*SZ) +SS*SS)                          00011900
   1  C                                                                           00012000
   7        FT=3.*ALFA*SM + SBAR - YLD                                          00012100
   1  C                                                                           00012200
   7        IF (FT) 170,170,300                                                 00012300
   1  C                                                                           00012400
   1  C     STATE OF STRESS WITHIN LOADING SURFACE  -  ELASTIC BEHAVIOR           00012500
   1  C                                                                           00012600
   3    170 IPEL=1                                                              00012700
   7        STRESS(4) = 0.                                                      00012800
   7        DO 180 I=1,IST                                                      00012900
   3    180 STRESS(I) = TAU(I)                                                  00013000
   7        IF (ITYP2D.EQ.2)  STRAIN(4)=EPS(4) + D2*(DELEPS(1)+DELEPS(2))       00013100
   7        GO TO 400                                                           00013200
   1  C                                                                           00013300
   1  C                                                                           00013400
   1  C     STATE OF STRESS OUTSIDE LOADING SURFACE  -  PLASTIC BEHAVIOR          00013500
   1  C                                                                           00013600
   3    300 IF (IPEL.EQ.1) GO TO 320                                            00013700
   1  C                                                                           00013800
   1  C     .......WAS PLASTIC                                                    00013900
   1  C                                                                           00014000
   7        IPEL=2                                                              00014100
   7        RATIO = 0.                                                          00014200
   7        DO 315 I=1,IST                                                      00014300
   3    315 TAU(I) = SIG(I)                                                     00014400
   7        GO TO 370                                                           00014500
   1  C                                                                           00014600
   1  C     .......WAS ELASTIC                                                    00014700
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   1  C                                                                      00014800
   1  C     DETERMINE PART OF STRAIN TAKEN ELASTICLY                         00014900
   1  C                                                                      00015000
   3    320 IPEL=2                                                          00015100
   1  C                                                                      00015200
   7        SM = (SIG(1)+SIG(2)+SIG(4))/3.                                  00015300
   7        SX = SIG(1) - SM                                                00015400
   7        SY = SIG(2) - SM                                                00015500
   7        SS = SIG(3)                                                     00015600
   7        SZ = SIG(4) - SM                                                00015700
   1  C                                                                      00015800
   7        DM = (DELSIG(1)+DELSIG(2)+DELSIG(4))/3.                         00015900
   7        DX = DELSIG(1) - DM                                             00016000
   7        DY = DELSIG(2) - DM                                             00016100
   7        DS = DELSIG(3)                                                  00016200
   7        DZ = DELSIG(4) - DM                                             00016300
   1  C                                                                      00016400
   7        A=DX*DX + DY*DY + 2.*DS*DS + DZ*DZ - 18.*( (ALFA*DM)**2 )       00016500
   7        B=SX*DX + SY*DY + 2.*SS*DS + SZ*DZ + ALFA*DM*(6.*YLD-18.*ALFA*SM) 00016600
   7        E=SX*SX + SY*SY + 2.*SS*SS + SZ*SZ +                            00016700
   6       1  ALFA*SM*(12.*YLD-18.*ALFA*SM) - 2.*YLD*YLD                    00016800
   1  C                                                                      00016900
   7        RATIO=(-B + DSQRT(B*B-A*E))/A                                   00017000
   1  C                                                                      00017100
   7        DO 350 I=1,IST                                                  00017200
   3    350 TAU (I) = SIG(I) + RATIO*DELSIG(I)                              00017300
   7        IF (ITYP2D.EQ.2)  STRAIN(4)=EPS(4)+RATIO*D2*(DELEPS(1)+DELEPS(2)) 00017400
   1  C                                                                      00017500
   1  C     *TAU* NOW CONTAINS (PREVIOUS STRESSES +                          00017600
   1  C     STRESSES DUE TO ELASTIC STRAIN INCREMENTS)                       00017700
   1  C                                                                      00017800
   1  C                                                                      00017900
   1  C        5. CALCULATE PLASTIC STRESSES                                 00018000
   1  C                                                                      00018100
   1  C     DETERMINE INCREMENT INTERVAL                                     00018200
   3    370 M=20.*DSQRT(2.*YLD*FT)/YLD+1.                                   00018300
   7        IF (M.GT.30) M=30                                               00018400
   7        XM = (1.-RATIO)/FLOAT(M)                                        00018500
   1  C                                                                      00018600
   1  C                                                                      00018700
   7        DO 380 I=1,ISR                                                  00018800
   3    380 DEPS(I) = XM*DELEPS(I)                                          00018900
   1  C                                                                      00019000
   1  C                                                                      00019100
   1  C     .....  CALCULATION OF ELASTOPLASTIC STRESSES  .....  (START)     00019200
   1  C                                                                      00019300
   7        DO 600 IM=1,M                                                   00019400
   1  C                                                                      00019500
   7        CALL PRAGER (TAU,DEPS,C)                                        00019600
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8
   1  C                                                                        00019700
   7        DO 560 I=1,IST                                                    00019800
   7        DO 560 J=1,ISR                                                    00019900
   3    560 TAU(I) = TAU(I) + C(I,J) * DEPS(J)                                00020000
   1  C                                                                       00020100
   1  C                                                                       00020200
   1  C     CORRECTION                                                        00020300
   1  C                                                                       00020400
   7        DM = (TAU(1)+TAU(2)+TAU(4))/3.                                    00020500
   7        DX = TAU(1) - DM                                                  00020600
   7        DY = TAU(2) - DM                                                  00020700
   7        DS = TAU(3)                                                       00020800
   7        DZ = TAU(4) - DM                                                  00020900
   1  C                                                                       00021000
   1  C     PERFECTLY PLASTIC MATERIAL                                        00021100
   1  C                                                                       00021200
   7        SBAR=DSQRT(.5*(DX*DX+DY*DY+DZ*DZ)+DS*DS)                          00021300
   7        FT=3.*ALFA*DM + SBAR - YLD                                        00021400
   7        FTA=DABS(FT)/YLD                                                  00021500
   7        IF (FTA.LT.0.005) GO TO 600                                       00021600
   1  C                                                                       00021700
   7        A=.25 - 9.*(ALFA**4)                                              00021800
   7        B=SBAR - 18.*(ALFA**3)*DM + 6.*(ALFA**2)*YLD                      00021900
   7        E=(SBAR**2) - 9.*(ALFA**2)*(DM**2) + 6.*ALFA*YLD*DM - YLD*YLD     00022000
   7        E=(B-DSQRT(B*B-4.*A*E))/(2.*A)                                    00022100
   1  C                                                                       00022200
   7        TAU(1)=TAU(1) - E*(ALFA + .5*DX/SBAR)                             00022300
   7        TAU(2)=TAU(2) - E*(ALFA + .5*DY/SBAR)                             00022400
   7        TAU(3)=TAU(3) - E*(       .5*DS/SBAR)                             00022500
   7        IF (IST.EQ.3) GO TO 600                                           00022600
   7        TAU(4)=TAU(4) - E*(ALFA + .5*DZ/SBAR)                             00022700
   1  C                                                                       00022800
   3    600 CONTINUE                                                          00022900
   1  C                                                                       00023000
   1  C     ..... CALCULATION OF ELASTOPLASTIC STRESSES ..... ( END )         00023100
   1  C                                                                       00023200
   7        STRESS(4) = 0.                                                    00023300
   7        DO 390 I=1,IST                                                    00023400
   3    390 STRESS(I) = TAU(I)                                                00023500
   1  C                                                                       00023600
   1  C                                                                       00023700
   1  C     UPDATING STRESSES,STRAINS,ANGLE,COHESION                          00023800
   1  C                                                                       00023900
   3    400 DO 410 I=1,IST                                                    00024000
   3    410 SIG(I) = STRESS(I)                                                00024100
   7        DO 420 I=1,ISR                                                    00024200
   3    420 EPS(I) = STRAIN(I)                                                00024300
   7        IF (ITYP2D.EQ.2)  EPS(4)=STRAIN(4)                                00024400
   7        ANGLE=ANG                                                         00024500
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7        COHES=CEE                                                      00024600
   7        IF (KPRI.EQ.0) GO TO 700                                       00024700
   1  C                                                                    00024800
   1  C                                                                    00024900
   7        IF (ICOUNT.EQ.3) RETURN                                        00025000
   1  C                                                                    00025100
   1  C        7. FORM THE MATERIAL LAW                                    00025200
   1  C                                                                    00025300
   7        IF (IPEL.EQ.1) GO TO 450                                       00025400
   1  C                                                                    00025500
   1  C     ELASTO-PLASTIC                                                 00025600
   7        CALL PRAGER (STRESS,DEPS,C)                                    00025700
   7        RETURN                                                         00025800
   1  C                                                                    00025900
   1  C     ELASTIC                                                        00026000
   3    450 DO 460 I=1,ISR                                                 00026100
   7        DO 460 J=1,ISR                                                 00026200
   3    460 C(I,J)=0.                                                      00026300
   7        C(1,1)=A1                                                      00026400
   7        C(2,1)=B1                                                      00026500
   7        C(1,2)=B1                                                      00026600
   7        C(2,2)=A1                                                      00026700
   7        C(3,3)=C1                                                      00026800
   7        IF (ITYP2D.EQ.1) RETURN                                        00026900
   7        IF (ITYP2D.EQ.2) GO TO 470                                     00027000
   7        C(1,4)=B1                                                      00027100
   7        C(2,4)=B1                                                      00027200
   7        C(4,1)=B1                                                      00027300
   7        C(4,2)=B1                                                      00027400
   7        C(4,4)=A1                                                      00027500
   1  C                                                                    00027600
   7        RETURN                                                         00027700
   1  C                                                                    00027800
   3    470 C(4,1)=B2                                                      00027900
   7        C(4,2)=B2                                                      00028000
   7        C(4,3)=0.                                                      00028100
   7        C(4,4)=A2                                                      00028200
   1  C                                                                    00028300
   7        RETURN                                                         00028400
   1  C                                                                    00028500
   1  C                                                                    00028600
   1  C     P R I N T I N G   O F   S T R E S S E S                        00028700
   1  C                                                                    00028800
   3    700 IF (IPEL.EQ.1) GO TO 800                                       00028900
   1  C                                                                    00029000
   7        DM = (STRESS(1)+STRESS(2)+STRESS(4))/3.                        00029100
   7        DX = STRESS(1) - DM                                            00029200
   7        DY = STRESS(2) - DM                                            00029300
   7        DS = STRESS(3)                                                 00029400
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7        DZ = STRESS(4) - DM                                                  00029500
   7        SBAR=DSQRT(.5*(DX*DX+DY*DY+DZ*DZ)+DS*DS)                             00029600
   7        FT=3.*ALFA*DM + SBAR - YLD                                           00029700
   1  C                                                                          00029800
   1  C                                                                          00029900
   3    800 IF (NG.NE.NGLAST) GO TO 802                                          00030000
   7        IF (NEL.GT.NELAST) GO TO 806                                         00030100
   7        IF (IPT-1) 810,808,810                                               00030200
   1  C                                                                          00030300
   3    802 NGLAST = NG                                                          00030400
   3    808 IF (ITYP2D) 803,805,803                                              00030500
   3    803 WRITE(KIO6,2002)                                                     00030600
   7        GO TO 806                                                            00030700
   3    805 WRITE(KIO6,2003)                                                     00030800
   1  C                                                                          00030900
   3    806 NELAST=NEL                                                           00031000
   7        WRITE(5236,2004) NEL                                                 00031100
   1  C                                                                          00031200
   3    810 CALL MAXMIN (STRESS,SX,SY,SM)                                        00031300
   7        IF (ITYP2D) 813,815,813                                              00031400
   1  C                                                                          00031500
   3    813 WRITE(KIO6,2005) IPT,STATE(IPEL),(STRESS(I),I=1,3),SX,SY,SM,FT       00031600
   7        RETURN                                                               00031700
   1  C                                                                          00031800
   3    815 WRITE(KIO6,2007)IPT,STATE(IPEL),STRESS(4),(STRESS(I),I=1,3),         00031900
   6       1                SX,SY,SM,FT                                         00032000
   7        RETURN                                                               00032100
   1  C                                                                          00032200
   1  C                                                                          00032300
   2   2002 FORMAT (90H ELEMENT  STRESS                                         00032400
   6       1                                ,21X,5HYIELD/99H NUM/IPT   STAT00032500
   6       2E      STRESS-YY     STRESS-ZZ     STRESS-YZ       MAX STRESS    M00032600
   6       3IN STRESS    ANGLE,9X,8HFUNCTION / )                               00032700
   2   2003 FORMAT ( 104H ELEMENT  STRESS                                       00032800
   6       1                                                ,21X,5HYIELD / 00032900
   6       2 113H NUM/IPT   STATE      STRESS-XX     STRESS-YY     STRESS-ZZ  00033000
   6       3   STRESS-YZ       MAX STRESS    MIN STRESS    ANGLE,9X,8HFUNCTION00033100
   6       4 / )                                                              00033200
   2    2004 FORMAT (I4)                                                      00033300
   2    2005 FORMAT (5X,I2,2X,A1,6HLASTIC,1X,3E14.6,3X,2E14.6,3X,F6.2,3X,E14.6)00033400
   2    2007 FORMAT (5X,I2,2X,A1,6HLASTIC,1X,4E14.6,3X,2E14.6,3X,F6.2,3X,E14.6)00033500
   2    2100 FORMAT (1H1,21H STOP  DRUCKER-PRAGER )                           00033600
   1  C                                                                       00033700
   7        END                                                               00033800
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7        SUBROUTINE ELCAL                                                      00000100
   7        IMPLICIT REAL*8(A-H,O-Z)                                              00000200
   1  C                                                                           00000300
   1  C . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 00000400
   1  C .                                                                       . 00000500
   1  C .   P R O G R A M                                                     . 00000600
   1  C .      . TO CALL THE APPROPRIATE ELEMENT ROUTINES FOR READING,        . 00000700
   1  C .        GENERATING AND STORING THE ELEMENT DATA                       . 00000800
   1  C .                                                                       . 00000900
   1  C . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 00001000
   7        COMMON/BBCCIO/KIO1,KIO2,KIO3,KIO4,KIO5,KIO6,KIO7,KIO8,KIO9,          00001100
   6       1KIO10,KIO11,KIO12,KIO13,KIO14,KIO15,KIO16,KIO17,KIO18,KIO19,KIO20,  00001200
   6       2KIO21,KIO22,KIO23,KIO24,KIO25,KIO26,KIO27,KIO28,KIO29,KIO30         00001300
   1  C                                                                           00001400
   7        COMMON /SOL/ NUMNP,NEQ,NWK,NWM,NWC,NUMEST,MIDEST,MAXEST,NSTE,MA      00001500
   7        COMMON /EL/ IND,ICOUNT,NPAR(20),NUMEG,NEGL,NEGNL,IMASS,IDAMP,ISTAT  00001600
   6       1           ,NDOF,KLIN,IEIG,IMASSN,IDAMPN                            00001700
   7        COMMON /VAR/ NG,KPRI,MODEX,KSTEP,ITE,ITEMAX,IREF,IEQREF,INOCMD      00001800
   7        COMMON /ELGFLG/ NP1MX,NP7MX                                         00001900
   7        COMMON /NSAPI/ OPVAR(7),TSTART,IEQUIT,NUMREF,IPRI,ISREF             00002000
   7        COMMON A(1)                                                         00002100
   1  C                                                                           00002200
   7        EQUIVALENCE (NPAR(10),NINT),(NPAR(11),NINTZ),(NPAR(16),NUMMAT)      00002300
   6       1           ,(NPAR(3),INDNL),(NPAR(15),MODEL)                        00002400
   6       2           ,(NPAR(5),ITYP2D)                                        00002500
   1  C                                                                           00002600
   7        REWIND KIO11                                                        00002700
   7        REWIND KIO12                                                        00002800
   7        NN = 11                                                             00002900
   7        K=1                                                                 00003000
   7        NP1MX=0                                                             00003100
   7        NP7MX=0                                                             00003200
   7        WRITE(KIO6,2000)                                                    00003300
   1  C                                                                           00003400
   1  C                                                                           00003500
   1  C     LOOP OVER ALL ELEMENT GROUPS                                          00003600
   1  C                                                                           00003700
   1  C                                                                           00003800
   7        DO 100 NG=1,NUMEG                                                   00003900
   7        IF (NG.NE.1) WRITE(KIO6,2005)                                       00004000
   7        IF (NG.GT.NEGL) NN=12                                               00004100
   7        IF (NG.GT.NEGL) K=2                                                 00004200
   1  C                                                                           00004300
   7        READ(KIO5,1000)NPAR                                                 00004400
   1  C                                                                           00004500
   1  C                                                                           00004600
   1  C     DEFAULTS AND DATA CHECK                                               00004700
   1  C                                                                           00004800
   1  C                                                                           00004900
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7        IF (INDNL.EQ.0) INDNL=1                                                00005000
   7        IF (NINT.EQ.0) NINT=2                                                  00005100
   7        IF (NINTZ.EQ.0) NINTZ=2                                                00005200
   7        IF (NINT.LE.4 .AND. NINTZ.LE.4) GO TO 10                               00005300
   7        WRITE(KIO6,3000)                                                       00005400
   7        STOP                                                                   00005500
   4     10 IF (NPAR(1).EQ.2 .AND. NPAR(7).EQ.0) NPAR(7)=8                         00005600
   7        IF (NPAR(1).EQ.3 .AND. NPAR(7).EQ.0) NPAR(7)=21                        00005700
   7        IF (NUMMAT.EQ.0) NUMMAT=1                                              00005800
   7        NPAR1=NPAR(1)                                                          00005900
   7        NPAR7=NPAR(7)                                                          00006000
   7        IF (NPAR1.GT.NP1MX) NP1MX=NPAR1                                        00006100
   7        IF (NPAR7.GT.NP7MX) NP7MX=NPAR7                                        00006200
   7        IF (NN.EQ.11) WRITE(KIO6,2011)NG                                       00006300
   7        IF (NN.EQ.12) WRITE (KIO6,2012)NG                                      00006400
   1  C                                                                            00006500
   1  C                                                                            00006600
   1  C     CHECK EACH MATERIAL MODEL FOR PERMISSIBILITY OF                         00006700
   1  C        1.   EQUILIBRIUM ITERATIONS (IF SPECIFIED)                          00006800
   1  C        2.   NONLINEAR FORMULATION SPECIFIED                                00006900
   1  C        2.   PLANE STRESS OR STRAIN, AXISYMMETIC FORMULATION (IF 2-D)       00007000
   1  C                                                                            00007100
   1  C                                                                            00007200
   1  C     ***** NONSAP VERSION OF FEBRUARY 1974 *****                            00007300
   1  C                                                                            00007400
   1  C                                                                            00007500
   7        GO TO (1,2,3) , NPAR1                                                  00007600
   1  C                                                                            00007700
   1  C                                                                            00007800
   1  C     TRUSS MATERIAL MODELS                                                  00007900
   1  C                                                                            00008000
   5     1  IF (MODEL.EQ.1) GO TO 101                                             00008100
   7        IF (NN.EQ.12) GO TO 20                                                 00008200
   7        WRITE(KIO6,3002)                                                       00008300
   7        STOP                                                                   00008400
   3   101  IF (INDNL.EQ.2.OR.NN.EQ.11) GO TO 20                                  00008500
   7        WRITE(KIO6,3003)                                                       00008600
   7        STOP                                                                   00008700
   1  C                                                                            00008800
   1  C     2/D CONTINUUM MATERIAL MODELS                                          00008900
   1  C                                                                            00009000
   5     2  IF (MODEL.LE.2) GO TO 220                                             00009100
   7        IF (MODEL.EQ.3 .AND. IEQUIT.GT.NSTE) GO TO 210                         00009200
   7        IF (MODEL.EQ.4 .AND. IEQUIT.GT.NSTE) GO TO 210                         00009300
   7        IF (MODEL.EQ.5 .AND. IEQUIT.GT.NSTE) GO TO 210                         00009400
   7        IF (MODEL.GE.6) GO TO 210                                              00009500
   7        WRITE(KIO6,3001)NG                                                     00009600
   7        STOP                                                                   00009700
   3   210  IF (NN.EQ.12) GO TO 220                                               00009800
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7        WRITE(KIO6,3002)                                                  00009900
   7        STOP                                                              00010000
   3    220 IF (NN.EQ.11) GO TO 230                                           00010100
   7        IF (INDNL.GT.1) GO TO 221                                         00010200
   7        IF (MODEL.GE.3 .AND. MODEL.LE.7) GO TO 230                        00010300
   7        IF (MODEL.GE.9) GO TO 230                                         00010400
   7        WRITE(KIO6,3003)                                                  00010500
   7        STOP                                                              00010600
   3    221 IF (INDNL.GT.2) GO TO 222                                         00010700
   7        IF (MODEL.LE.3) GO TO 230                                         00010800
   7        IF (MODEL.EQ.6) GO TO 230                                         00010900
   7        IF (MODEL.GE.8) GO TO 230                                         00011000
   7        WRITE(KIO6,3003)                                                  00011100
   7        STOP                                                              00011200
   3    222 IF (INDNL.GT.3) GO TO 229                                         00011300
   7        IF (MODEL.EQ.1) GO TO 230                                         00011400
   3    229 WRITE(KIO6,3003)                                                  00011500
   7        STOP                                                              00011600
   3    230 IF (ITYP2D.GT.0) GO TO 231                                        00011700
   7        IF (MODEL.LE.4) GO TO 20                                          00011800
   7        IF (MODEL.EQ.6 .OR. MODEL.EQ.7) GO TO 20                          00011900
   7        IF (MODEL.GE.9) GO TO 20                                          00012000
   7        WRITE(KIO6,3004)                                                  00012100
   7        STOP                                                              00012200
   3    231 IF (ITYP2D.GT.1) GO TO 232                                        00012300
   7        IF (MODEL.NE.8) GO TO 20                                          00012400
   7        WRITE (6,3004)                                                    00012500
   7        STOP                                                              00012600
   3    232 IF (ITYP2D.GT.2) GO TO 239                                        00012700
   7        IF (MODEL.LE.2) GO TO 20                                          00012800
   7        IF (MODEL.GT.5) GO TO 20                                          00012900
   3    239 WRITE(KIO6,3004)                                                  00013000
   7        STOP                                                              00013100
   1  C                                                                       00013200
   1  C     3/D CONTINUUM MATERIAL MODELS                                     00013300
   1  C                                                                       00013400
   5      3 IF (MODEL.LE.2) GO TO 320                                         00013500
   7        IF (MODEL.EQ.3 .AND. IEQUIT.GT.NSTE) GO TO 310                    00013600
   7        IF (MODEL.GT.4) GO TO 310                                         00013700
   7        WRITE(KIO6,3001)NG                                                00013800
   7        STOP                                                              00013900
   3    310 IF (NN.EQ.12)GO TO 320                                            00014000
   7        WRITE(KIO6,3002)                                                  00014100
   7        STOP                                                              00014200
   3    320 IF (NN.EQ.11)GO TO 20                                             00014300
   7        IF (INDNL.NE.1) GO TO 329                                         00014400
   7        IF (MODEL.GE.3) GO TO 20                                          00014500
   3    329 WRITE(KIO6,3003)                                                  00014600
   7        STOP                                                              00014700
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   1  C                                                                      00014800
   1  C                                                                      00014900
   4     20 CALL ELEMNT                                                      00015000
   1  C                                                                      00015100
   3    150 IF (MIDEST.GT.MAXEST) MAXEST=MIDEST                             00015200
   1  C                                                                      00015300
   7        WRITE (NN) MIDEST,NPAR,(A(I),I=1,MIDEST)                        00015400
   1  C                                                                      00015500
   3    100 CONTINUE                                                        00015600
   1  C                                                                      00015700
   7        RETURN                                                          00015800
   1  C                                                                      00015900
   2   1000 FORMAT (20I4)                                                   00016000
   2   2000 FORMAT (1H1,36HE L E M E N T   G R O U P   D A T A ///)         00016100
   2   2005 FORMAT (1H1)                                                    00016200
   2   2011 FORMAT (27H E L E M E N T   G R O U P ,27(1H.),2H =,I5,4X,      00016300
   6      1         10H( LINEAR )///)                                      00016400
   2   2012 FORMAT (27H E L E M E N T   G R O U P ,27(1H.),2H =,I5,4X,      00016500
   6      1         13H( NONLINEAR )///)                                   00016600
   2   3000 FORMAT (///40H **STOP** NPAR(10) OR NPAR(11) .GT. 4            )00016700
   2   3001 FORMAT (///41H **STOP** ITERATION NOT POSSIBLE IN EL GP,I5)     00016800
   2   3002 FORMAT (//47H *STOP* NONLINEAR MATERIAL MODEL USED IN LINEAR    00016900
   6      1           15H ELEMENT GROUP  )                                 00017000
   2   3003 FORMAT (//51H *STOP* NPAR(15) AND NPAR(3) INCOMPATIBLE IN EL GRP )00017100
   2   3004 FORMAT (//51H *STOP* NPAR(5) AND NPAR(15) INCOMPATIBLE IN EL GRP )00017200
   1  C                                                                      00017300
   7        END                                                             00017400
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7        SUBROUTINE ELEMNT                                                    00000100
   7        IMPLICIT REAL*8(A-H,O-Z)                                             00000200
   1  C                                                                          00000300
   1  C                                                                          00000400
   7        COMMON /EL/ IND,ICOUNT,NPAR(20),NUMEG,NEGL,NEGNL,IMASS,IDAMP,ISTAT00000500
   6       1           ,NDOF,KLIN,IEIG,IMASSN,IDAMPN                            00000600
   1  C                                                                          00000700
   7        NPAR1=NPAR(1)                                                        00000800
   7        GO TO (1,2,3) ,NPAR1                                                 00000900
   1  C                                                                          00001000
   1  1     CALL TRUSS                                                           00001100
   7        RETURN                                                               00001200
   1  C                                                                          00001300
   1  2     CALL TODMFE                                                          00001400
   7        RETURN                                                               00001500
   1  C                                                                          00001600
   1  3     CALL THREDM                                                          00001700
   7        RETURN                                                               00001800
   1  C                                                                          00001900
   7        END                                                                  00002000
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8
   7        SUBROUTINE ELPAL (PROP,SIG,EPS,YIELD,IPEL)                          00000100
   7        IMPLICIT REAL*8(A-H,O-Z)                                            00000200
   1  C                                                                         00000300
   1  C                                                                         00000400
   1  C . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 00000500
   1  C .                                                                     . 00000600
   1  C .      IST     NUMBER OF STRESS COMPONENTS                            . 00000700
   1  C .      ISR     NUMBER OF STRAIN COMPONENTS                            . 00000800
   1  C .      SIG     STRESSES AT THE END OF THE PREVIOUS UPDATE             . 00000900
   1  C .      EPS     STRAINS  AT THE END OF THE PREVIOUS UPDATE             . 00001000
   1  C .      RATIO   PART OF STRAIN INCREMENT TAKEN ELASTICALLY             . 00001100
   1  C .      DELEPS  INCREMENT IN STRAINS                                   . 00001200
   1  C .      DELSIG  INCREMENT IN STRESSES, ASSUMING ELASTIC BEHAVIOR       . 00001300
   1  C .                                                                     . 00001400
   1  C .      PROP(1) YOUNG S  MODULUS                                       . 00001500
   1  C .      PROP(2) POISSON S  RATIO                                       . 00001600
   1  C .      PROP(3) INITIAL YIELD STRESS IN SIMPLE TENSION                 . 00001700
   1  C .      PROP(4) STRAIN HARDENING MODULUS                               . 00001800
   1  C .                                                                     . 00001900
   1  C .      IPEL    = 1, MATERIAL ELASTIC                                  . 00002000
   1  C .              = 2, MATERIAL PLASTIC                                  . 00002100
   1  C .                                                                     . 00002200
   1  C . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 00002300
   1  C                                                                         00002400
   7        COMMON/BBCCIO/KIO1,KIO2,KIO3,KIO4,KIO5,KIO6,KIO7,KIO8,KIO9,        00002500
   6       1KIO10,KIO11,KIO12,KIO13,KIO14,KIO15,KIO16,KIO17,KIO18,KIO19,KIO20,00002600
   6       2KIO21,KIO22,KIO23,KIO24,KIO25,KIO26,KIO27,KIO28,KIO29,KIO30        00002700
   1  C                                                                         00002800
   7        COMMON /EL/ IND,ICOUNT,NPAR(20),NUMEG,NEGL,NEGNL,IMASS,IDAMP,ISTAT00002900
   6       1           ,NDOF,KLIN,IEIG,IMASSN,IDAMPN                          00003000
   7        COMMON /VAR   / NG,KPRI,MODEX,KSTEP,ITE,ITEMAX,IREF,IEQREF,INOCMD 00003100
   7        COMMON /VMISES/ A1,B1,C1,D1,A2,B2,C2,D2,YLD,BM,ISR,IST             00003200
   7        COMMON /MATMOD/ STRESS(4),STRAIN(4),C(4,4),IPT,NEL                 00003300
   7        COMMON /DISDER/ DISD(5)                                            00003400
   1  C                                                                         00003500
   7        DIMENSION PROP(1),SIG(1),EPS(1)                                    00003600
   7        DIMENSION TAU(4),DELSIG(4),DELEPS(4),DEPS(4),STATE(2)              00003700
   1  C                                                                         00003800
   7        EQUIVALENCE (NPAR(3),INDNL),(NPAR(5),ITYP2D),(DELEPS(1),DEPS(1))   00003900
   1  C                                                                         00004000
   7        DATA NGLAST/1000/, STATE/1HE,1HP/                                  00004100
   1  C                                                                         00004200
   1  C                                                                         00004300
   1  C                                                                         00004400
   7        IF (IPT.NE.1) GO TO 110                                            00004500
   1  C                                                                         00004600
   7        IST=4                                                              00004700
   7        IF (ITYP2D.EQ.2) IST=3                                             00004800
   7        ISR=3                                                              00004900
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7        IF (ITYP2D.EQ.0) ISR=4                                          00005000
   7        YM=PROP(1)                                                      00005100
   7        PV=PROP(2)                                                      00005200
   7        D1=PV/(PV - 1.)                                                 00005300
   7        A2=YM/(1.+PV)                                                   00005400
   7        B2=(1.-PV)/(1.-2.*PV)                                           00005500
   7        C2=PV/(1.-2.*PV)                                                00005600
   7        D2=YM*PROP(4)/(YM-PROP(4))                                      00005700
   7        C1=A2/2.                                                        00005800
   7        BM=YM/(1. - 2.*PV)/3.                                           00005900
   1  C                                                                     00006000
   7        IF (ITYP2D.EQ.2) GO TO 105                                      00006100
   1  C                                                                     00006200
   1  C     PLANE STRAIN  /  AXISYMMETRIC                                   00006300
   7        B1=A2*C2                                                        00006400
   7        A1=B1+A2                                                        00006500
   7        GO TO 110                                                       00006600
   1  C                                                                     00006700
   1  C     PLANE STRESS                                                    00006800
   3    105 A1=YM/(1.-PV*PV)                                                00006900
   7        B1=A1*PV                                                        00007000
   1  C                                                                     00007100
   3    110 YLD = YIELD                                                     00007200
   1  C                                                                     00007300
   1  C                                                                     00007400
   1  C        1. CALCULATE INCREMENTAL STRAINS                             00007500
   1  C                                                                     00007600
   7        DO 120 I=1,ISR                                                  00007700
   3    120 DELEPS(I) = STRAIN(I) - EPS(I)                                  00007800
   1  C                                                                     00007900
   1  C                                                                     00008000
   1  C        2. CALCULATE THE STRESS INCREMENT,                           00008100
   1  C           ASSUMING ELASTIC BEHAVIOR                                 00008200
   1  C                                                                     00008300
   7        DELSIG(1) = A1*DELEPS(1) + B1*DELEPS(2)                         00008400
   7        DELSIG(2) = B1*DELEPS(1) + A1*DELEPS(2)                         00008500
   7        DELSIG(3) = C1*DELEPS(3)                                        00008600
   7        DELSIG(4) = 0.                                                  00008700
   7        IF (ITYP2D.EQ.2) GO TO 150                                      00008800
   7        DELSIG(4) = B1 * (DELEPS(1)+DELEPS(2))                          00008900
   7        IF (ITYP2D.EQ.1) GO TO 150                                      00009000
   7        DELSIG(1) = DELSIG(1) + B1*DELEPS(4)                            00009100
   7        DELSIG(2) = DELSIG(2) + B1*DELEPS(4)                            00009200
   7        DELSIG(4) = DELSIG(4) + A1*DELEPS(4)                            00009300
   1  C                                                                     00009400
   1  C                                                                     00009500
   1  C        3. CALCULATE TOTAL STRESSES,                                 00009600
   1  C           ASSUMING ELASTIC BEHAVIOR                                 00009700
   1  C                                                                     00009800
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   3    150 TAU(4) = 0.                                                          00009900
   7        DO 160 I=1,IST                                                       00010000
   3    160 TAU(I) = SIG(I) + DELSIG(I)                                          00010100
   1  C                                                                          00010200
   1  C        4. CHECK WHETHER *TAU* STATE OF STRESS FALLS                      00010300
   1  C           OUTSIDE THE LOADING SURFACE                                   00010400
   1  C                                                                          00010500
   7        SM = (TAU(1)+TAU(2)+TAU(4))/3.                                       00010600
   7        SX = TAU(1) - SM                                                     00010700
   7        SY = TAU(2) - SM                                                     00010800
   7        SS = TAU(3)                                                          00010900
   7        SZ = TAU(4) - SM                                                     00011000
   1  C                                                                          00011100
   7        FT =.5*(SX*SX+SY*SY+SZ*SZ) + SS*SS - YLD*YLD/3.                      00011200
   1  C                                                                          00011300
   7        IF (FT) 170,170,300                                                  00011400
   1  C                                                                          00011500
   1  C     STATE OF STRESS WITHIN LOADING SURFACE  -  ELASTIC BEHAVIOR          00011600
   1  C                                                                          00011700
   3    170 IPEL=1                                                             00011800
   7        STRESS(4) = 0.                                                       00011900
   7        DO 180 I=1,IST                                                       00012000
   3    180 STRESS(I) = TAU(I)                                                 00012100
   7        IF (ITYP2D.EQ.2) STRAIN(4)=EPS(4) + D1*(DELEPS(1) + DELEPS(2))       00012200
   7        GO TO 400                                                            00012300
   1  C                                                                          00012400
   1  C                                                                          00012500
   1  C     STATE OF STRESS OUTSIDE LOADING SURFACE  -  PLASTIC BEHAVIOR         00012600
   1  C                                                                          00012700
   3    300 IF (IPEL.EQ.1) GO TO 320                                           00012800
   1  C                                                                          00012900
   1  C     .......WAS PLASTIC                                                  00013000
   1  C                                                                          00013100
   7        IPEL=2                                                               00013200
   7        RATIO = 0.                                                           00013300
   7        DO 315 I=1,IST                                                       00013400
   3    315 TAU(I) = SIG(I)                                                    00013500
   7        GO TO 370                                                            00013600
   1  C                                                                          00013700
   1  C     .......WAS ELASTIC                                                   00013800
   1  C                                                                          00013900
   1  C     DETERMINE PART OF STRAIN TAKEN ELASTICLY                             00014000
   1  C                                                                          00014100
   3    320 IPEL=2                                                             00014200
   1  C                                                                          00014300
   7        SM = (SIG(1)+SIG(2)+SIG(4))/3.                                       00014400
   7        SX = SIG(1) - SM                                                     00014500
   7        SY = SIG(2) - SM                                                     00014600
   7        SS = SIG(3)                                                          00014700
```

```
START
 COL  ~---+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7          SZ = SIG(4) - SM                                                    00014800
   1  C                                                                           00014900
   7          DM = (DELSIG(1)+DELSIG(2)+DELSIG(4))/3.                             00015000
   7          DX = DELSIG(1) - DM                                                 00015100
   7          DY = DELSIG(2) - DM                                                 00015200
   7          DS = DELSIG(3)                                                      00015300
   7          DZ = DELSIG(4) - DM                                                 00015400
   1  C                                                                           00015500
   7          A = DX*DX + DY*DY + 2.*DS*DS + DZ*DZ                                00015600
   7          B = SX*DX + SY*DY + 2.*SS*DS + SZ*DZ                                00015700
   7          E= SX*SX + SY*SY + 2.*SS*SS + SZ*SZ - 2.*YLD*YLD/3.                 00015800
   1  C                                                                           00015900
   7          RATIO=(-B+DSQRT(B*B-A*E))/A                                         00016000
   1  C                                                                           00016100
   7          DO 350 I=1,IST                                                      00016200
   3    350 TAU (I) = SIG(I) + RATIO*DELSIG(I)                                    00016300
   7          IF (ITYP2D.EQ.2) STRAIN(4)=EPS(4) + RATIO*D1*(DELEPS(1)             00016400
   6         1                    + DELEPS(2))                                    00016500
   1  C                                                                           00016600
   1  C     *TAU* NOW CONTAINS (PREVIOUS STRESSES +                               00016700
   1  C     STRESSES DUE TO ELASTIC STRAIN INCREMENTS)                            00016800
   1  C                                                                           00016900
   1  C                                                                           00017000
   1  C        5. CALCULATE PLASTIC STRESSES                                      00017100
   1  C                                                                           00017200
   1  C     DETERMINE INCREMENT INTERVAL                                          00017300
   3    370 M=20.*DSQRT(FT)/YLD+1                                                 00017400
   7          IF (M.GT.30) M=30                                                   00017500
   7          XM = (1. - RATIO)/M                                                 00017600
   1  C                                                                           00017700
   1  C                                                                           00017800
   7          DO 380 I=1,ISR                                                      00017900
   3    380 DEPS(I) = XM*DELEPS(I)                                                00018000
   1  C                                                                           00018100
   1  C                                                                           00018200
   1  C     .....  CALCULATION OF ELASTOPLASTIC STRESSES  .....  (START)          00018300
   1  C                                                                           00018400
   7          DO 600 IM=1,M                                                       00018500
   1  C                                                                           00018600
   7          CALL MIDEP (TAU,DEPS,C)                                             00018700
   1  C                                                                           00018800
   7          DO 560 I=1,IST                                                      00018900
   7          DO 560 J=1,ISR                                                      00019000
   3    560 TAU(I) = TAU(I) + C(I,J) * DEPS(J)                                    00019100
   1  C                                                                           00019200
   1  C                                                                           00019300
   1  C     CORRECTION                                                            00019400
   1  C                                                                           00019500
   7          DM = (TAU(1)+TAU(2)+TAU(4))/3.                                      00019600
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7        DX = TAU(1) - DM                                                    00019700
   7        DY = TAU(2) - DM                                                    00019800
   7        DS = TAU(3)                                                         00019900
   7        DZ = TAU(4) - DM                                                    00020000
   1  C                                                                         00020100
   7        IF (PROP(4).EQ.0.) GO TO 580                                        00020200
   1  C                                                                         00020300
   1  C     STRAIN-HARDENING MATERIAL  -  UPDATE YLD                            00020400
   1  C                                                                         00020500
   7        YLD=DSQRT (1.5*(DX*DX+DY*DY+2.*DS*DS+DZ*DZ))                        00020600
   7        GO TO 600                                                           00020700
   1  C                                                                         00020800
   1  C     PERFECTLY PLASTIC MATERIAL                                          00020900
   1  C                                                                         00021000
   3    580 FTA=.5*(DX*DX + DY*DY + DZ*DZ) + DS*DS                              00021100
   7        FTB=(YLD*YLD)/3.                                                    00021200
   7        FT=FTA - FTB                                                        00021300
   7        IF (FT.EQ.0) GO TO 600                                              00021400
   7        IF (ITYP2D.EQ.2) GO TO 590                                          00021500
   1  C                                                                         00021600
   7        COEF=-1. +DSQRT(FTB/FTA)                                            00021700
   7        TAU(1) = TAU(1) + COEF*DX                                           00021800
   7        TAU(2) = TAU(2) + COEF*DY                                           00021900
   7        TAU(3) = TAU(3) + COEF*DS                                           00022000
   7        TAU(4)=TAU(4) + COEF*DZ                                             00022100
   7        GO TO 600                                                           00022200
   1  C                                                                         00022300
   3    590 COEF=DSQRT(FTB/FTA)                                                 00022400
   7        TAU(1)=TAU(1)*COEF                                                  00022500
   7        TAU(2)=TAU(2)*COEF                                                  00022600
   7        TAU(3)=TAU(3)*COEF                                                  00022700
   7        STRAIN(4)=STRAIN(4) + (COEF - 1.)*DM/BM                             00022800
   1  C                                                                         00022900
   3    600 CONTINUE                                                            00023000
   1  C                                                                         00023100
   1  C     .....  CALCULATION OF ELASTOPLASTIC STRESSES  .....  ( END )       00023200
   1  C                                                                         00023300
   7        STRESS(4) = 0.                                                      00023400
   7        DO 390 I=1,IST                                                      00023500
   3    390 STRESS(I) = TAU(I)                                                  00023600
   1  C                                                                         00023700
   1  C                                                                         00023800
   1  C        6. UPDATING STRESSES, STRAINS, YIELD, NS                         00023900
   1  C                                                                         00024000
   3    400 DO 410 I=1,IST                                                      00024100
   3    410 SIG(I) = STRESS(I)                                                  00024200
   7        DO 420 I=1,ISR                                                      00024300
   3    420 EPS(I) = STRAIN(I)                                                  00024400
   7        YIELD = YLD                                                         00024500
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7        IF (ITYP2D.EQ.2) EPS(4)=STRAIN(4)                                00024600
   7        IF (KPRI.EQ.0) GO TO 700                                         00024700
   1  C                                                                      00024800
   1  C                                                                      00024900
   7        IF (ICOUNT.EQ.3) RETURN                                          00025000
   1  C                                                                      00025100
   1  C        7. FORM THE MATERIAL LAW                                      00025200
   1  C                                                                      00025300
   7        IF (IPEL.EQ.1) GO TO 450                                         00025400
   1  C                                                                      00025500
   1  C     ELASTO-PLASTIC                                                   00025600
   7        CALL MIDEP (TAU,DEPS,C)                                          00025700
   7        RETURN                                                           00025800
   1  C                                                                      00025900
   1  C     ELASTIC                                                          00026000
   3    450 DO 460 I=1,ISR                                                   00026100
   7        DO 460 J=1,ISR                                                   00026200
   3    460 C(I,J)=0.                                                        00026300
   7        C(1,1)=A1                                                        00026400
   7        C(2,1)=B1                                                        00026500
   7        C(1,2)=B1                                                        00026600
   7        C(2,2)=A1                                                        00026700
   7        C(3,3)=C1                                                        00026800
   7        IF (ITYP2D.NE.0) RETURN                                          00026900
   7        C(1,4)=B1                                                        00027000
   7        C(2,4)=B1                                                        00027100
   7        C(4,1)=B1                                                        00027200
   7        C(4,2)=B1                                                        00027300
   7        C(4,4)=A1                                                        00027400
   1  C                                                                      00027500
   7        RETURN                                                           00027600
   1  C                                                                      00027700
   1  C                                                                      00027800
   1  C     P R I N T I N G   O F   S T R E S S E S                          00027900
   1  C                                                                      00028000
   3    700 IF (IPEL.EQ.1) GO TO 705                                         00028100
   1  C                                                                      00028200
   7        DM=(STRESS(1) + STRESS(2) + STRESS(4))/3.                        00028300
   7        DX=STRESS(1) - DM                                                00028400
   7        DY=STRESS(2) - DM                                                00028500
   7        DS=STRESS(3)                                                     00028600
   7        DZ=STRESS(4) - DM                                                00028700
   7        FT=.5*(DX*DX + DY*DY + DZ*DZ) + DS*DS - YLD*YLD/3.               00028800
   1  C                                                                      00028900
   3    705 IF (INDNL.NE.2) GO TO 800                                        00029000
   1  C                                                                      00029100
   1  C     IN TOTAL LAGRANGIAN FORMULATION,                                 00029200
   1  C     CAUCHY STRESSES ARE CALCULATED AND PRINTED                       00029300
   1  C                                                                      00029400
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7        CALL CAUCHY                                                          00029500
   1  C                                                                          00029600
   3    800 IF (NG.NE.NGLAST) GO TO 802                                          00029700
   7        IF (NEL.GT.NELAST) GO TO 806                                         00029800
   7        IF (IPT-1) 810,808,810                                               00029900
   1  C                                                                          00030000
   3    802 NGLAST = NG                                                          00030100
   3    808 IF (ITYP2D) 803,805,803                                              00030200
   3    803 WRITE(KIO6,2002)                                                     00030300
   7        GO TO 806                                                            00030400
   3    805 WRITE(KIO6,2003)                                                     00030500
   1  C                                                                          00030600
   3    806 NELAST=NEL                                                           00030700
   7        WRITE(KIO6,2004) NEL                                                 00030800
   3    810 CALL MAXMIN (STRESS,SX,SY,SM)                                        00030900
   7        IF (ITYP2D) 813,815,813                                              00031000
   1  C                                                                          00031100
   3    813 WRITE(KIO6,2005) IPT,STATE(IPEL),                                    00031200
   6       1               (STRESS(I),I=1,3),SX,SY,SM,FT                       00031300
   7        RETURN                                                               00031400
   1  C                                                                          00031500
   3    815 WRITE(KIO6,2007) IPT,STATE(IPEL),STRESS(4),                          00031600
   6       1               (STRESS(I),I=1,3),SX,SY,SM,FT                       00031700
   7        RETURN                                                               00031800
   1  C                                                                          00031900
   1  C                                                                          00032000
   2   2002 FORMAT (90H ELEMENT  STRESS      STRESS-YY     STRESS-ZZ     STRES00032100
   6       1S-YZ       MAX STRESS    MIN STRESS,21X,5HYIELD/99H NUM/IPT   STAT00032200
   6       2E                                                                   00032300
   6       3             ANGLE,9X,8HFUNCTION / )                                00032400
   2   2003 FORMAT ( 104H ELEMENT  STRESS      STRESS-XX     STRESS-YY     STR00032500
   6       1ESS-ZZ     STRESS-YZ       MAX STRESS    MIN STRESS,21X,5HYIELD / 00032600
   6       2 113H NUM/IPT   STATE                                               00032700
   6       3                                                ANGLE,9X,8HFUNCTION00032800
   6       4 / )                                                                00032900
   2   2004 FORMAT (I4/)                                                        00033000
   2   2005 FORMAT (5X,I2,2X,A1,6HLASTIC,1X,3E14.6,3X,2E14.6,3X,F6.2,3X,E14.6)00033100
   2   2007 FORMAT (5X,I2,2X,A1,6HLASTIC,1X,4E14.6,3X,2E14.6,3X,F6.2,3X,E14.6)00033200
   1  C                                                                          00033300
   7        END                                                                  00033400
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7        SUBROUTINE ELTTT6(NATP,IATP)                                        00000100
   7        IMPLICIT REAL*8(A-H,O-Z)                                            00000200
   7        COMMON /MATMOD/ STRESS(4),STRAIN(4),D(4,4),IPT,NEL                  00000300
   7        COMMON A(1)                                                         00000400
   7        DIMENSION NATP(1)                                                   00000500
   7        IATP=NATP(NEL)                                                      00000600
   7        RETURN                                                              00000700
   7        END                                                                 00000800
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7        SUBROUTINE ELT2D3                                                    00000100
   7        IMPLICIT REAL*8(A-H,O-Z)                                             00000200
   1  C                                                                          00000300
   1  C                                                                          00000400
   1  C                        M O D E L  =  3                                   00000500
   1  C                                                                          00000600
   1  C     V A R I A B L E  T A N G E N T  M O D U L I  M O D E L               00000700
   1  C                                                                          00000800
   1  C                                                                          00000900
   7        COMMON /EL/ IND,ICOUNT,NPAR(20),NUMEG,NEGL,NEGNL,IMASS,IDAMP,ISTAT 00001000
   6       1           ,NDOF,KLIN,IEIG,IMASSN,IDAMPN                           00001100
   7        COMMON /DIMEL/ N101,N102,N103,N104,N105,N106,N107,N108,N109,N110,   00001200
   6       1               N111,N112,N113,N114,N120,N121,N122,N123,N124,N125    00001300
   7        COMMON /MATMOD/ STRESS(4),STRAIN(4),D(4,4),IPT,NEL                  00001400
   7        COMMON A(1)                                                         00001500
   7        DIMENSION IA(1)                                                     00001600
   1  C                                                                         00001700
   7        EQUIVALENCE (NPAR(10),NINT)                                         00001800
   7        EQUIVALENCE (A(1),IA(1))                                            00001900
   1  C                                                                         00002000
   1  C     FOR ADDRESSES N101,N102,N103,...  SEE SUBROUTINE TODMFE             00002100
   1  C                                                                         00002200
   7        IF (IND.NE.0) GO TO 100                                             00002300
   1  C                                                                         00002400
   1  C                                                                         00002500
   1  C     I N I T I A L I Z E  W A  W O R K I N G  A R R A Y                   00002600
   1  C                                                                         00002700
   1  C                                                                         00002800
   7        IDW=10                                                              00002900
   7        NPT=NINT*NINT                                                       00003000
   7        NN=N110 + (NEL-1)*IDW*NPT                                           00003100
   7        CALL IVTMOD (A(NN),A(NN),IDW,NPT)                                   00003200
   7        RETURN                                                              00003300
   1  C                                                                         00003400
   1  C                                                                         00003500
   1  C     F I N D  S T R E S S - S T R A I N  L A W  A N D  S T R E S S       00003600
   1  C                                                                         00003700
   1  C                                                                         00003800
   3    100 IDW=10                                                            00003900
   7        NCON=7                                                              00004000
   1  C     MATP=IA(N107 + NEL - 1)                                             00004100
   7        MATP=IA(2*N107+NEL-2)                                               00004200
   7        NM=N109 + (MATP-1) * NCON                                           00004300
   7        NPT=NINT*NINT                                                       00004400
   7        NN=N110 + (NEL - 1)*IDW*NPT + (IPT - 1)*IDW                         00004500
   1  C                                                                         00004600
   7        CALL VTMOD (A(NM),A(NN),A(NN+4),A(NN+8),A(NN+9))                    00004700
   1  C                                                                         00004800
   1  C                                                                         00004900
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7          RETURN                                                      00005000
   1  C                                                                   00005100
   7          END                                                         00005200
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7        SUBROUTINE ELT2D4                                                      00000100
   7        IMPLICIT REAL*8(A-H,O-Z)                                               00000200
   1  C                                                                            00000300
   1  C                                                                            00000400
   1  C              M O D E L  =  4  O R  5                                       00000500
   1  C                                                                            00000600
   1  C     C U R V E  D E S C R I P T I O N  M O D E L                            00000700
   1  C     W I T H  O R  W I T H O U T  T E N S I O N  C U T - O F F              00000800
   1  C                                                                            00000900
   1  C                                                                            00001000
   7        COMMON /EL/ IND,ICOUNT,NPAR(20),NUMEG,NEGL,NEGNL,IMASS,IDAMP,ISTAT00001100
   6       1           ,NDOF,KLIN,IEIG,IMASSN,IDAMPN                            00001200
   7        COMMON /DIMEL/ N101,N102,N103,N104,N105,N106,N107,N108,N109,N110,  00001300
   6       1               N111,N112,N113,N114,N120,N121,N122,N123,N124,N125   00001400
   7        COMMON /MATMOD/ STRESS(4),STRAIN(4),D(4,4),IPT,NEL                     00001500
   7        COMMON A(1)                                                            00001600
   7        DIMENSION IA(1)                                                        00001700
   1  C                                                                            00001800
   7        EQUIVALENCE (NPAR(7),MXNODS),(NPAR(10),NINT),(NPAR(17),NCON)           00001900
   7        EQUIVALENCE (A(1),IA(1))                                               00002000
   1  C                                                                            00002100
   1  C     FOR ADDRESSES N101,N102,N103,...  SEE SUBROUTINE TODMFE                00002200
   1  C                                                                            00002300
   7        IF (IND.NE.0) GO TO 100                                                00002400
   1  C                                                                            00002500
   1  C                                                                            00002600
   1  C     I N I T I A L I Z E  W A  W O R K I N G  A R R A Y                      00002700
   1  C                                                                            00002800
   1  C                                                                            00002900
   7        IDW=12                                                                 00003000
   1  C     MATP=IA(N107 + NEL - 1)                                                00003100
   7        MATP=IA(2*N107+NEL-2)                                                  00003200
   7        NM=N109 + (MATP - 1)*NCON                                              00003300
   7        NDM=2*MXNODS                                                           00003400
   7        NO=(NEL - 1)*NDM                                                       00003500
   7        ND5DIM=MXNODS - 4                                                      00003600
   7        NOO=(NEL - 1)*ND5DIM                                                   00003700
   7        NPT=NINT*NINT                                                          00003800
   7        NT=IDW*NPT                                                             00003900
   7        NN=(NEL - 1)*NT                                                        00004000
   1  C                                                                            00004100
   7        CALL ICDMOD (NEL,NPT,IDW,A(NM),A(N110+NN),A(N102+NO),A(N111+NOO))      00004200
   1  C                                                                            00004300
   7        RETURN                                                                 00004400
   1  C                                                                            00004500
   1  C                                                                            00004600
   1  C     F I N D  S T R E S S - S T R A I N  L A W  A N D  S T R E S S          00004700
   1  C                                                                            00004800
   1  C                                                                            00004900
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   3    100 IDW=12                                                          00005000
   7        IP=NCON/4                                                       00005100
   1  C     MATP=IA(N107 + NEL - 1)                                         00005200
   7        MATP=IA(2*N107+NEL-2)                                           00005300
   7        NM=N109 + (MATP - 1)*NCON                                       00005400
   7        NPT=NINT*NINT                                                   00005500
   7        NN=N110 + (NEL - 1)*IDW*NPT + (IPT - 1)*IDW                     00005600
   1  C                                                                     00005700
   7        CALL CDMOD (NEL,A(NM),A(NM+IP),A(NM+2*IP),A(NM+3*IP),A(NN),     00005800
   6       1           A(NN+4),A(NN+8),A(NN+9),A(NN+10),A(NN+11),STRESS,   00005900
   6       2           STRAIN,D,IPT)                                       00006000
   1  C                                                                     00006100
   1  C                                                                     00006200
   7        RETURN                                                          00006300
   1  C                                                                     00006400
   7        END                                                             00006500
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8
   7          SUBROUTINE ELT2D6                                                    00000100
   7          IMPLICIT REAL*8(A-H,O-Z)                                             00000200
   1  C                                                                            00000300
   1  C                                                                            00000400
   1  C                        M O D E L  =  6                                     00000500
   1  C                                                                            00000600
   1  C      E L A S T O P L A S T I C   M O D E L   (VON MISES)                    00000700
   1  C                                                                            00000800
   7          COMMON /EL/ IND,ICOUNT,NPAR(20),NUMEG,NEGL,NEGNL,IMASS,IDAMP,ISTAT00000900
   6         1           ,NDOF,KLIN,IEIG,IMASSN,IDAMPN                            00001000
   7          COMMON /DIMEL/ N101,N102,N103,N104,N105,N106,N107,N108,N109,N110,  00001100
   6         1               N111,N112,N113,N114,N120,N121,N122,N123,N124,N125    00001200
   7          COMMON /MATMOD/ STRESS(4),STRAIN(4),D(4,4),IPT,NEL                   00001300
   7          COMMON A(1)                                                          00001400
   1  C                                                                            00001500
   7          DIMENSION IA(1)                                                      00001600
   7          EQUIVALENCE (A(1),IA(1))                                             00001700
   7          EQUIVALENCE (NPAR(10),NINT)                                          00001800
   1  C                                                                            00001900
   1  C      FOR ADDRESSES N101,N102,N103,...  SEE SUBROUTINE TODMFE               00002000
   1  C                                                                            00002100
   7          IF (IND.NE.0) GO TO 100                                              00002200
   1  C                                                                            00002300
   1  C                                                                            00002400
   1  C      I N I T I A L I Z E  W A  W O R K I N G  A R R A Y                     00002500
   1  C                                                                            00002600
   1  C                                                                            00002700
   7          IDW=10                                                               00002800
   7          NPT=NINT*NINT                                                        00002900
   7          NN=N110 + (NEL - 1)*NPT*IDW                                          00003000
   1  C       CALL ELTTT6(A(N107),MATP)                                            00003100
   7          MATP=IA(2*N107+NEL-2)                                                00003200
   7          NM=N109 + (MATP - 1)*4                                               00003300
   7          DO 10 J=1,NPT                                                        00003400
   4     10   CALL IELPAL(A(NN),A(NN+9+(J-1)*10),A(NM),NPT,J)                      00003500
   7          RETURN                                                               00003600
   1  C                                                                            00003700
   1  C                                                                            00003800
   1  C      F I N D   S T R E S S - S T R A I N   L A W   A N D   S T R E S S     00003900
   1  C                                                                            00004000
   1  C                                                                            00004100
   3    100 IDW=10                                                                 00004200
   7          NPT=NINT*NINT                                                        00004300
   7          NN=N110 + (NEL - 1)*NPT*IDW + (IPT - 1)*IDW                          00004400
   1  C       CALL ELTTT6(A(N107),MATP)                                            00004500
   7          MATP=IA(2*N107+NEL-2)                                                00004600
   7          NM=N109 + (MATP - 1)*4                                               00004700
   1  C                                                                            00004800
   7          CALL ELPAL (A(NM),A(NN),A(NN+4),A(NN+8),A(NN+9))                     00004900
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   1  C                                                                    00005000
   7        RETURN                                                        00005100
   1  C                                                                    00005200
   1  C                                                                    00005300
   7        END                                                           00005400
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7          SUBROUTINE ELT2D7                                                    00000100
   7          IMPLICIT REAL*8(A-H,O-Z)                                             00000200
   1  C                                                                            00000300
   1  C       M O D E L    =    7                                                   00000400
   1  C                                                                            00000500
   1  C       E L A S T O P L A S T I C   M O D E L   (DRUCKER-PRAGER)              00000600
   1  C                                                                            00000700
   1  C                                                                            00000800
   7          COMMON /EL     / IND,ICOUNT,NPAR(20),NUMEG,NEGL,NEGNL,IMASS,IDAMP,  00000900
   6         1                 ISTAT,NDOF,KLIN,IEIG,IMASSN,IDAMPN                 00001000
   7          COMMON /DIMEL / N101,N102,N103,N104,N105,N106,N107,N108,N109,N110,   00001100
   6         1              N111,N112,N113,N114,N120,N121,N122,N123,N124,N125      00001200
   7          COMMON /MATMOD/ STRESS(4),STRAIN(4),D(4,4),IPT,NEL                   00001300
   7          COMMON A(1)                                                          00001400
   7          DIMENSION IA(1)                                                      00001500
   1  C                                                                            00001600
   7          EQUIVALENCE (NPAR(10),NINT)                                          00001700
   7          EQUIVALENCE (A(1),IA(1))                                             00001800
   1  C                                                                            00001900
   1  C       FOR ADDRESSES N101,N102,N103,...  SEE SUBROUTINE TODMFE              00002000
   1  C                                                                            00002100
   1  C                                                                            00002200
   7          IDW=11                                                               00002300
   7          NCON=4                                                               00002400
   7          NPT=NINT*NINT                                                        00002500
   1  C       MATP=IA(N107 + NEL - 1)                                              00002600
   7          MATP=IA(2*N107+NEL-2)                                                00002700
   7          NM=N109 + (MATP-1) * NCON                                            00002800
   7          IF (IND.NE.0) GO TO 100                                              00002900
   1  C                                                                            00003000
   1  C                                                                            00003100
   1  C       I N I T I A L I Z E  W A  W O R K I N G  A R R A Y                    00003200
   1  C                                                                            00003300
   7          NN=N110 + (NEL - 1)*NPT*IDW                                          00003400
   1  C                                                                            00003500
   7          CALL IDRUCK (A(NN),A(NN),A(NM),NPT,IDW)                              00003600
   1  C                                                                            00003700
   7          RETURN                                                               00003800
   1  C                                                                            00003900
   1  C                                                                            00004000
   1  C       F I N D  S T R E S S - S T R A I N  L A W  A N D  S T R E S S        00004100
   1  C                                                                            00004200
   1  C                                                                            00004300
   3    100 NS=N110 + ( (NEL-1)*NPT + (IPT-1) ) * IDW                              00004400
   1  C                                                                            00004500
   7          CALL DRUCK  (A(NM),A(NS),A(NS+4),A(NS+8),A(NS+9),A(NS+10))           00004600
   1  C                                                                            00004700
   7          RETURN                                                               00004800
   1  C                                                                            00004900
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   1  C                                                                    00005000
   7        END                                                            00005100
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8
    7         SUBROUTINE ELT2D8                                                    00000100
    7         IMPLICIT REAL*8(A-H,O-Z)                                             00000200
    1  C                                                                           00000300
    1  C                                                                           00000400
    1  C                          M O D E L  =  8                                  00000500
    1  C                                                                           00000600
    1  C      I N C O M P R E S S I B L E  E L A S T I C  M O D E L                 00000700
    1  C                                                                           00000800
    1  C                                                                           00000900
    7         COMMON /EL/ IND,ICOUNT,NPAR(20),NUMEG,NEGL,NEGNL,IMASS,IDAMP,ISTAT00001000
    6        1           ,NDOF,KLIN,IEIG,IMASSN,IDAMPN                           00001100
    7         COMMON /DIMEL/ N101,N102,N103,N104,N105,N106,N107,N108,N109,N110,  00001200
    6        1              N111,N112,N113,N114,N120,N121,N122,N123,N124,N125     00001300
    7         COMMON /MATMOD/ STRESS(4),STRAIN(4),D(4,4),IPT,NEL                   00001400
    7         COMMON A(1)                                                          00001500
    7         DIMENSION IA(1)                                                      00001600
    1  C                                                                           00001700
    7         EQUIVALENCE (A(1),IA(1))                                             00001800
    1  C                                                                           00001900
    1  C      FOR ADDRESSES N101,N102,N103,...  SEE SUBROUTINE TODMFE              00002000
    1  C                                                                           00002100
    1  C                                                                           00002200
    1  C      F I N D  S T R E S S - S T R A I N  L A W  A N D  S T R E S S        00002300
    1  C                                                                           00002400
    1  C                                                                           00002500
    7         MATP=IA((2*N107)+NEL-2)                                              00002600
    7         NM=N109 + (MATP - 1)*2                                               00002700
    7         CALL MOONEY (NEL,A(NM),D,STRESS,STRAIN,IPT)                          00002800
    1  C                                                                           00002900
    1  C                                                                           00003000
    7         RETURN                                                               00003100
    1  C                                                                           00003200
    7         END                                                                  00003300
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7          SUBROUTINE ELT2D9                                              00000100
   1  C                                                                      00000200
   1  C       M O D E L    =    9                                            00000300
   1  C                                                                      00000400
   7          RETURN                                                         00000500
   7          END                                                            00000600
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7        SUBROUTINE ELT3D3                                                      00000100
   7        IMPLICIT REAL*8(A-H,O-Z)                                               00000200
   1  C                                                                            00000300
   1  C                                                                            00000400
   1  C . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 00000500
   1  C .                                                                        . 00000600
   1  C .                  M O D E L    =    3                                   . 00000700
   1  C .                                                                        . 00000800
   1  C .   C U R V E   D E S C R I P T I O N   N O N L I N E A R   M O D E L    00000900
   1  C .                                                                        . 00001000
   1  C .   THIS OVERLAY CONTAINS ALL SUBROUTINES PERTAINING TO MODEL 3           . 00001100
   1  C .                                                                        . 00001200
   1  C . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 00001300
   1  C                                                                            00001400
   1  C                                                                            00001500
   1  C                                                                            00001600
   7        COMMON /EL/ IND,ICOUNT,NPAR(20),NUMEG,NEGL,NEGNL,IMASS,IDAMP,ISTAT   00001700
   6       1          ,NDOF,KLIN,IEIG,IMASSN,IDAMPN                              00001800
   7        COMMON /DIMEL/ N101,N102,N103,N104,N105,N106,N107,N108,N109,N110,     00001900
   6       1               N111,N112,N113,N114,N120,N121,N122,N123,N124,N125     00002000
   7        COMMON /MTMD3D/  D(6,6),STRESS(6),STRAIN(6),IPT,NEL                  00002100
   7        COMMON A(1)                                                          00002200
   7        DIMENSION IA(1)                                                      00002300
   1  C                                                                          00002400
   7        EQUIVALENCE (NPAR(10),NINT), (NPAR(11),NINTZ), (NPAR(17),NCON)       00002500
   7        EQUIVALENCE (A(1),IA(1))                                             00002600
   1  C                                                                          00002700
   1  C                                                                          00002800
   1  C     FOR ADDRESSES N101,N102,N103,.... REFER TO PROGRAM THREDM             00002900
   1  C                                                                          00003000
   1  C                                                                          00003100
   7        IF (IND.NE.0) GO TO 100                                              00003200
   1  C                                                                          00003300
   1  C                                                                          00003400
   1  C     I N I T I A L I Z E   W A   W O R K I N G   V E C T O R              00003500
   1  C                                                                          00003600
   1  C                                                                          00003700
   7        IDW=13                                                               00003800
   7        NPT=NINT*NINT*NINTZ                                                  00003900
   7        NT=IDW*NPT                                                           00004000
   7        NN=(NEL - 1)*NT                                                      00004100
   7        CALL ICMOD3 (A(N112+NN),NPT)                                         00004200
   7        RETURN                                                               00004300
   1  C                                                                          00004400
   1  C                                                                          00004500
   1  C     F I N D   S T R E S S - S T R A I N   L A W   A N D   S T R E S S    00004600
   1  C                                                                          00004700
   1  C                                                                          00004800
   3    100 IDW=13                                                             00004900
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

    7         IP=NCON/4                                                          00005000
    1  C      MATP=IA(N107 + NEL - 1)                                            00005100
    7         MATP=IA(2*N107+NFL-2)                                              00005200
    7         NM=N111 + (MATP-1)*NCON                                            00005300
    7         NPT=NINT*NINT*NINTZ                                                00005400
    7         NN=N112 + (NEL-1)*IDW*NPT + (IPT-1)*IDW                            00005500
    1  C                                                                         00005600
    7         CALL CMOD3D (NEL,A(NM),A(NM+IP),A(NM+2*IP),A(NM+3*IP),A(NN),       00005700
    6        1             A(NN+6),A(NN+12),STRESS,STRAIN,D,IPT)                00005800
    1  C                                                                         00005900
    1  C                                                                         00006000
    1  C                                                                         00006100
    7         RETURN                                                             00006200
    1  C                                                                         00006300
    7         END                                                                00006400
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7        SUBROUTINE ELT3D4                                              00000100
   1  C                                                                    00000200
   1  C     M O D E L    =    4                                            00000300
   1  C                                                                    00000400
   7        RETURN                                                         00000500
   7        END                                                            00000600
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7          SUBROUTINE ELT3D5                                          00000100
   1  C                                                                  00000200
   1  C       M O D E L      =     5                                     00000300
   1  C                                                                  00000400
   7          RETURN                                                     00000500
   7          END                                                        00000600
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7        SUBROUTINE ELT3D6                                              00000100
   1  C                                                                    00000200
   1  C     M O D E L    =     6                                           00000300
   1  C                                                                    00000400
   7        RETURN                                                         00000500
   7        END                                                            00000600
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7          SUBROUTINE EL2D10                                          00000100
   1  C                                                                  00000200
   1  C       M O D E L     =    10                                       00000300
   1  C                                                                  00000400
   7          RETURN                                                     00000500
   7          END                                                        00000600
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7        SUBROUTINE EL2D11                                                  00000100
   1  C                                                                        00000200
   1  C     M O D E L    =    11                                               00000300
   1  C                                                                        00000400
   7        RETURN                                                             00000500
   7        END                                                                00000600
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7        SUBROUTINE EL2D12                                                  00000100
   1  C                                                                        00000200
   1  C     M O D E L    =    12                                               00000300
   1  C                                                                        00000400
   7        RETURN                                                             00000500
   7        END                                                                00000600
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7        SUBROUTINE EQUIT (AA,DISPI,RE,DISP,VEL,ACC,MAXA,WV,XM)                  00000100
   7        IMPLICIT REAL*8(A-H,O-Z)                                                00000200
   1  C                                                                             00000300
   1  C . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 00000400
   1  C .                                                                         . 00000500
   1  C .   P R O G R A M                                                       . 00000600
   1  C .      . TO ITERATE FOR DYNAMIC EQUILIBRIUM                             . 00000700
   1  C .                                                                       . 00000800
   1  C .        A    = EFFECTIVE STIFFNESS MATRIX AND WORKING STORAGE          . 00000900
   1  C .        RE   = OUT OF BALANCE LOADS                                    . 00001000
   1  C .        DISP = DISPLACEMENT AT PREVIOUS TIME STEP                      . 00001100
   1  C .        DISPI= DISPLACEMENT INCREMENT AT CURRENT TIME STEP             . 00001200
   1  C .        VEL  = VELOCITY AT PREVIOUS TIME STEP                          . 00001300
   1  C .        ACC  = ACCELERATION AT PREVIOUS TIME STEP                      . 00001400
   1  C .        MAXA = ADDRESSES OF DIAGONAL ELEMENTS IN EFFECTIVE             . 00001500
   1  C .               STIFFNESS MATRIX                                        . 00001600
   1  C .        WV   = WORKING VECTOR                                          . 00001700
   1  C .                                                                       . 00001800
   1  C . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 00001900
   1  C                                                                             00002000
                                                                                    00002100
   7        COMMON A(1)                                                             00002200
   7        COMMON/BBCCIO/KIO1,KIO2,KIO3,KIO4,KIO5,KIO6,KIO7,KIO8,KIO9,            00002300
   6       1KIO10,KIO11,KIO12,KIO13,KIO14,KIO15,KIO16,KIO17,KIO18,KIO19,KIO20,     00002400
   6       2KIO21,KIO22,KIO23,KIO24,KIO25,KIO26,KIO27,KIO28,KIO29,KIO30            00002500
   7        COMMON /SOL/ NUMNP,NEQ,NWK,NWM,NWC,NUMEST,MIDEST,MAXEST,NSTE,MA        00002600
   7        COMMON /EL/ IND,ICOUNT,NPAR(20),NUMEG,NEGL,NEGNL,IMASS,IDAMP,ISTAT     00002700
   6       1           ,NDOF,KLIN,IEIG,IMASSN,IDAMPN                               00002800
   7        COMMON /CONST/ DT,DTA,A0,A1,A2,A3,A4,A5,A6,A7,A8,A9,A10,A11            00002900
   6       1              ,A12,A13,A14,A15,A16,A17,A18,A19,A20,IOPE                 00003000
   7        COMMON /DAMPNG/ ADAMP,BDAMP                                            00003100
   7        COMMON /VAR/ NG,KPRI,MODEX,KSTEP,ITE,ITEMAX,IREF,IEQREF,INOCMD         00003200
   7        COMMON /TAPES/ NREAD,NWRITE                                            00003300
   7        COMMON /NORMS/ RNORM,RENORM,RTOL                                       00003400
   7        DIMENSION AA(1),DISPI(1),RE(1),DISP(1),VEL(1),ACC(1),MAXA(1),WV(1)     00003500
   6       1          ,XM(1)                                                      00003600
   1  C                                                                             00003700
   7        ICOUNT=3                                                               00003800
   7        ITE=0                                                                  00003900
   7        DO 20 I=1,NEQ                                                          00004000
   4     20 WV(I)=DISP(I) + DISPI(I)                                               00004100
   1  C                                                                             00004200
   3    500 ITE=ITE + 1                                                          00004300
   7        REWIND KIO20                                                           00004400
   1  C                                                                             00004500
   1  C                                                                             00004600
   1  C     C A L C U L A T E   C U R R E N T   R H S   E F F E C T I V E         00004700
   1  C                                                 L O A D S                  00004800
   1  C                                                                             00004900
   7        BACKSPACE KIO13
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8
   7          READ(KIO13) (RE(I),I=1,NEQ)                                    00005000
   7          REWIND KIO14                                                   00005100
   1  C                                                                      00005200
   1  C                                                                      00005300
   1  C     C A L C U L A T E  L I N E A R  B A L A N C E D  L O A D S       00005400
   1  C                                                                      00005500
   1  C                                                                      00005600
   7          IF (NEGL.EQ.0) GO TO 100                                       00005700
   1  C                                                                      00005800
   1  C     STIFFNESS EFFECT                                                 00005900
   1  C                                                                      00006000
   7          READ(KIO14) (AA(I),I=1,NWK)                                    00006100
   7          IF (BDAMP.EQ.0.0) GO TO 29                                     00006200
   7          DO 25 I=1,NEQ                                                  00006300
   4       25 WV(I)=WV(I) + BDAMP*(A1*DISPI(I) - A4*VEL(I) - A5*ACC(I))      00006400
   4       29 CALL MULT (RE,AA,WV,MAXA,NEQ)                                 00006500
   1  C                                                                      00006600
   3      100 IF (ISTAT.EQ.0) GO TO 140                                      00006700
   1  C                                                                      00006800
   1  C     MASS EFFECT                                                      00006900
   1  C                                                                      00007000
   7          FAC1=A0 + A1*ADAMP                                             00007100
   7          FAC2=-A2 - A4*ADAMP                                            00007200
   7          FAC3=-A3 - A5*ADAMP                                            00007300
   7          DO 30 I=1,NEQ                                                  00007400
   4       30 WV(I)=FAC1*DISPI(I) + FAC2*VEL(I) + FAC3*ACC(I)                00007500
   7          IF (IMASS.EQ.2) GO TO 60                                       00007600
   7          DO 40 I=1,NEQ                                                  00007700
   4       40 RE(I)=RE(I) - WV(I)*XM(I)                                      00007800
   7          GO TO 70                                                       00007900
   4       60 READ(14) (AA(I),I=1,NWM)                                      00008000
   7          CALL MULT (RE,AA,WV,MAXA,NEQ)                                  00008100
   1  C                                                                      00008200
   1  C     DAMPING EFFECT                                                   00008300
   1  C                                                                      00008400
   4       70 IF (IDAMPN.EQ.0) GO TO 110                                    00008500
   7          READ(KIO14) (AA(I),I=1,NEQ)                                    00008600
   7          DO 90 I=1,NEQ                                                  00008700
   4       90 RE(I)=RE(I) - AA(I)*(A1*DISPI(I) - A4*VEL(I) - A5*ACC(I))     00008800
   1  C                                                                      00008900
   3      110 DO 120 I=1,NEQ                                                 00009000
   3      120 WV(I)=DISP(I) + DISPI(I)                                       00009100
   1  C                                                                      00009200
   1  C                                                                      00009300
   1  C     A D D  N O N L I N E A R  C O N T R I B U T I O N S              00009400
   1  C                                                                      00009500
   1  C                                                                      00009600
   3      140 REWIND NREAD                                                   00009700
   7          DO 200 N=1,NEGNL                                               00009800
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8
   1  C                                                                      00009900
   7        READ (NREAD) NUMEST,NPAR,(A(I),I=1,NUMEST)                       00010000
   1  C                                                                      00010100
   7        CALL ELEMNT                                                      00010200
   1  C                                                                      00010300
   3     200 CONTINUE                                                       00010400
   1  C                                                                      00010500
   1  C                                                                      00010600
   1  C     C A L C U L A T E  I N C R E M E N T  I N                        00010700
   1  C                          D I S P L A C E M E N T S                   00010800
   1  C                                                                      00010900
   1  C                                                                      00011000
   7        READ(KIO20) (AA(I),I=1,NWK)                                      00011100
   7        IF (ITE .LE. (ITEMAX/2+2)) GO TO 240                             00011200
   1  C                                                                      00011300
   1  C     CALCULATE NORM OF OUT-OF-BALANCE LOAD AND CHECK FOR DIVERGENCE   00011400
   1  C                                                                      00011500
   7        RENORM=0.                                                        00011600
   7        DO 220 I=1,NEQ                                                   00011700
   3     220 RENORM=RENORM + RE(I)*RE(I)                                     00011800
   7        IF (RENORM.LT.RNORM) GO TO 240                                   00011900
   7        WRITE(KIO6,2050)ITE                                              00012000
   7        IEQREF=1                                                         00012100
   7        ICOUNT=2                                                         00012200
   7        BACKSPACE KIO13                                                  00012300
   7        RETURN                                                           00012400
   1  C                                                                      00012500
   3     240 CALL COLSOL (AA,RE,MAXA,NEQ,MA,NWK,2)                          00012600
   1  C                                                                      00012700
   1  C                                                                      00012800
   1  C     C H E C K  F O R  C O N V E R G E N C E                          00012900
   1  C                                                                      00013000
   1  C                                                                      00013100
   7        DNORM=0.                                                         00013200
   7        DINORM=0.                                                        00013300
   7        DO 300 I=1,NEQ                                                   00013400
   7        DISPI(I)=DISPI(I) + RE(I)                                        00013500
   7        WV(I)=WV(I) + RE(I)                                              00013600
   7        DNORM=DNORM + WV(I)*WV(I)                                        00013700
   3     300 DINORM=DINORM + RE(I)*RE(I)                                     00013800
   7        TOL=DNORM*RTOL                                                   00013900
   1  C                                                                      00014000
   7        IF (DINORM.LT.TOL) GO TO 400                                     00014100
   7        IF (ITE.LT.ITEMAX) GO TO 500                                     00014200
   7        WRITE(KIO6,2010) KSTEP,ITE                                       00014300
   7        WRITE(KIO6,2020)                                                 00014400
   7        STOP                                                             00014500
   1  C                                                                      00014600
   3     400 ICOUNT=2                                                        00014700
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7        RETURN                                                              00014800
   1  C                                                                         00014900
   2   2010 FORMAT (//// 37H EQUILIBRIUM ITERATION IN TIME STEP = ,I5 //        00015000
   6      1             37H NUMBER OF ITERATIONS                 = ,I5  /)     00015100
   2   2020 FORMAT (////46H ITERATION LIMIT REACHED  S T O P  OF SOLUTION   ) 00015200
   2   2050 FORMAT (////67H OUT-OF-BALANCE LOADS LARGER THAN INCREMENTAL LOADS00015300
   6       1 AFTER ITERATION ,I5/20H STOP IF NUMREF.EQ.0 )                     00015400
   7        END                                                                00015500
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7        SUBROUTINE ERROR(N)                                                  00000100
   7        IMPLICIT REAL*8(A-H,O-Z)                                             00000200
   7        COMMON/BBCCIO/KI01,KI02,KI03,KI04,KI05,KI06,KI07,KI08,KI09,          00000300
   6       1KI010,KI011,KI012,KI013,KI014,KI015,KI016,KI017,KI018,KI019,KI020,00000400
   6       2KI021,KI022,KI023,KI024,KI025,KI026,KI027,KI028,KI029,KI030         00000500
   7        WRITE(KI06,2000) N                                                   00000600
   2   2000 FORMAT (// 32H *** ERROR   STORAGE EXCEEDED BY, I6)                  00000700
   7        STOP                                                                 00000800
   7        END                                                                  00000900
```

```
START                                                                    MOD
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8  FLAGS

   7        SUBROUTINE FREQS                                                  00000100
   7        IMPLICIT REAL*8(A-H,O-Z)                                          00000200
   1  C                                                                       00000300
   1  C                                                                       00000400
   1  C . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .  00000500
   1  C .                                                                   .  00000600
   1  C .   P R O G R A M                                                   .  00000700
   1  C .      TO FIND THE LOWEST FREQUENCIES AND ASSOCIATED                .  00000800
   1  C .      MODE SHAPES OF LINEARIZED STRUCTURE BY RAYLEIGH METHOD       .  00000901   *
   1  C .   PROGRAMED  BY S.N.MUKHERJEE. BBC,BADEN. 1975-76               .   00001000
   1  C . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .  00001100
   1  C                                                                       00001200
   7        COMMON/BBCCIO/KI01,KI02,KI03,KI04,KI05,KI06,KI07,KI08,KI09,       00001300
   6       1KI010,KI011,KI012,KI013,KI014,KI015,KI016,KI017,KI018,KI019,KI020,00001400
   6       2KI021,KI022,KI023,KI024,KI025,KI026,KI027,KI028,KI029,KI030       00001500
   1  C                                                                       00001600
   7        COMMON /SOL/ NUMNP,NEQ,NWK,NWM,NWC,NUMEST,MIDEST,MAXEST,NSTE,MA   00001700
   7        COMMON /DIM/ N1,N2,N3,N4,N5,N6,N7,N8,N9,N10,N11,N12,N13,N14,N15   00001800
   7        COMMON /EL/ IND,ICOUNT,NPAR(20),NUMEG,NEGL,NEGNL,IMASS,IDAMP,ISTAT00001900
   6       1           ,NDOF,KLIN,IEIG,IMASSN,IDAMPN                         00002000
   7        COMMON /JUNK/ HED(12),MTOT                                        00002100
   7        COMMON /VAR/ NG,KPRI,MODEX,KSTEP,ITE,ITEMAX,IREF,IEQREF,INOCMD    00002200
   7        COMMON /TAPEIG/  NSTIF,NT                                         00002300
   7        COMMON /DAMPNG/ ADAMP,BDAMP                                       00002400
   7        COMMON A(1)                                                       00002500
   1  C                                                                       00002600
   1  C     INPUT FREQUENCY CONTROL DATA                                      00002700
   1  C                                                                       00002800
   7        CALL ERRSET (213,20,10,2)                                         00002900
   7        READ(KI05,1000) NFREQ,COFQ,IFPR                                   00003000
   7        IF (NFREQ.EQ.0) NFREQ=1                                           00003100
   7        IF (COFQ.EQ.0) COFQ=1.E+08                                        00003200
   7        COFQ2=COFQ*COFQ                                                   00003300
   7        WRITE(KI06,1003) NFREQ,COFQ,IFPR                                  00003400
   1  C                                                                       00003500
   7        IF (MODEX.EQ.0) GO TO 300                                         00003600
   7        NT = KI021                                                        00003700
   7        NSTIF =KI020                                                      00003800
   1  C                                                                       00003900
   1  C     FIND PSEUDONORM OF STIFFNESS MATRIX                               00004000
   1  C                                                                       00004100
   7        CALL PNORM (A(N4),A(N1),NEQ,ANORM)                                00004200
   1  C                                                                       00004300
   1  C     CALCULATE NEW ARRAY ADDRESSES                                     00004400
   1  C                                                                       00004500
   3    300 NC=NFREQ + 5                                                      00004600
   7        M1=1                                                              00004700
   7        M2=M1 + NEQ + 1                                                   00004800
   7        M3=M2 + NC                                                        00004900
```

```
START                                                                                                    MOD
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8                    FLAGS
   7        M4=M3 + 6*NEQ                                                                    00005000
   7        M5=M4 + NEQ                                                                      00005100
   7        M6=M5 + NEQ                                                                      00005200
   7        M7=M6 + NC + 1                                                                   00005300
   7        M8=M7 + NC                                                                       00005400
   7        M9=M8 + NC                                                                       00005500
   7        M10=M9 + NC                                                                      00005600
   7        M11=M10 + NWK                                                                    00005700
   7        M12=M11 + NWM                                                                    00005800
   7        IF (M12.GT.MTOT) CALL ERROR (M12-MTOT)                                           00005900
   7        IF (MODEX.EQ.0) RETURN                                                           00006000
   1  C                                                                                      00006100
   1  C     SHIFT MAXA INTO ITS NEW STORAGE PLACE                                            00006200
   1  C                                                                                      00006300
   1  C     NN=NEQ + 1                                                                       00006400
   1  C     DO 350 I=1,NN                                                                    00006500
   7        CALL MAXAEX(A(M1),A(N1),NEQ)                                                     00006700
   1  C                                                                                      00006800
   1  C     PUT MASS MATRIX IN CORE                                                          00006900
   1  C                                                                                      00007000
   7        NN=M12 - 1                                                                       00007100
   7        IF (IMASS.EQ.1) GO TO 450                                                        00007200
   7        REWIND KIO14                                                                     00007300
   7        IF (KLIN.EQ.0 .AND. BDAMP.EQ.0.0) GO TO 440                                      00007400
   7        IF (NEGL.EQ.0) GO TO 440                                                         00007500
   7        READ(KIO14)                                                                      00007600
   3    440 READ(KIO14) (A(I),I=M11,NN)                                                      00007700
   7        GO TO 490                                                                        00007800
   3    450 BACKSPACE KIO17                                                                  00007900
   7        READ(KIO17) (A(I),I=M11,NN)                                                      00008000
   3    490 CONTINUE                                                                         00008100
   1  C                                                                                      00008200
   7        CALL SECANT (A(M10),A(M10),A(M11),A(M4),A(M1),A(M5),A(M3),A(M2),                 00008300
   6       1   A(M7),A(M8),A(M9),A(M6),NEQ,MA,NWK,NWM,NFREQ,NC,                             00008400
   6       2   IFPR,ANORM,COFQ2,1)                                                          00008500
   1  C                                                                                      00008600
   7        REWIND NT                                                                        00008700
   7        M3=M2 + NFREQ                                                                    00008800
   1  C     PRINT THE FREQUENCIES AND MODE SHAPES                                            00008900
   1  C                                                                                      00009000
   7        NN=M3 - 1                                                                        00009100
   7        READ (NT) (A(I),I=M2,NN)                                                         00009200
   7        WRITE(KIO6,1004)                                                                 00009300
   1  C                                                                                      00009400
   7        DO 500 I=1,NFREQ                                                                 00009500
   7        PI=4.0*DATAN(1.0D0)                                                              00009600
   7        ACIRC=A(M2+I-1)/(2.*PI)                                                          00009700
   7        APERD=1./ACIRC                                                                   00009800
   3    500 WRITE(KIO6,1006) I,A(M2+I-1),ACIRC,APERD                                         00009900
```

```
START                                                                                                  MOD
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8  FLAGS
   1  C                                                                                  00010000
   7        M4=M3 + NFREQ*NEQ                                                          00010100
   7        NN=M4 - 1                                                                  00010200
   7        READ (NT) (A(I),I=M3,NN)                                                   00010300
   7        CALL READID(A(M4),NDOF,NUMNP)                                              00010700
   7        DO 550 IM=1,NFREQ                                                          00010800
   3     550 CALL WRTFRQ  (A(M2),A(M3),A(M4),NUMNP,NDOF,NEQ,IM)                        00010900
   1  C                                                                                  00011000
   1  C     PUT MAXA BACK IN ITS OLD STORAGE PLACE                                     00011100
   1  C                                                                                  00011200
   1  C     NN=NEQ + 1                                                                 00011300
   1  C     DO 5 I=1,NN                                                                00011400
   1  C   5 A(N1+I-1)=A(M1+I-1)                                                         00011500
   7        CALL MAXAEX(A(N1),A(M1),NEQ)                                               00011600
   1  C                                                                                  00011700
   7        RETURN                                                                     00011800
   1  C                                                                                  00011900
   2     1000 FORMAT(I5,E10.4,I5)                                                      00012000
   2     1003 FORMAT (1H1,40H F R E Q U E N C I E S  S O L U T I O N // 3X,            00012100
   6         124H C O N T R O L  D A T A //6X,                                        00012200
   6         254H NUMBER OF FREQUENCIES TO BE DETERMINED      (NFREQ) =   I5/6X,00012300
   6         354H CUTOFF FREQUENCY                             (COFQ) =E10.4/6X,00012400
   6         454H PRINT OPTION                                 (IFPR) =   I5/6X,00012500
   6         540H       EQ.0, NO PRINTING                                 /6X, 00012600
   6         640H       EQ.1, PRINT                                       ///) 00012700
   2     1004 FORMAT (//24H   F R E Q U E N C I E S   //  2X,                      00012800
   6         121H    FREQUENCY NUMBER  ,10X,20H FREQUENCY (RAD/SEC) ,4X,          00012900
   6         224H  FREQUENCY (CYCLES/SEC),8X,16HPERIOD (SECONDS)  / )             00013000
   2     1006 FORMAT (9X,I5,24X,E11.4,15X,E11.4,15X,E11.4)                          00013100
   2     2000 FORMAT (50H BLANK COMMON TOO SMALL FOR EIGENVALUE,S NEEDS            )00013200
   1  C                                                                                  00013300
   7        END                                                                        00013400
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7        SUBROUTINE FUNCT (R,S,T,H,P,NOD9,XJ,DET,XX)                          00000100
   7        IMPLICIT REAL*8(A-H,O-Z)                                             00000200
                                                                                 00000300
   1  C                                                                          00000400
   1  C                                                                          00000500
   1  C . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 00000600
   1  C .                                                                     . 00000700
   1  C .   P R O G R A M                                                     . 00000800
   1  C .                                                                     . 00000900
   1  C .      TO FIND INTERPOLATION FUNCTIONS ( H )                          . 00001000
   1  C .      AND DERIVATIVES ( P ) CORRESPONDING TO THE NODAL               . 00001100
   1  C .      POINTS OF A CURVILINEAR ISOPARAMETRIC HEXAHEDRON               . 00001200
   1  C .      OR SUBPARAMETRIC HEXAHEDRON (8 TO 21 NODES)                    . 00001300
   1  C .                                                                     . 00001400
   1  C .      TO FIND JACOBIAN ( XJ ) AND ITS DETERMINANT ( DET )            . 00001500
   1  C .                                                                     . 00001600
   1  C . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 00001700
   1  C                                                                          00001800
   7        COMMON/BBCCIO/KIO1,KIO2,KIO3,KIO4,KIO5,KIO6,KIO7,KIO8,KIO9,         00001900
   6       1KIO10,KIO11,KIO12,KIO13,KIO14,KIO15,KIO16,KIO17,KIO18,KIO19,KIO20,  00002000
   6       2KIO21,KIO22,KIO23,KIO24,KIO25,KIO26,KIO27,KIO28,KIO29,KIO30         00002100
   1  C                                                                          00002200
   7        COMMON /THREED/ DE,IELD,IELX,IEL,NPT,IDW,NND9                       00002300
   7        DIMENSION H(1),P(3,1),NOD9(1),IPERM(8),XJ(3,3),XX(3,1)              00002400
   1  C                                                                          00002500
   7        DATA IPERM / 2,3,4,1,6,7,8,5 /                                      00002600
   1  C                                                                          00002700
   7        RP=1.0 + R                                                          00002800
   7        SP=1.0 + S                                                          00002900
   7        TP=1.0 + T                                                          00003000
   7        RM=1.0 - R                                                          00003100
   7        SM=1.0 - S                                                          00003200
   7        TM=1.0 - T                                                          00003300
   7        RR=1.0 - R*R                                                        00003400
   7        SS=1.0 - S*S                                                        00003500
   7        TT=1.0 - T*T                                                        00003600
   1  C                                                                          00003700
   1  CC                                                                         00003800
   1  CC     INTERPOLATION FUNCTIONS AND THEIR DERIVATIVES                       00003900
   1  CC                                                                         00004000
   1  CC                                                                         00004100
   1  CC     8-NODE BRICK                                                        00004200
   1  C                                                                          00004300
   7        H(1)=0.125*RP*SP*TP                                                 00004400
   7        H(2)=0.125*RM*SP*TP                                                 00004500
   7        H(3)=0.125*RM*SM*TP                                                 00004600
   7        H(4)=0.125*RP*SM*TP                                                 00004700
   7        H(5)=0.125*RP*SP*TM                                                 00004800
   7        H(6)=0.125*RM*SP*TM                                                 00004900
   7        H(7)=0.125*RM*SM*TM
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7        H(8)=0.125*RP*SM*TM                                                  00005000
   1  C                                                                          00005100
   7        P(1,1)= 0.125*SP*TP                                                  00005200
   7        P(1,2)=-P(1,1)                                                       00005300
   7        P(1,3)=-0.125*SM*TP                                                  00005400
   7        P(1,4)=-P(1,3)                                                       00005500
   7        P(1,5)= 0.125*SP*TM                                                  00005600
   7        P(1,6)=-P(1,5)                                                       00005700
   7        P(1,7)=-0.125*SM*TM                                                  00005800
   7        P(1,8)=-P(1,7)                                                       00005900
   1  C                                                                          00006000
   7        P(2,1)= 0.125*RP*TP                                                  00006100
   7        P(2,2)= 0.125*RM*TP                                                  00006200
   7        P(2,3)=-P(2,2)                                                       00006300
   7        P(2,4)=-P(2,1)                                                       00006400
   7        P(2,5)= 0.125*RP*TM                                                  00006500
   7        P(2,6)= 0.125*RM*TM                                                  00006600
   7        P(2,7)=-P(2,6)                                                       00006700
   7        P(2,8)=-P(2,5)                                                       00006800
   1  C                                                                          00006900
   7        P(3,1)= 0.125*RP*SP                                                  00007000
   7        P(3,2)= 0.125*RM*SP                                                  00007100
   7        P(3,3)= 0.125*RM*SM                                                  00007200
   7        P(3,4)= 0.125*RP*SM                                                  00007300
   7        P(3,5)=-P(3,1)                                                       00007400
   7        P(3,6)=-P(3,2)                                                       00007500
   7        P(3,7)=-P(3,3)                                                       00007600
   7        P(3,8)=-P(3,4)                                                       00007700
   1  C                                                                          00007800
   7        IF (IEL.EQ.8) GO TO 50                                               00007900
   1  C                                                                          00008000
   1  C                                                                          00008100
   1  C     ADD DEGREES OF FREEDOM IN EXCESS OF 8                                00008200
   1  C                                                                          00008300
   7        I=0                                                                  00008400
   5      2 I=I + 1                                                              00008500
   7        IF (I.GT.NND9) GO TO 40                                              00008600
   7        NN=NOD9(I) - 8                                                       00008700
   7        GO TO (9,10,11,12,13,14,15,16,17,18,19,20,21) ,NN                    00008800
   1  C                                                                          00008900
   5      9 H(9) =0.25*RR*SP*TP                                                  00009000
   7        P(1,9) =-0.50*R*SP*TP                                                00009100
   7        P(2,9) = 0.25*RR*TP                                                  00009200
   7        P(3,9) = 0.25*RR*SP                                                  00009300
   7        GO TO 2                                                              00009400
   4     10 H(10)=0.25*RM*SS*TP                                                  00009500
   7        P(1,10)=-0.25*SS*TP                                                  00009600
   7        P(2,10)=-0.50*RM*S*TP                                                00009700
   7        P(3,10)= 0.25*RM*SS                                                  00009800
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7        GO TO 2                                                      00009900
   4     11 H(11)=0.25*RR*SM*TP                                          00010000
   7        P(1,11)=-0.50*R*SM*TP                                        00010100
   7        P(2,11)=-0.25*RR*TP                                          00010200
   7        P(3,11)= 0.25*RR*SM                                          00010300
   7        GO TO 2                                                      00010400
   4     12 H(12)=0.25*RP*SS*TP                                          00010500
   7        P(1,12)= 0.25*SS*TP                                          00010600
   7        P(2,12)=-0.50*RP*S*TP                                        00010700
   7        P(3,12)= 0.25*RP*SS                                          00010800
   7        GO TO 2                                                      00010900
   4     13 H(13)=0.25*RR*SP*TM                                          00011000
   7        P(1,13)=-0.50*R*SP*TM                                        00011100
   7        P(2,13)= 0.25*RR*TM                                          00011200
   7        P(3,13)=-0.25*RR*SP                                          00011300
   7        GO TO 2                                                      00011400
   4     14 H(14)=0.25*RM*SS*TM                                          00011500
   7        P(1,14)=-0.25*SS*TM                                          00011600
   7        P(2,14)=-0.50*RM*S*TM                                        00011700
   7        P(3,14)=-0.25*RM*SS                                          00011800
   7        GO TO 2                                                      00011900
   4     15 H(15)=0.25*RR*SM*TM                                          00012000
   7        P(1,15)=-0.50*R*SM*TM                                        00012100
   7        P(2,15)=-0.25*RR*TM                                          00012200
   7        P(3,15)=-0.25*RR*SM                                          00012300
   7        GO TO 2                                                      00012400
   4     16 H(16)=0.25*RP*SS*TM                                          00012500
   7        P(1,16)= 0.25*SS*TM                                          00012600
   7        P(2,16)=-0.50*RP*S*TM                                        00012700
   7        P(3,16)=-0.25*RP*SS                                          00012800
   7        GO TO 2                                                      00012900
   4     17 H(17)=0.25*RP*SP*TT                                          00013000
   7        P(1,17)= 0.25*SP*TT                                          00013100
   7        P(2,17)= 0.25*RP*TT                                          00013200
   7        P(3,17)=-0.50*RP*SP*T                                        00013300
   7        GO TO 2                                                      00013400
   4     18 H(18)=0.25*RM*SP*TT                                          00013500
   7        P(1,18)=-0.25*SP*TT                                          00013600
   7        P(2,18)= 0.25*RM*TT                                          00013700
   7        P(3,18)=-0.50*RM*SP*T                                        00013800
   7        GO TO 2                                                      00013900
   4     19 H(19)=0.25*RM*SM*TT                                          00014000
   7        P(1,19)=-0.25*SM*TT                                          00014100
   7        P(2,19)=-0.25*RM*TT                                          00014200
   7        P(3,19)=-0.50*RM*SM*T                                        00014300
   7        GO TO 2                                                      00014400
   4     20 H(20)=0.25*RP*SM*TT                                          00014500
   7        P(1,20)= 0.25*SM*TT                                          00014600
   7        P(2,20)=-0.25*RP*TT                                          00014700
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8
   7         P(3,20)=-0.50*RP*SM*T                                                  00014800
   7         GO TO 2                                                                00014900
   4      21 H(21)=RR*SS*TT                                                         00015000
   7         P(1,21)=-2.0*R*SS*TT                                                   00015100
   7         P(2,21)=-2.0*S*RR*TT                                                   00015200
   7         P(3,21)=-2.0*T*RR*SS                                                   00015300
   7         GO TO 2                                                                00015400
   1  C                                                                             00015500
   1  C      MODIFY FIRST 8 FUNCTIONS IF 9 OR MORE NODES IN ELEMENT                 00015600
   1  C                                                                             00015700
   4      40 IH=0                                                                   00015800
   4      41 IH=IH + 1                                                              00015900
   7         IF (IH.GT.NND9) GO TO 50                                               00016000
   7         II=IH + 7                                                              00016100
   7         IF (II.EQ.IELX) GO TO 51                                               00016200
   4      42 IN=NOD9(IH)                                                            00016300
   7         IF (IN.GT.16) GO TO 46                                                 00016400
   7         I1=IN - 8                                                              00016500
   7         I2=IPERM(I1)                                                           00016600
   7         H(I1)=H(I1) - 0.5*H(IN)                                                00016700
   7         H(I2)=H(I2) - 0.5*H(IN)                                                00016800
   7         H(IH+8)=H(IN)                                                          00016900
   7         DO 45 J=1,3                                                            00017000
   7         P(J,I1)=P(J,I1) - 0.5*P(J,IN)                                          00017100
   7         P(J,I2)=P(J,I2) - 0.5*P(J,IN)                                          00017200
   4      45 P(J,IH+8)=P(J,IN)                                                      00017300
   7         GO TO 41                                                               00017400
   4      46 IF (IN.EQ.21) GO TO 30                                                 00017500
   7         I1=IN - 16                                                             00017600
   7         I2=I1 + 4                                                              00017700
   7         H(I1)=H(I1) - 0.5*H(IN)                                                00017800
   7         H(I2)=H(I2) - 0.5*H(IN)                                                00017900
   7         H(IH+8)=H(IN)                                                          00018000
   7         DO 47 J=1,3                                                            00018100
   7         P(J,I1)=P(J,I1) - 0.5*P(J,IN)                                          00018200
   7         P(J,I2)=P(J,I2) - 0.5*P(J,IN)                                          00018300
   4      47 P(J,IH+8)=P(J,IN)                                                      00018400
   7         GO TO 41                                                               00018500
   1  C                                                                             00018600
   1  C      MODIFY FIRST 20 FUNCTIONS IF NODE 21 IS PRESENT                        00018700
   1  C                                                                             00018800
   4      30 IH=0                                                                   00018900
   4      31 IH=IH + 1                                                              00019000
   7         IN=NOD9(IH)                                                            00019100
   7         IF (IN.EQ.21) GO TO 35                                                 00019200
   7         IF (IN.GT.16) GO TO 33                                                 00019300
   7         I1=IN - 8                                                              00019400
   7         I2=IPERM(I1)                                                           00019500
   7         H(I1)=H(I1) + 0.125*H(21)                                              00019600
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8
   7        H(I2)=H(I2) + 0.125*H(21)                                          00019700
   7        DO 32 J=1,3                                                        00019800
   7        P(J,I1)=P(J,I1) + 0.125*P(J,21)                                    00019900
   4     32 P(J,I2)=P(J,I2) + 0.125*P(J,21)                                    00020000
   7        GO TO 31                                                           00020100
   4     33 I1=IN - 16                                                         00020200
   7        I2=I1 + 4                                                          00020300
   7        H(I1)=H(I1) + 0.125*H(21)                                          00020400
   7        H(I2)=H(I2) + 0.125*H(21)                                          00020500
   7        DO 34 J=1,3                                                        00020600
   7        P(J,I1)=P(J,I1) + 0.125*P(J,21)                                    00020700
   4     34 P(J,I2)=P(J,I2) + 0.125*P(J,21)                                    00020800
   7        GO TO 31                                                           00020900
   4     35 DO 36 I=1,8                                                        00021000
   7        H(I)=H(I) - 0.125*H(21)                                            00021100
   7        DO 36 J=1,3                                                        00021200
   4     36 P(J,I)=P(J,I) - 0.125*P(J,21)                                      00021300
   7        NN=NND9 + 7                                                        00021400
   7        IF (NN.EQ.8) GO TO 50                                              00021500
   7        DO 38 I=9,NN                                                       00021600
   7        H(I)=H(I) - 0.25*H(21)                                             00021700
   7        DO 38 J=1,3                                                        00021800
   4     38 P(J,I)=P(J,I) - 0.25*P(J,21)                                       00021900
   7        H(NND9+8)=H(21)                                                    00022000
   7        DO 39 J=1,3                                                        00022100
   4     39 P(J,NND9+8)=P(J,21)                                                00022200
   1  C                                                                        00022300
   1  C                                                                        00022400
   1  C     EVALUATE JACOBIAN MATRIX AT POINT (R,S,T)                          00022500
   1  C                                                                        00022600
   1  C                                                                        00022700
   4     50 IF (IELX.LT.IELD) RETURN                                           00022800
   4     51 DO 100 I=1,3                                                       00022900
   7        DO 100 J=1,3                                                       00023000
   7        DUM=0.0                                                            00023100
   7        DO 90 K=1,IELX                                                     00023200
   4     90 DUM=DUM + P(I,K)*XX(J,K)                                           00023300
   3    100 XJ(I,J)=DUM                                                        00023400
   1  C                                                                        00023500
   1  C                                                                        00023600
   1  C     COMPUTE DETERMINANT OF JACOBIAN MATRIX AT POINT (R,S,T)            00023700
   1  C                                                                        00023800
   1  C                                                                        00023900
   7        DET = XJ(1,1)*XJ(2,2)*XJ(3,3)                                      00024000
   6       1    + XJ(1,2)*XJ(2,3)*XJ(3,1)                                     00024100
   6       2    + XJ(1,3)*XJ(2,1)*XJ(3,2)                                     00024200
   6       3    - XJ(1,3)*XJ(2,2)*XJ(3,1)                                     00024300
   6       4    - XJ(1,2)*XJ(2,1)*XJ(3,3)                                     00024400
   6       5    - XJ(1,1)*XJ(2,3)*XJ(3,2)                                     00024500
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7        DUM=DABS(DET)                                                  00024600
   7        IF (DUM.GT.1.0E-08) GO TO 110                                  00024700
   7        WRITE(KIO6,2000)                                               00024800
   7        STOP                                                           00024900
   3   110  IF (IELX.LT.IELD) GO TO 42                                     00025000
   1   C                                                                   00025100
   1   C                                                                   00025200
   7        RETURN                                                         00025300
   1   C                                                                   00025400
   1   C                                                                   00025500
   1   C                                                                   00025600
   2    2000 FORMAT (40H STOP  -  NEGATIVE OR ZERO DETERMINANT       )     00025700
   1   C                                                                   00025800
   1   C                                                                   00025900
   7        END                                                            00026000
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8
   7        SUBROUTINE FUNCT2 (R,S,H,P,NOD5,XJ,DET,XX,NEL)                          00000100
   7        IMPLICIT REAL*8(A-H,O-Z)                                                00000200
   7        COMMON /TODIM/ BET,THIC,DE,IEL,NND5                                     00000300
   7        COMMON/BBCCIO/KIO1,KIO2,KIO3,KIO4,KIO5,KIO6,KIO7,KIO8,KIO9,             00000400
   6       1KIO10,KIO11,KIO12,KIO13,KIO14,KIO15,KIO16,KIO17,KIO18,KIO19,KIO20,     00000500
   6       2KIO21,KIO22,KIO23,KIO24,KIO25,KIO26,KIO27,KIO28,KIO29,KIO30            00000600
   7        DIMENSION H(1),P(2,1),NOD5(1),IPERM(4),XJ(2,2),XX(2,1)                  00000700
   7        DATA IPERM/2,3,4,1/                                                     00000800
   1  C                                                                             00000900
   7        RP = 1.0 + R                                                            00001000
   7        SP = 1.0 + S                                                            00001100
   7        RM = 1.0 - R                                                            00001200
   7        SM = 1.0 - S                                                            00001300
   7        R2 = 1.0 - R*R                                                          00001400
   7        S2 = 1.0 - S*S                                                          00001500
   1  C                                                                             00001600
   1  C                                                                             00001700
   1  C     INTERPOLATION FUNCTIONS AND THEIR DERIVATIVES                           00001800
   1  C                                                                             00001900
   1  C     4-NODE ELEMENT                                                          00002000
   1  C                                                                             00002100
   7        H(1) = 0.25* RP* SP                                                     00002200
   7        H(2) = 0.25* RM* SP                                                     00002300
   7        H(3) = 0.25* RM* SM                                                     00002400
   7        H(4) = 0.25* RP* SM                                                     00002500
   7        P(1,1)=0.25*SP                                                          00002600
   7        P(1,2)=-P(1,1)                                                          00002700
   7        P(1,3)=-0.25*SM                                                         00002800
   7        P(1,4)=-P(1,3)                                                          00002900
   7        P(2,1)=0.25*RP                                                          00003000
   7        P(2,2)=0.25*RM                                                          00003100
   7        P(2,3)=-P(2,2)                                                          00003200
   7        P(2,4)=-P(2,1)                                                          00003300
   1  C                                                                             00003400
   7        IF (IEL.EQ.4) GO TO 50                                                  00003500
   1  C                                                                             00003600
   1  C     ADD DEGREES OF FREEDOM IN EXCESS OF 4                                   00003700
   1  C                                                                             00003800
   7          I=0                                                                   00003900
   5        2 I=I + 1                                                             00004000
   7          IF (I.GT.NND5) GO TO 40                                             00004100
   7          NN=NOD5(I) - 4                                                      00004200
   7          GO TO (5,6,7,8), NN                                                 00004300
   1  C                                                                           00004400
   5        5 H(5) = 0.50* R2* SP                                                 00004500
   7          P(1,5)=-R*SP                                                        00004600
   7          P(2,5)=0.50*R2                                                      00004700
   7          GO TO 2                                                             00004800
   5        6 H(6) = 0.50* RM* S2                                                 00004900
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7        P(1,6)=-0.50*S2                                                  00005000
   7        P(2,6)=-RM*S                                                     00005100
   7        GO TO 2                                                          00005200
   5     7  H(7) = 0.50* R2* SM                                              00005300
   7        P(1,7)=-R*SM                                                     00005400
   7        P(2,7)=-0.50*R2                                                  00005500
   7        GO TO 2                                                          00005600
   5     8  H(8) = 0.50* RP* S2                                              00005700
   7        P(1,8)=0.50*S2                                                   00005800
   7        P(2,8)=-RP*S                                                     00005900
   7        GO TO 2                                                          00006000
   1  C                                                                      00006100
   1  C     CORRECT FUNCTIONS AND DERIVATIVES IF 5 OR MORE NODES ARE          00006200
   1  C     USED TO DESCRIBE THE ELEMENT                                     00006300
   1  C                                                                      00006400
   4    40 IH=0                                                            00006500
   4    41 IH=IH + 1                                                       00006600
   7        IF (IH.GT.NND5) GO TO 50                                         00006700
   7        IN=NOD5(IH)                                                      00006800
   7        I1=IN - 4                                                        00006900
   7        I2=IPERM(I1)                                                     00007000
   7        H(I1)=H(I1) - 0.5*H(IN)                                          00007100
   7        H(I2)=H(I2) - 0.5*H(IN)                                          00007200
   7        H(IH + 4)=H(IN)                                                  00007300
   7        DO 45 J=1,2                                                      00007400
   7        P(J,I1)=P(J,I1) - 0.5*P(J,IN)                                    00007500
   7        P(J,I2)=P(J,I2) - 0.5*P(J,IN)                                    00007600
   4    45 P(J,IH + 4)=P(J,IN)                                             00007700
   7        GO TO 41                                                         00007800
   1  C                                                                      00007900
   1  C     EVALUATE THE JACOBIAN MATRIX AT POINT (R,S)                      00008000
   1  C                                                                      00008100
   4    50 DO 100 I=1,2                                                    00008200
   7        DO 100 J=1,2                                                     00008300
   7        DUM = 0.0                                                        00008400
   7        DO 90 K=1,IEL                                                    00008500
   4    90 DUM = DUM + P(I,K)* XX(J,K)                                     00008600
   3   100 XJ(I,J) = DUM                                                   00008700
   1  C                                                                      00008800
   1  C     COMPUTE THE DETERMINANT OF THE JACOBIAN MATRIX AT POINT (R,S)    00008900
   1  C                                                                      00009000
   7        DET = XJ(1,1)* XJ(2,2) - XJ(2,1)* XJ(1,2)                        00009100
   7        DUM=DABS(DET)                                                    00009200
   7        IF(DUM.GT.1.0E-8) GO TO 110                                      00009300
   7        WRITE(KIO6,2000)                                                 00009400
   7        STOP                                                             00009500
   3   110 CONTINUE                                                        00009600
   1  C                                                                      00009700
   7        RETURN                                                           00009800
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   1  C                                                                      00009900
   1  C                                                                      00010000
   2   2000 FORMAT (10H0*** ERROR,                                           00010100
   6       1       40H ZERO JACOBIAN DETERMINANT FOR ELEMENT (,I4,1H)  )     00010200
   1  C                                                                      00010300
   7         END                                                            00010400
```

```
START                                                                                                                MOD
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8                               FLAGS
   7        SUBROUTINE ICDMOD (NEL,NPT,IDW,PROP,WA,YZ,NOD5)                                                    00000100
   7        IMPLICIT REAL*8(A-H,O-Z)                                                                           00000200
   1  C                                                                                                        00000300
   1  C                                                                                                        00000400
   7        COMMON/BBCCIO/KIO1,KIO2,KIO3,KIO4,KIO5,KIO6,KIO7,KIO8,KIO9,                                        00000500
   6       1KIO10,KIO11,KIO12,KIO13,KIO14,KIO15,KIO16,KIO17,KIO18,KIO19,KIO20,                                 00000600
   6       2KIO21,KIO22,KIO23,KIO24,KIO25,KIO26,KIO27,KIO28,KIO29,KIO30                                       00000700
   7        COMMON /EL/ IND,ICOUNT,NPAR(20),NUMEG,NEGL,NEGNL,IMASS,IDAMP,ISTAT                                 00000800
   6       1           ,NDOF,KLIN,IEIG,IMASSN,IDAMPN                                                        00000900
   7        COMMON /TODIM/ BET,THIC,DE,IEL,NND5                                                                00001000
   7        COMMON /CRACK/ GAMMA,STIFAC,SHEFAC                                                                 00001100
   7        DIMENSION PROP(1),WA(IDW,1),YZ(1),NOD5(1)                                                          00001200
   7        DIMENSION XG(4,4),H(8),IPERM(4)                                                                    00001300
   7        DIMENSION P(2,8),XJ(2,2)                                                                           00001400
   7        DIMENSION PGRV(30)                                                                                 00001500
   1  C     REAL KBULK                                                                                         00001600
   1  C                                                                                                        00001700
   7        EQUIVALENCE (NPAR(17),NCON),(NPAR(10),NINT),(NPAR(15),MODEL)                                       00001800
   1  C                                                                                                        00001900
   7        DATA IPERM / 2,3,4,1 /                                                                             00002000
   7        DATA XG /     0.,              0.,              0.,              0., 00002100
   6       1 -.5773502691896, .5773502691896,              0.,              0., 00002200
   6       2 -.7745966692415, .0000000000000, .7745966692415,              0., 00002300
   6       3 -.8611363115941,-.3399810435849, .3399810435849, .8611363115941/ 00002400
   1  C                                                                                                        00002500
   1  C                                                                                                        00002600
   1  C     INITIALIZE WA                                                                                      00002700
   1  C                                                                                                        00002800
   1  C                                                                                                        00002900
   7        DO 11 J=1,NPT                                                                                      00003000
   7        DO 10 I=1,11                                                                                       00003100
   4     10 WA(I,J)=0.                                                                                         00003200
   4     11 WA(12,J)=1000.                                                                                     00003300
   7        IF (MODEL.EQ.4) RETURN                                                                             00003400
   1  C                                                                                                        00003500
   1  C                                                                                                        00003600
   1  C     FIND PRESSURE AT EACH STRAIN INTERPOLATION POINT IN THE                                            00003700
   1  C     MATERIAL CURVE FOR THIS ELEMENT                                                                    00003800
   1  C                                                                                                        00003900
   1  C                                                                                                        00004000
   7        IPOINT=NCON/4                                                                                      00004100
   7        PGRV(1)=0.0                                                                                        00004200
   7        DO 8 K=2,IPOINT                                                                                    00004300
   7        DEV=PROP(K) - PROP(K-1)                                                                            00004400
   7        BULKK=(PROP(IPOINT+K) + PROP(IPOINT+K-1))/2.0                                                      00004500
   5     8  PGRV(K)=PGRV(K-1) + BULKK*DEV                                                                      00004600
   1  C                                                                                                        00004700
   1  C                                                                                                        00004800
   1  C     FIND GROUND PRESSURE AT EACH INTEGRATION POINT OF ELEMENT                                          00004900
```

```
START                                                                                                  MOD
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8  FLAGS
   1  C     ADDED BY S.N.MUKHERJEE. BBC,BADEN. JUNE 1977.                                00005000
   1  C                                                                                  00005100
   7        DO 100 LX=1,NINT                                                           00005200
   7        R=XG(LX,NINT)                                                              00005300
   7        DO 100 LZ=1,NINT                                                           00005400
   7        S=XG(LZ,NINT)                                                              00005500
   7        IPT=(LX-1)*NINT + LZ                                                       00005600
   1  C                                                                                00005700
   7        CALL FUNCT2 (R,S,H,P,NOD5,XJ,DET,YZ,NEL)                                   00005800
   1  C                                                                                00005900
   7        KK=0                                                                       00006000
   7        ZDEPTH=0.                                                                  00006100
   7        DO 30 K=1,IEL                                                              00006200
   7        KK=KK + 2                                                                  00006300
   4     30 ZDEPTH=ZDEPTH + H(K)*YZ(KK)                                                00006400
   7        PGRAV=-GAMMA*ZDEPTH                                                        00006500
   3    100 WA(11,IPT)=PGRAV                                                           00006600
   1  C                                                                                00006700
   1  C                                                                                00006800
   1  C     FIND VOLUMETRIC STRAIN FROM GROUND PRESSURE AT INTEGRATION POINTS          00006900
   1  C                                                                                00007000
   1  C                                                                                00007100
   7        DO 55 IPT=1,NPT                                                            00007200
   7        PGRAV=WA(11,IPT)                                                           00007300
   7        DO 35 L=2,IPOINT                                                           00007400
   7        J=L                                                                        00007500
   7        IF (PGRAV.LT.PGRV(L)) GO TO 40                                             00007600
   4     35 CONTINUE                                                                   00007700
   7        WRITE(KIO6,2002)                                                           00007800
   7        STOP                                                                       00007900
   4     40 CONTINUE                                                                   00008000
   7        I=J - 1                                                                    00008100
   7        DD=PROP(J) - PROP(I)                                                       00008200
   7        C1=PROP(IPOINT+I)                                                          00008300
   7        C2=(PROP(IPOINT+J) - PROP(IPOINT+I))/(2.0*DD)                              00008400
   7        IF (C2.GT.1.E-08) GO TO 41                                                 00008500
   7        DEV=(PGRAV - PGRV(I))/C1                                                   00008600
   7        GO TO 50                                                                   00008700
   4     41 FAC=C1*C1 + 4.0*C2*(PGRAV - PGRV(I))                                       00008800
   7        FAC=DSQRT(FAC)                                                             00008900
   7        DEV1=(-C1 + FAC)/(2.0*C2)                                                  00009000
   7        DEV2=(-C1 - FAC)/(2.0*C2)                                                  00009100
   7        I1=0                                                                       00009200
   7        I2=0                                                                       00009300
   7        IF (DEV1.GE.0.0 .AND. DEV1.LE.DD) I1=1                                     00009400
   7        IF (DEV2.GE.0.0 .AND. DEV2.LE.DD) I2=1                                     00009500
   7        IF (I1.NE.I2) GO TO 45                                                     00009600
   7        WRITE(KIO6,2003)                                                           00009700
   7        STOP                                                                       00009800
```

```
START                                                                                MOD
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8  FLAGS

   4     45 IF (I1.EQ.1) DEV=DEV1                                                    00009900
   7        IF (I2.EQ.1) DEV=DEV2                                                    00010000
   4     50 EVGRAV=PROP(I) + DEV                                                     00010100
   7        WA(10,IPT)=EVGRAV                                                        00010200
   4     55 CONTINUE                                                                 00010300
   1  C                                                                             00010400
   1  C                                                                             00010500
   7        RETURN                                                                   00010600
   1  C                                                                             00010700
   1  C                                                                             00010800
   2   2002 FORMAT (55H **STOP - GRAVITATIONAL STRAIN TOO LARGE FOR MATERIAL C      00010900
   6       1         ,12HURVE INPUT    )                                           00011000
   2   2003 FORMAT (52H **STOP - ERROR IN PRESSURE-VOLUMETRIC STRAIN CALC.  )      00011100
   1  C                                                                             00011200
   7        END                                                                     00011300
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7        SUBROUTINE ICMOD3 (WA,NPT)                                      00000100
   7        IMPLICIT REAL*8(A-H,O-Z)                                        00000200
   1  C                                                                     00000300
   1  C     INITIALIZE WORKING STORAGE WA FOR EACH INTEGRATION POINT         00000400
   1  C        ( I.E., STRESSES AND STRAINS SET TO ZERO )                   00000500
   1  C                                                                     00000600
   7        DIMENSION WA(13,1)                                              00000700
   1  C                                                                     00000800
   7        DO 10 I=1,NPT                                                   00000900
   7        DO 10 J=1,13                                                    00001000
   4     10 WA(J,I)=0.0                                                     00001100
   1  C                                                                     00001200
   7        RETURN                                                          00001300
   7        END                                                             00001400
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7        SUBROUTINE IDRUCK (WA,IWA,PROP,NPT,IDW)                          00000100
   7        IMPLICIT REAL*8(A-H,O-Z)                                         00000200
   1  C                                                                      00000300
   7        DIMENSION WA(IDW,1),IWA(IDW,1),PROP(1)                           00000400
   1  C                                                                      00000500
   1  C                                                                      00000600
   7        DO 25 J=1,NPT                                                    00000700
   1  C                                                                      00000800
   1  C     SET INITIAL STRESSES AND STRAINS EQUAL TO ZERO                   00000900
   1  C                                                                      00001000
   7        DO 15 I=1,8                                                      00001100
   7        WA(I,J)=0.0                                                      00001200
   4     15 CONTINUE                                                         00001300
   1  C                                                                      00001400
   1  C     ANGLE=PROP(3)   AND   COHES=PROP(4)                              00001500
   1  C                                                                      00001600
   7        WA( 9,J)=PROP(3)/57.296                                          00001700
   7        WA(10,J)=PROP(4)                                                 00001800
   1  C                                                                      00001900
   1  C     SET INITIAL STRESS STATE TO  -ELASTIC-                           00002000
   1  C                                                                      00002100
   7        IWA(11,J)=1                                                      00002200
   1  C                                                                      00002300
   4     25 CONTINUE                                                         00002400
   1  C                                                                      00002500
   1  C                                                                      00002600
   7        RETURN                                                           00002700
   1  C                                                                      00002800
   7        END                                                              00002900
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7        SUBROUTINE IELPAL(WA,IWA,PROP,NPT,J)                          00000100
   7        IMPLICIT REAL*8(A-H,O-Z)                                      00000200
   1  C                                                                   00000300
   7        DIMENSION WA(10,1),PROP(1)                                    00000400
   1  C                                                                   00000500
   1  C     SET INITIAL STRESSES AND STRAINS TO ZERO                      00000600
   1  C     SET INITIAL YIELD POINT TO PROP(3)                            00000700
   1  C     SET INITIAL STRESS STATE TO *ELASTIC*                         00000800
   1  C                                                                   00000900
   7        DO 15 I=1,8                                                   00001000
   4     15 WA(I,J)=0.0                                                   00001100
   7        WA(9,J)=PROP(3)                                               00001200
   7        IWA=1                                                         00001300
   1  C                                                                   00001400
   7        RETURN                                                        00001500
   7        END                                                           00001600
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7        SUBROUTINE INITAL  (DISP,VEL,ACC,ID,NDOF)                              00000100
   7        IMPLICIT REAL*8(A-H,O-Z)                                                00000200
   1  C                                                                             00000300
   1  C . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 00000400
   1  C .                                                                         . 00000500
   1  C .   P R O G R A M                                                       . 00000600
   1  C .      TO INITIALIZE DISPLACEMENTS, VELOCITIES AND ACCELERATIONS        . 00000700
   1  C .                                                                         . 00000800
   1  C .        DISP = DISPLACEMENTS                                             . 00000900
   1  C .        VEL  = VELOCITIES                                                . 00001000
   1  C .        ACC  = ACCELERATIONS                                             . 00001100
   1  C .                                                                         . 00001200
   1  C . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 00001300
   7        COMMON/BBCCIO/KIO1,KIO2,KIO3,KIO4,KIO5,KIO6,KIO7,KIO8,KIO9,          00001400
   6       1KIO10,KIO11,KIO12,KIO13,KIO14,KIO15,KIO16,KIO17,KIO18,KIO19,KIO20,00001500
   6       2KIO21,KIO22,KIO23,KIO24,KIO25,KIO26,KIO27,KIO28,KIO29,KIO30          00001600
   1  C                                                                             00001700
   7        COMMON /SOL/ NUMNP,NEQ,NWK,NWM,NWC,NUMEST,MIDEST,MAXEST,NSTE,MA       00001800
   7        COMMON /EL/ IND,ICOUNT,NPAR(20),NUMEG,NEGL,NEGNL,IMASS,IDAMP,ISTAT00001900
   6       1             ,NDOFDM,KLIN,IEIG,IMASSN,IDAMPN                         00002000
   7        COMMON /VAR/ NG,KPRI,MODEX,KSTEP,ITE,ITEMAX,IREF,IEQREF,INOCMD       00002100
   7        DIMENSION DISP(1),VEL(1),ACC(1),ID(NDOF,1)                           00002200
   1  C                                                                             00002300
   7        READ(KIO5,1000) ICON                                                 00002400
   7        WRITE(KIO6,2000) ICON                                                00002500
   7        IF (MODEX.EQ.2 .AND. ICON.NE.0) GO TO 140                            00002600
   7        IF (MODEX.EQ.2) RETURN                                               00002700
   1  C                                                                             00002800
   7        REWIND KIO18                                                         00002900
   7        WRITE(KIO18)                                                         00003000
   6       *          ((ID(I,J),I=1,NDOF),J=1,NUMNP)                            00003100
   1  C                                                                             00003200
   7        IF (ICON.NE.0) GO TO 140                                             00003300
   1  C                                                                             00003400
   7        DO 100 I=1,NEQ                                                       00003500
   3    100 DISP(I)=0.                                                           00003600
   7        IF (ISTAT.EQ.0) GO TO 200                                            00003700
   7        DO 110 I=1,NEQ                                                       00003800
   7        VEL(I)=0.                                                            00003900
   3    110 ACC(I)=0.                                                            00004000
   7        GO TO 200                                                            00004100
   1  C                                                                             00004200
   3    140 READ(KIO5,1010) (DISP(I),I=1,NEQ)                                    00004300
   7        WRITE(KIO6,2010)                                                     00004400
   7        WRITE(KIO6,2020) (DISP(I),I=1,NEQ)                                   00004500
   7        IF (ISTAT.EQ.0) GO TO 200                                            00004600
   7        READ(KIO5,1010) (VEL(I),I=1,NEQ)                                     00004700
   7        WRITE(KIO6,2020) (VEL(I),I=1,NEQ)                                    00004800
   7        READ(KIO5,1010) (ACC(I),I=1,NEQ)                                     00004900
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7        WRITE(KIO6,2020) (ACC(I),I=1,NEQ)                                  00005000
   1  C                                                                        00005100
   3    200 IF (MODEX.EQ.2) RETURN                                             00005200
   7        WRITE(KIO18) (DISP(I),I=1,NEQ)                                     00005300
   7        IF (ISTAT.EQ.0) RETURN                                             00005400
   1  C                                                                        00005500
   7        WRITE(KIO18) (VEL(I),I=1,NEQ)                                      00005600
   7        WRITE(KIO18) (ACC(I),I=1,NEQ)                                      00005700
   7        RETURN                                                             00005800
   1  C                                                                        00005900
   2   1000 FORMAT (I5)                                                        00006000
   2   1010 FORMAT (6E12.6)                                                    00006100
   2   2000 FORMAT (1H1,36H I N I T I A L  C O N D I T I O N S     ///         00006200
   6       1  5X,51HINITIAL CONDITIONS CODE                    (ICON) =,I5/4X 00006300
   6       2            42H    EQ.0, ZERO INITIAL CONDITIONS              /4X,00006400
   6       3          42H    EQ.1, INITIAL CONDITIONS ARE READ            /10X,00006500
   6       4            34H    (BUT RESTART OVER-RIDES ICON)    /)            00006600
   2   2010 FORMAT (//34H INITIAL CONDITIONS DATA AS INPUT  //                 00006700
   6      1        28H         ( CARD BY CARD  )         //)                 00006800
   2   2020 FORMAT (6E12.6)                                                    00006900
   1  C                                                                        00007000
   7        END                                                                00007100
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8
   7        SUBROUTINE INITWA (MODEL)                                          00000100
   7        IMPLICIT REAL*8(A-H,O-Z)                                           00000200
   1  C                                                                        00000300
   1  C     INITIALIZES THE WORKING VECTOR WA                                  00000400
   1  C     FOR TWO-DIMENSIONAL MATERIAL MODELS                                00000500
   1  C                                                                        00000600
   1  C                                                                        00000700
   7        GO TO ( 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12 ) , MODEL           00000800
   1  C                                                                        00000900
   1  C                                                                        00001000
   1  C.... MODEL = 1    L I N E A R   I S O T R O P I C                       00001100
   1  C                                                                        00001200
   5     1 RETURN                                                            00001300
   1  C                                                                        00001400
   1  C                                                                        00001500
   1  C.... MODEL = 2    L I N E A R   O R T H O T R O P I C                   00001600
   1  C                                                                        00001700
   5     2 RETURN                                                            00001800
   1  C                                                                        00001900
   1  C                                                                        00002000
   1  C.... MODEL = 3    V A R I A B L E   T A N G E N T   M O D U L I         00002100
   1  C                                                                        00002200
   1  3     CALL ELT2D3                                                       00002300
   7        RETURN                                                            00002400
   1  C                                                                        00002500
   1  C                                                                        00002600
   1  C.... MODEL = 4    C U R V E   D E S C R I P T I O N   M O D E L         00002700
   1  C                                                                        00002800
   1  4     CALL ELT2D4                                                       00002900
   7        RETURN                                                            00003000
   1  C                                                                        00003100
   1  C                                                                        00003200
   1  C.... MODEL = 5    C D M O D   W I T H   C R A C K I N G                 00003300
   1  C                                                                        00003400
   1  5     CALL ELT2D4                                                       00003500
   7        RETURN                                                            00003600
   1  C                                                                        00003700
   1  C                                                                        00003800
   1  C.... MODEL = 6    E L A S T O P L A S T I C (VON MISES YIELD CRITERION)00003900
   1  C                                                                        00004000
   1  6     CALL ELT2D6                                                       00004100
   7        RETURN                                                            00004200
   1  C                                                                        00004300
   1  C                                                                        00004400
   1  C.... MODEL = 7    E L A S T O P L A S T I C (DRUCKER-PRAGER)             00004500
   1  C                                                                        00004600
   1  7     CALL ELT2D7                                                       00004700
   7        RETURN                                                            00004800
   1  C                                                                        00004900
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8
   1  C                                                                      00005000
   1  C.... MODEL = 8    I N C O M P R E S S I B L E   E L A S T I C         00005100
   1  C                                                                      00005200
   5      8 RETURN                                                           00005300
   1  C                                                                      00005400
   1  C                                                                      00005500
   1  C.... MODEL.GT.8    U S E R   S U P P L I E D   M O D E L S            00005600
   1  C                                                                      00005700
   1  C                                                                      00005800
   1  9      CALL ELT2D9                                                     00005900
   7         RETURN                                                         00006000
   1  10     CALL EL2D10                                                     00006100
   7         RETURN                                                         00006200
   1  11     CALL EL2D11                                                     00006300
   7         RETURN                                                         00006400
   1  12     CALL EL2D12                                                     00006500
   7         RETURN                                                         00006600
   1  C                                                                      00006700
   1  C                                                                      00006800
   1  C                                                                      00006900
   7         END                                                            00007000
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8
   7        SUBROUTINE INPUT (ID,X,Y,Z,NUMNP,NDOF,NEQ)                          00000100
   1  C                                                                         00000200
   1  C . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 00000300
   1  C .                                                                     . 00000400
   1  C .   P R O G R A M                                                   . 00000500
   1  C .     1. TO READ AND PRINT NODAL POINT INPUT DATA                   . 00000600
   1  C .     2. TO GENERATE AND PRINT ALL NODAL DATA                       . 00000700
   1  C .     3. TO CALCULATE EQUATION NUMBERS AND STORE THEM IN ID ARRRAY  . 00000800
   1  C .                                                                     . 00000900
   1  C .           N=ELEMENT NUMBER                                        . 00001000
   1  C .           ID=BOUNDARY CONDITION CODES (0=FREE,1=FIXED)            . 00001100
   1  C .           X,Y,Z= COORDINATES                                      . 00001200
   1  C .           KN= GENERATION CODE, I.E. INCREMENT ON NODAL POINT NO   . 00001300
   1  C .           IT= INPUT COORDINATE TYPE                               . 00001400
   1  C .               EQ.0  RECTANGULAR COORDINATES                       . 00001500
   1  C .               EQ.X  CYLINDRICAL COORDINATES (X,R, THETA IN DEG)   . 00001600
   1  C .                                                                     . 00001700
   1  C . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 00001800
   1  C                                                                         00001900
   1  C                                                                         00002000
   7        IMPLICIT REAL*8(A-H,O-Z)                                          00002100
   7        COMMON/BBCCIO/KIO1,KIO2,KIO3,KIO4,KIO5,KIO6,KIO7,KIO8,KIO9,       00002200
   6       1KIO10,KIO11,KIO12,KIO13,KIO14,KIO15,KIO16,KIO17,KIO18,KIO19,KIO20,00002300
   6       2KIO21,KIO22,KIO23,KIO24,KIO25,KIO26,KIO27,KIO28,KIO29,KIO30       00002400
   7        DIMENSION X(1),Y(1),Z(1),ID(NDOF,1)                               00002500
   7        DIMENSION IDT(6),IDTOLD(6)                                        00002600
   7        DIMENSION IPRC(4)                                                 00002700
   1  C                                                                         00002800
   7        DATA IPRC /1H ,1HA,1HB,1HC /                                      00002900
   1  C                                                                         00003000
   1  C                                                                         00003100
   1  C     READ AND PRINT NODAL POINT DATA                                     00003200
   1  C                                                                         00003300
   7        WRITE(KIO6,2000)                                                  00003400
   7        WRITE(KIO6,2001)                                                  00003500
   7        NOLD=0                                                            00003600
   7        ITOLD=0                                                           00003700
   7        DUM=0.                                                            00003800
   7        IPR=IPRC(1)                                                       00003900
   7        RAD=DATAN(1.0D0)/45.0D0                                           00004000
   1  C                                                                         00004100
   4     10 READ(5,1000) IT,N,JPR,(IDT(I),I=1,6),X(N),Y(N),Z(N),KN            00004200
   7        WRITE(KIO6,2002)N,(IDT(I),I=1,6),X(N),Y(N),Z(N),KN,IT             00004300
   7        IF (N.EQ.1) IPR=JPR                                               00004400
   7        IF (IT.EQ.IPRC(1)) GO TO 12                                       00004500
   7        DUM=Z(N)*RAD                                                      00004600
   7        Z(N)=Y(N)*DSIN(DUM)                                               00004700
   7        Y(N)=Y(N)*DCOS(DUM)                                               00004800
   4     12 DO 15 I=1,NDOF                                                    00004900
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7         II=I                                                                    00005000
   7         IF (NDOF.EQ.2) II=I + 1                                                 00005100
   4     15 ID(I,N)=IDT(II)                                                          00005200
   7         IF (NOLD.EQ.0) GO TO 50                                                 00005300
   7         DO 20 I=1,NDOF                                                          00005400
   7         II=I                                                                    00005500
   7         IF (NDOF.EQ.2) II=I + 1                                                 00005600
   7         IF (IDT(II).EQ.0 .AND. IDTOLD(II).LT.0) ID(I,N)=ID(I,NOLD)              00005700
   7         IF (IDT(II).EQ.0 .AND. IDTOLD(II).LT.0) IDT(II)=IDTOLD(II)              00005800
   4     20 CONTINUE                                                                 00005900
   7         IF (KNOLD.EQ.0) GO TO 50                                                00006000
   7         NUM=(N-NOLD) / KNOLD                                                    00006100
   7         NUMN=NUM-1                                                              00006200
   7         IF(NUMN.LT.1) GO TO 50                                                  00006300
   7         XNUM=NUM                                                                00006400
   7         DX=(X(N)-X(NOLD))/XNUM                                                  00006500
   7         IF (IT.EQ.IPRC(1)) GO TO 21                                             00006600
   7         ROLD=Y(NOLD)/DCOS(DUMOLD)                                               00006700
   7         RNEW=Y(N)/DCOS(DUM)                                                     00006800
   7         DR=(RNEW-ROLD)/XNUM                                                     00006900
   7         DT=(DUM-DUMOLD)/XNUM                                                    00007000
   7         GO TO 22                                                                00007100
   4     21 DY=(Y(N)-Y(NOLD))/XNUM                                                   00007200
   7         DZ=(Z(N)-Z(NOLD))/XNUM                                                  00007300
   4     22 K=NOLD                                                                   00007400
   7         DO 30 J=1,NUMN                                                          00007500
   7         KK=K                                                                    00007600
   7         K=K + KNOLD                                                             00007700
   7         X(K)=X(KK)+DX                                                           00007800
   7         IF (IT.EQ.IPRC(1)) GO TO 26                                             00007900
   7         ROLD=ROLD+DR                                                            00008000
   7         DUMOLD=DUMOLD+DT                                                        00008100
   7         Y(K)=ROLD*DCOS(DUMOLD)                                                  00008200
   7         Z(K)=ROLD*DSIN(DUMOLD)                                                  00008300
   7         GO TO 28                                                                00008400
   4     26 Y(K)=Y(KK)+DY                                                            00008500
   7         Z(K)=Z(KK)+DZ                                                           00008600
   4     28 DO 30 I=1,NDOF                                                           00008700
   7         ID(I,K)=ID(I,KK)                                                        00008800
   4     30 CONTINUE                                                                 00008900
   1  C                                                                              00009000
   4     50 NOLD=N                                                                   00009100
   7         KNOLD=KN                                                                00009200
   7         DUMOLD=DUM                                                              00009300
   7         DO 60 I=1,6                                                             00009400
   4     60 IDTOLD(I)=IDT(I)                                                         00009500
   7         IF(N.NE.NUMNP) GO TO 10                                                 00009600
   1  C                                                                              00009700
   7         IF (IPR.EQ.IPRC(2) .OR. IPR.EQ.IPRC(4)) GO TO 52                        00009800
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7        WRITE(KIO6,2003)                                                      00009900
   7        WRITE(KIO6,2008)                                                      00010000
   7        ID0=1                                                                 00010100
   7        IF(NDOF.EQ.1) WRITE(KIO6,2005) (N,ID(1,N),ID0,ID0,ID0,ID0,ID0,       00010200
   6       1                               X(N),Y(N),Z(N),N=1,NUMNP)             00010300
   7        IF(NDOF.EQ.2)WRITE(KIO6,2005)(N,ID0,ID(1,N),ID(2,N),ID0,ID0,ID0,     00010400
   6       1                               X(N),Y(N),Z(N),N=1,NUMNP)             00010500
   7        IF (NDOF.EQ.3) WRITE(KIO6,2005)(N,(ID(I,N),I=1,3),ID0,ID0,ID0,       00010600
   6       1                               X(N),Y(N),Z(N),N=1,NUMNP)             00010700
   7        IF(NDOF.EQ.6) WRITE(KIO6,2005) (N,(ID(I,N),I=1,6),                   00010800
   6       1                               X(N),Y(N),Z(N),N=1,NUMNP)             00010900
   4     52 CONTINUE                                                             00011000
   1  C                                                                          00011100
   1  C      NUMBER UNKNOWNS                                                     00011200
   1  C                                                                          00011300
   7        NEQ=0                                                               00011400
   7        DO 100 N=1,NUMNP                                                    00011500
   7        DO 100 I=1,NDOF                                                     00011600
   7        IF (ID(I,N)) 110,120,110                                            00011700
   3    120 NEQ=NEQ + 1                                                         00011800
   7        ID(I,N)=NEQ                                                         00011900
   7        GO TO 100                                                           00012000
   3    110 ID(I,N)=0                                                           00012100
   3    100 CONTINUE                                                            00012200
   1  C                                                                          00012300
   7        IF (IPR.EQ.IPRC(3) .OR. IPR.EQ.IPRC(4)) RETURN                      00012400
   7        ID0=0                                                               00012500
   7        IF (NDOF.EQ.1) WRITE(KIO6,2004) (N,ID(1,N),ID0,ID0,                 00012600
   6       1                               ID0,ID0,ID0,N=1,NUMNP)               00012700
   7        IF (NDOF.EQ.2) WRITE(KIO6,2004) (N,ID0,ID(1,N),ID(2,N),             00012800
   6       1                               ID0,ID0,ID0,N=1,NUMNP)               00012900
   7        IF (NDOF.EQ.3) WRITE(KIO6,2004) (N,(ID(I,N),I=1,3),                 00013000
   6       1                               ID0,ID0,ID0,N=1,NUMNP)               00013100
   7        IF (NDOF.EQ.6) WRITE(KIO6,2004) (N,(ID(I,N),I=1,6),N=1,NUMNP)       00013200
   1  C                                                                          00013300
   7        RETURN                                                              00013400
   1  C                                                                          00013500
   2   1000 FORMAT (A1,I4,A1,I4,5I5,3F10.0,I5)                                  00013600
   2   2000 FORMAT(1H1,33H N O D A L   P O I N T   D A T A   //)                 00013700
   2   2001 FORMAT (18H INPUT NODAL DATA //5H NODE 10X 24HBOUNDARY CONDITION C00013800
   6       1ODES 14X 23HNODAL POINT COORDINATES 12X 21HMESH GENERATING CODES//00013900
   6       214X 1HX 4X 1HY 4X 1HZ 3X 2HXX 3X 2HYY 3X 2HZZ 12X 1HX 12X 1HY 12X 00014000
   6       31HZ 13X 2HKN 5X 2HIT/)                                             00014100
   2   2002 FORMAT (I5,5X,6I5,3F13.3,10X,I5,6X,A1)                              00014200
   2   2003 FORMAT (//21H GENERATED NODAL DATA)                                 00014300
   2   2004 FORMAT (//17H EQUATION NUMBERS//                                    00014400
   6       1 35H    N    X    Y    Z   XX   YY   ZZ //(7I5))                   00014500
   2   2008 FORMAT(/5H NODE 10X 24HBOUNDARY CONDITION CODES 14X 23HNODAL POINT00014600
   6       1 COORDINATES //14X 1HX 4X 1HY 4X 1HZ 3X 2HXX 3X 2HYY 3X 2HZZ 12X  00014700
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   6       21HX 12X 1HY 12X 1HZ /)                                          00014800
   2   2005 FORMAT (I5,5X,6I5,3F13.3)                                       00014900
   1  C                                                                     00015000
   7        END                                                             00015100
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7        SUBROUTINE INTWA3 (MODEL)                                        00000100
   7        IMPLICIT REAL*8(A-H,O-Z)                                         00000200
   1  C                                                                      00000300
   1  C     INITIALIZES WORKING VECTOR WA FOR THIS ELEMENT                   00000400
   1  C                                                                      00000500
   7        GO TO (1,2,3,4,5,6) , MODEL                                      00000600
   1  C                                                                      00000700
   1  C                                                                      00000800
   1  C     M O D E L   1     LINEAR ISOTROPIC                               00000900
   1  C                                                                      00001000
   5      1 RETURN                                                           00001100
   1  C                                                                      00001200
   1  C     M O D E L   2     LINEAR ORTHOTROPIC                             00001300
   1  C                                                                      00001400
   5      2 RETURN                                                           00001500
   1  C                                                                      00001600
   1  C     M O D E L   3     CURVE DESCRIPTION NONLINEAR MODEL              00001700
   1  C                                                                      00001800
   1  3     CALL ELT3D3                                                      00001900
   7        RETURN                                                           00002000
   1  C                                                                      00002100
   1  C     M O D E L   . G E .   4     USER SUPPLIED MATERIAL MODEL         00002200
   1  C                                                                      00002300
   1  4     CALL ELT3D4                                                      00002400
   7        RETURN                                                           00002500
   1  C                                                                      00002600
   1  5     CALL ELT3D5                                                      00002700
   7        RETURN                                                           00002800
   1  C                                                                      00002900
   1  6     CALL ELT3D6                                                      00003000
   7        RETURN                                                           00003100
   1  C                                                                      00003200
   1  C                                                                      00003300
   7        END                                                              00003400
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7          SUBROUTINE IVTMOD (WA,IWA,IDW,NPT)                                  00000100
   7          IMPLICIT REAL*8(A-H,O-Z)                                            00000200
   1  C                                                                           00000300
   7          DIMENSION WA(IDW,1),IWA(IDW,1)                                      00000400
   1  C                                                                           00000500
   1  C       SET STRESSES AND STRAINS TO ZERO AND SET LOADING CONDITIONS         00000600
   1  C       TO *LOADING*                                                        00000700
   1  C                                                                           00000800
   7          DO 10 I=1,NPT                                                       00000900
   7          DO 15 J=1,9                                                         00001000
   4       15 WA(J,I)=0.0                                                         00001100
   4       10 IWA(10,I)=4                                                         00001200
   1  C                                                                           00001300
   7          RETURN                                                              00001400
   1  C                                                                           00001500
   7          END                                                                 00001600
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7        SUBROUTINE LOADEF (MAXA,DISP,VEL,ACC,R,WV,A,XM)                    00000100
   7        IMPLICIT REAL*8(A-H,O-Z)                                           00000200
   1  C                                                                        00000300
   1  C . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 00000400
   1  C .                                                                    . 00000500
   1  C .   P R O G R A M                                                    . 00000600
   1  C .      . TO CALCULATE EFFECTIVE LOADS (EXCLUDING NONLINEAR           . 00000700
   1  C .        CONTRIBUTIONS)                                              . 00000800
   1  C .                                                                    . 00000900
   1  C . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 00001000
   1  C                                                                        00001100
   7        COMMON/BBCCIO/KIO1,KIO2,KIO3,KIO4,KIO5,KIO6,KIO7,KIO8,KIO9,       00001200
   6       1KIO10,KIO11,KIO12,KIO13,KIO14,KIO15,KIO16,KIO17,KIO18,KIO19,KIO20,00001300
   6       2KIO21,KIO22,KIO23,KIO24,KIO25,KIO26,KIO27,KIO28,KIO29,KIO30       00001400
   7        COMMON /SOL/ NUMNP,NEQ,NWK,NWM,NWC,NUMEST,MIDEST,MAXEST,NSTE,MA   00001500
   7        COMMON /CONST/ DT,DTA,A0,A1,A2,A3,A4,A5,A6,A7,A8,A9,A10,A11       00001600
   6       1              ,A12,A13,A14,A15,A16,A17,A18,A19,A20,IOPE           00001700
   7        COMMON /EL/ IND,ICOUNT,NPAR(20),NUMEG,NEGL,NEGNL,IMASS,IDAMP,ISTAT00001800
   6       1           ,NDOF,KLIN,IEIG,IMASSN,IDAMPN                         00001900
   7        COMMON /VAR/ NG,KPRI,MODEX,KSTEP,ITE,ITEMAX,IREF,IEQREF,INOCMD   00002000
   7        COMMON /DAMPNG/ ADAMP,BDAMP                                       00002100
   7        DIMENSION MAXA(1),DISP(1),VEL(1),ACC(1),R(1),WV(1),A(1),XM(1)     00002200
   1  C                                                                        00002300
   1  C                                                                        00002400
   7        DO 50 I=1,NEQ                                                     00002500
   4     50 R(I)=0.                                                           00002600
   1  C                                                                        00002700
   1  C                                                                        00002800
   1  C     S T I F F N E S S   E F F E C T                                    00002900
   1  C                                                                        00003000
   1  C              ( OR BDAMP*STIFFNESS PORTION OF RAYLEIGH DAMPING )        00003100
   1  C                                                                        00003200
   1  C                                                                        00003300
   7        IF (KLIN.EQ.0) GO TO 40                                           00003400
   1  C                                                                        00003500
   1  C     NONLINEAR ANALYSIS                                                 00003600
   1  C                                                                        00003700
   7        IF (NEGL.EQ.0) GO TO 60                                           00003800
   7        READ(KIO14) (A(I),I=1,NWK)                                        00003900
   7        DO 52 I=1,NEQ                                                     00004000
   4     52 WV(I)=DISP(I)                                                     00004100
   7        IF (BDAMP.EQ.0.0) GO TO 57                                        00004200
   1  C                                                                        00004300
   1  C     ALLOW FOR DAMPING                                                  00004400
   1  C                                                                        00004500
   7        DO 55 I=1,NEQ                                                     00004600
   4     55 WV(I)=WV(I) + BDAMP*(-A4*VEL(I) - A5*ACC(I))                      00004700
   4     57 CALL MULT (R,A,WV,MAXA,NEQ)                                       00004800
   7        GO TO 60                                                          00004900
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   1  C                                                                    00005000
   1  C                                                                    00005100
   1  C     LINEAR ANALYSIS                                                00005200
   1  C                                                                    00005300
   4     40 IF (BDAMP.EQ.0.0) GO TO 60                                     00005400
   7        READ(KIO14)(A(I),I=1,NWK)                                      00005500
   7        DO 45 I=1,NEQ                                                  00005600
   4     45 WV(I)=BDAMP*(-A1*DISP(I) - A4*VEL(I) - A5*ACC(I))              00005700
   7        CALL MULT (R,A,WV,MAXA,NEQ)                                    00005800
   1  C                                                                    00005900
   4     60 IF (ISTAT.EQ.0) GO TO 140                                      00006000
   1  C                                                                    00006100
   1  C                                                                    00006200
   1  C     M A S S   E F F E C T                                          00006300
   1  C                                                                    00006400
   1  C              ( OR ADAMP*MASS PORTION OF RAYLEIGH DAMPING )         00006500
   1  C                                                                    00006600
   1  C                                                                    00006700
   7        IF (KLIN.GT.0) GO TO 110                                       00006800
   1  C                                                                    00006900
   1  C     LINEAR ANALYSIS                                                00007000
   1  C                                                                    00007100
   7        FAC1=-A0 - ADAMP*A1                                            00007200
   7        FAC2=-A2 - ADAMP*A4                                            00007300
   7        FAC3=-A3 - ADAMP*A5                                            00007400
   7        DO 100 I=1,NEQ                                                 00007500
   3   100  WV(I)=FAC1*DISP(I) + FAC2*VEL(I) + FAC3*ACC(I)                00007600
   7        IF (IMASS.EQ.2) GO TO 128                                      00007700
   7        GO TO 115                                                      00007800
   1  C                                                                    00007900
   1  C     NONLINEAR ANALYSIS                                             00008000
   1  C                                                                    00008100
   3   110  FAC2=-A2 - ADAMP*A4                                           00008200
   7        FAC3=-A3 - ADAMP*A5                                            00008300
   7        DO 120 I=1,NEQ                                                 00008400
   3   120  WV(I)=FAC2*VEL(I) + FAC3*ACC(I)                               00008500
   1  C                                                                    00008600
   7        IF (IMASS.EQ.2) GO TO 128                                      00008700
   3   115  DO 170 I=1,NEQ                                                00008800
   3   170  R(I)=R(I) - XM(I)*WV(I)                                        00008900
   7        GO TO 150                                                      00009000
   3   128  READ(KIO14) (A(I),I=1,NWM)                                    00009100
   3   130  CALL MULT (R,A,WV,MAXA,NEQ)                                   00009200
   1  C                                                                    00009300
   1  C                                                                    00009400
   1  C     N O D A L   D A M P I N G   E F F E C T                        00009500
   1  C                                                                    00009600
   1  C                                                                    00009700
   3   150  IF (IDAMPN.EQ.0) GO TO 140                                     00009800
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7          IF (KLIN.GT.0) GO TO 125                                    00009900
   1  C                                                                   00010000
   1  C       LINEAR ANALYSIS                                             00010100
   1  C                                                                   00010200
   7          DO 132 I=1,NEQ                                              00010300
   3    132   WV(I)=-A1*DISP(I) - A4*VEL(I) - A5*ACC(I)                   00010400
   7          GO TO 135                                                   00010500
   1  C                                                                   00010600
   1  C       NONLINEAR ANALYSIS                                          00010700
   1  C                                                                   00010800
   3    125   DO 138 I=1,NEQ                                             00010900
   3    138   WV(I)=           -A4*VEL(I) - A5*ACC(I)                     00011000
   3    135   READ(KIO14) (A(I),I=1,NEQ)                                 00011100
   7          DO 139 I=1,NEQ                                             00011200
   3    139   R(I)=R(I) - A(I)*WV(I)                                      00011300
   1  C                                                                   00011400
   1  C                                                                   00011500
   1  C       E X T E R N A L L Y   A P P L I E D   L O A D S             00011600
   1  C                                                                   00011700
   1  C                                                                   00011800
   3    140   IF (KLIN.NE.0 .OR. ISTAT.NE.0) GO TO 145                    00011900
   7          READ(KIO13) (R(I),I=1,NEQ)                                 00012000
   7          RETURN                                                     00012100
   3    145   READ(KIO13) (WV(I),I=1,NEQ)                                00012200
   7          DO 160 I=1,NEQ                                             00012300
   3    160   R(I)=R(I) + WV(I)                                          00012400
   1  C                                                                   00012500
   7          RETURN                                                     00012600
   7          END                                                        00012700
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7        SUBROUTINE LOADS (R,RV,TIMV,RG,NOD,IDIRN,NCUR,FAC,ID,NLCUR,NDOF)  00000100
   7        IMPLICIT REAL*8(A-H,O-Z)                                          00000200
   1  C                                                                       00000300
   1  C                                                                       00000400
   1  C . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 00000500
   1  C .                                                                   . 00000600
   1  C .   P R O G R A M                                                   . 00000700
   1  C .     1. TO READ THE LOAD CURVES AND CALCULATE FUNCTIONAL VALUES    . 00000800
   1  C .        AT REQUESTED POINTS IN TIME                                . 00000900
   1  C .     2. TO READ NODAL LOAD DATA                                    . 00001000
   1  C .     3. TO CALCULATE THE LOAD VECTOR R FOR EACH TIME STEP AND      . 00001100
   1  C .        WRITE ONTO TAPE 13                                         00001200
   1  C .                                                                   . 00001300
   1  C . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 00001400
   1  C                                                                       00001500
   7        COMMON/BBCCIO/KIO1,KIO2,KIO3,KIO4,KIO5,KIO6,KIO7,KIO8,KIO9,      00001600
   6       1KIO10,KIO11,KIO12,KIO13,KIO14,KIO15,KIO16,KIO17,KIO18,KIO19,KIO20,00001700
   6       2KIO21,KIO22,KIO23,KIO24,KIO25,KIO26,KIO27,KIO28,KIO29,KIO30      00001800
   1  C                                                                       00001900
   7        COMMON /SOL/ NUMNP,NEQ,NWK,NWM,NWC,NUMEST,MIDEST,MAXEST,NSTE,MA   00002000
   7        COMMON /CONST/ DT,DTA,A0,A1,A2,A3,A4,A5,A6,A7,A8,A9,A10,A11       00002100
   6       1              ,A12,A13,A14,A15,A16,A17,A18,A19,A20,IOPE           00002200
   7        COMMON /LOA/ NLOAD,NPTM                                           00002300
   7        COMMON /VAR/ NG,KPRI,MODEX,KSTEP,ITE,ITEMAX,IREF,IEQREF,INOCMD    00002400
   7        COMMON /NSAPI/ OPVAR(7),TSTART,IEQUIT,NUMREF,IPRI,ISREF           00002500
   7        DIMENSION R(1),RV(1),TIMV(1),NOD(1),IDIRN(1),NCUR(1),FAC(1)       00002600
   7        DIMENSION ID(NDOF,1),RG(NLCUR,1)                                  00002700
   1  C                                                                       00002800
   1  C                                                                       00002900
   1  C     CALCULATION OF LOAD CURVE DATA AT ALL TIME POINTS                 00003000
   1  C                                                                       00003100
   1  C                                                                       00003200
   7        DO 100 L=1,NLCUR                                                  00003300
   7        READ(KIO5,1000) LL,NPTS                                           00003400
   7        IF (L-LL) 80,90,80                                                00003500
   4     80 WRITE(KIO6,2000)                                                  00003600
   7        STOP                                                              00003700
   4     90 WRITE(KIO6,2002) L,NPTS                                           00003800
   7        READ(KIO5,1020)                                                   00003900
   6       *             (TIMV(I),RV(I),I=1,NPTS)                           00004000
   7        WRITE(KIO6,2004)                                                  00004100
   6       *             (TIMV(I),RV(I),I=1,NPTS)                           00004200
   7        IF (MODEX.EQ.0) GO TO 100                                         00004300
   1  C                                                                       00004400
   7        TIME=TSTART + DT                                                  00004500
   7        TIMEP=TSTART + DTA                                                00004600
   7        I=0                                                               00004700
   7        K=1                                                               00004800
   3    120 I=I + 1                                                           00004900
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7        IF (I-NPTS) 190,130,130                                          00005000
   3      130 WRITE(KIO6,2010)                                               00005100
   7          STOP                                                           00005200
   1  C                                                                      00005300
   3      190 DDR=RV(I+1) - RV(I)                                            00005400
   7          DDT=TIMV(I+1) - TIMV(I)                                        00005500
   7          IF (DDT) 110,120,150                                           00005600
   3      110 WRITE(KIO6,2020)                                               00005700
   7          STOP                                                           00005800
   3      150 SLOPE=DDR/DDT                                                  00005900
   3      180 IF (TIMV(I+1) - TIME) 120,140,140                              00006000
   3      140 RG(L,K)=RV(I) + SLOPE*(TIMEP - TIMV(I))                        00006100
   7          TIMEP=TIME + DTA                                               00006200
   7          TIME=TIME + DT                                                 00006300
   7          K=K + 1                                                        00006400
   7          IF (NSTE-K) 100,180,180                                        00006500
   3      100 CONTINUE                                                       00006600
   1  C                                                                      00006700
   1  C                                                                      00006800
   1  C      LOAD CURVE DATA ARE USED TO CALCULATE THE LOAD VECTORS          00006900
   1  C                                                                      00007000
   1  C                                                                      00007100
   7          WRITE(KIO6,2024)                                               00007200
   7          DO 105 I=1,NLOAD                                               00007300
   7          FACT=FAC(I)                                                    00007400
   3      105 IF (FACT.EQ.0.) FAC(I)=1.0                                     00007500
   7          READ(KIO5,1030)                                                00007600
   6         *                (NOD(I),IDIRN(I),NCUR(I),FAC(I),I=1,NLOAD)    00007700
   7          WRITE(KIO6,2030)                                               00007800
   6         *                (NOD(I),IDIRN(I),NCUR(I),FAC(I),I=1,NLOAD)    00007900
   7          IF (MODEX.EQ.0) RETURN                                         00008000
   1  C                                                                      00008100
   7          REWIND KIO13                                                   00008200
   7          DO 200 K=1,NSTE                                                00008300
   7          DO 210 I=1,NEQ                                                 00008400
   3      210 R(I)=0.                                                        00008500
   1  C                                                                      00008600
   7          DO 220 L=1,NLOAD                                               00008700
   7          LN=NOD(L)                                                      00008800
   7          LI=IDIRN(L)                                                    00008900
   7          IF (NDOF.EQ.2) LI=LI - 1                                       00009000
   7          LC=NCUR(L)                                                     00009100
   7          II=ID(LI,LN)                                                   00009200
   7          IF (II) 220,220,240                                            00009300
   3      240 R(II)=R(II) + RG(LC,K)*FAC(L)                                  00009400
   1  C                                                                      00009500
   3      220 CONTINUE                                                       00009600
   1  C                                                                      00009700
   7          WRITE(KIO13) (R(I),I=1,NEQ)                                    00009800
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   3    200 CONTINUE                                                              00009900
   1  C                                                                           00010000
   2   1000 FORMAT (2I5)                                                        00010100
   2   1020 FORMAT (8F10.0)                                                     00010200
   2   1030 FORMAT (3I5,F10.0)                                                  00010300
   2   2000 FORMAT (40H *** ERROR   LOAD CURVES OUT OF ORDER    )               00010400
   2   2002 FORMAT (///4X,25H LOAD FUNCTION NUMBER   =,I5/4X,                   00010500
   6       1             25H NUMBER OF TIME POINTS  =,I5//4X,                  00010600
   6       2             25H TIME VALUE      FUNCTION/)                        00010700
   2   2004 FORMAT (3X,F12.5,2X,E12.4)                                          00010800
   2   2010 FORMAT (50H *** ERROR   TIME IS LARGER THAN IN THE LOAD CURVE  )    00010900
   2   2020 FORMAT (42H *** ERROR   LOAD POINTS ARE OUT OF ORDER  )             00011000
   2   2024 FORMAT (////20H  CONCENTRATED LOADS  // 4X,                         00011100
   6       1        54H NODE    DIRECTION    LOAD CURVE   LOAD CURVE MULTIPL )00011200
   2   2030 FORMAT (1H0,2X,I6,4X,I7,9X,I5,8X,E12.4)                             00011300
   7        RETURN                                                             00011400
   7        END                                                                00011500
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7        SUBROUTINE MATBAR (E,AREA,MODEL,IP,N,MTYPE,XL,EPST,P,EAL,SIG,      00000100
   6       1                   PINT,STRAIN,STRESS)                             00000200
   7        IMPLICIT REAL*8(A-H,O-Z)                                           00000300
   1  C                                                                        00000400
   7        COMMON /VAR/ NG,KPRI,MODEX,KSTEP,ITE,ITEMAX,IREF,IEQREF,INOCMD     00000500
   7        DIMENSION E(1),AREA(1),STRAIN(1),STRESS(1)                         00000600
   1  C                                                                        00000700
   1  C                                                                        00000800
   7        GO TO (10,20), MODEL                                               00000900
   1  C                                                                        00001000
   1  C                                                                        00001100
   1  C     M A T E R I A L L Y   L I N E A R                                  00001200
   1  C                                                                        00001300
   4     10 EA=E(MTYPE)*AREA(MTYPE)                                            00001400
   7        EAL=EA/XL                                                          00001500
   7        SIG=EPST*E(MTYPE) + PINT/AREA(MTYPE)                               00001600
   7        P=EA*EPST + PINT                                                   00001700
   7        RETURN                                                             00001800
   1  C                                                                        00001900
   1  C                                                                        00002000
   1  C     M A T E R I A L L Y   N O N L I N E A R                            00002100
   1  C                                                                        00002200
   1  C        FIND SIG CORRESPONDING TO EPST USING STRESS-STRAIN CURVE        00002300
   1  C        SUPPLIED AS INPUT DATA                                          00002400
   1  C                                                                        00002500
   4     20 DO 25 I=2,IP                                                       00002600
   7        L=I                                                                00002700
   7        IF (EPST.LE.STRAIN(I)) GO TO 30                                    00002800
   4     25 CONTINUE                                                           00002900
   1  C                                                                        00003000
   4     30 DS=STRESS(L) - STRESS(L-1)                                         00003100
   7        DE=STRAIN(L) - STRAIN(L-1)                                         00003200
   7        EA=(DS/DE)*AREA(MTYPE)                                             00003300
   7        EAL=EA/XL                                                          00003400
   7        SIG=STRESS(L-1) + ((EPST-STRAIN(L-1))/DE)*DS + PINT/AREA(MTYPE)    00003500
   7        P=SIG*AREA(MTYPE)                                                  00003600
   7        RETURN                                                             00003700
   1  C                                                                        00003800
   7        END                                                                00003900
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7        SUBROUTINE MATRT2 (N,DEN,PROP)                                        00000100
   7        IMPLICIT REAL*8(A-H,O-Z)                                              00000200
   1  C                                                                           00000300
   1  C                                                                           00000400
   1  C     SUBROUTINE TO PRINT OUT MATERIAL PROPERTIES                           00000500
   1  C     FOR TWO-DIMENSIONAL ELEMENTS                                          00000600
   1  C                                                                           00000700
   7        COMMON/BBCCIO/KIO1,KIO2,KIO3,KIO4,KIO5,KIO6,KIO7,KIO8,KIO9,          00000800
   6       1KIO10,KIO11,KIO12,KIO13,KIO14,KIO15,KIO16,KIO17,KIO18,KIO19,KIO20,00000900
   6       2KIO21,KIO22,KIO23,KIO24,KIO25,KIO26,KIO27,KIO28,KIO29,KIO30        00001000
   1  C                                                                           00001100
   7        COMMON /EL/ IND,ICOUNT,NPAR(20),NUMEG,NEGL,NEGNL,IMASS,IDAMP,ISTAT00001200
   6       1           ,NDOF,KLIN,IEIG,IMASSN,IDAMPN                           00001300
   7        COMMON /CRACK/ GAMMA,STIFAC,SHEFAC                                  00001400
   7        DIMENSION PROP(1)                                                   00001500
   1  C                                                                           00001600
   7        EQUIVALENCE (NPAR(15),MODEL),(NPAR(17),NCON),(NPAR(18),IDW)         00001700
   6       1           ,(NPAR(16),NUMMAT)                                      00001800
   1  C                                                                           00001900
   1  C                                                                           00002000
   7        IF (N.EQ.1) WRITE (6,2000) MODEL,NUMMAT,NCON,IDW                    00002100
   7        WRITE(KIO6,2100) N,DEN                                              00002200
   1  C                                                                           00002300
   7        GO TO ( 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12 ) , MODEL             00002400
   1  C                                                                           00002500
   1  C                                                                           00002600
   1  C.... MODEL = 1    L I N E A R   I S O T R O P I C                         00002700
   1  C                                                                           00002800
   5      1 WRITE(KIO6,2102) (PROP(I),I=1,NCON)                                 00002900
   7        RETURN                                                              00003000
   1  C                                                                           00003100
   1  C                                                                           00003200
   1  C.... MODEL = 2    L I N E A R   O R T H O T R O P I C                     00003300
   1  C                                                                           00003400
   5      2 WRITE(KIO6,2102) (PROP(I),I=1,NCON)                                 00003500
   7        RETURN                                                              00003600
   1  C                                                                           00003700
   1  C                                                                           00003800
   1  C.... MODEL = 3    V A R I A B L E   T A N G E N T   M O D U L I           00003900
   1  C                                                                           00004000
   5      3 WRITE(KIO6,2103) (PROP(I), I=1,NCON)                                00004100
   7        RETURN                                                              00004200
   1  C                                                                           00004300
   1  C                                                                           00004400
   1  C.... MODEL = 4    C U R V E   D E S C R I P T I O N   M O D E L           00004500
   1  C                                                                           00004600
   5      4 GO TO 45                                                            00004700
   1  C                                                                           00004800
   1  C                                                                           00004900
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   1  C.... MODEL = 5   C D M O D   W I T H   T E N S I O N   C U T - O F F   00005000
   1  C                                                                      00005100
   5       5 READ(KIO5,1000) GAMMA,STIFAC,SHEFAC                             00005200
   7         WRITE(KIO6,2200) GAMMA,STIFAC,SHEFAC                            00005300
   4      45 IP=NCON/4                                                       00005400
   7         WRITE(KIO6,2200)                                                00005500
   7         DO 20 I=1,IP                                                    00005600
   4      20 WRITE(KIO6,2210) I,PROP(I),PROP(IP+I),PROP(2*IP+I),PROP(3*IP+I) 00005700
   7         RETURN                                                          00005800
   1  C                                                                      00005900
   1  C                                                                      00006000
   1  C.... MODEL = 6    E L A S T O P L A S T I C (VON MISES YIELD CRITERION)00006100
   1  C                                                                      00006200
   5       6 WRITE(KIO6,2106) (PROP(I),I=1,NCON)                             00006300
   7         RETURN                                                          00006400
   1  C                                                                      00006500
   1  C                                                                      00006600
   5       7 GO TO 50                                                        00006700
   1  C                                                                      00006800
   1  C                                                                      00006900
   1  C.... MODEL = 8    I N C O M P R E S S I B L E   N O N L I N E A R     00007000
   1  C                    E L A S T I C    ( MOONEY-RIVLIN MATERIAL IN STATE 00007100
   1  C                    OF PLANE STRESS )                                 00007200
   1  C                                                                      00007300
   5       8 WRITE(KIO6,2109) PROP(1),PROP(2)                                00007400
   7         RETURN                                                          00007500
   1  C                                                                      00007600
   5       9 GO TO 50                                                        00007700
   1  C                                                                      00007800
   4      10 GO TO 50                                                        00007900
   1  C                                                                      00008000
   4      11 GO TO 50                                                        00008100
   1  C                                                                      00008200
   4      12 GO TO 50                                                        00008300
   1  C                                                                      00008400
   1  C                                                                      00008500
   4      50 WRITE (6,2500) (I,PROP(I), I=1,NCON)                           00008600
   7         RETURN                                                          00008700
   1  C                                                                      00008800
   1  C                                                                      00008900
   2   1000 FORMAT (3F10.0)                                                 00009000
   2   2000 FORMAT (38H M A T E R I A L   D E F I N I T I O N///,           00009100
   6        1        16H MATERIAL MODEL.,12(2H .),16H( NPAR(15)). . =,I5/,  00009200
   6        2        20H     EQ.1, ISOTROPIC/,                             00009300
   6        3        22H     EQ.2, ORTHOTROPIC/,                           00009400
   6        4        34H     EQ.3, VARIABLE TANGENT MODULI/,               00009500
   6        5        53H     EQ.4, CURVE DESCRIPTION MODEL                ,/00009600
   6        7        53H     EQ.5, CURVE DESCRIPTION MODEL (TENSION CUT-OFF) ,/00009700
   6        A        38H     EQ.6, ELASTOPLASTIC ( VON MISES )/,           00009800
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   6       B         42H     EQ.7, ELASTOPLASTIC (DRUCKER-PRAGER)          /,   00009900
   6       C         50H     EQ.8, INCOMPRESSIBLE ELASTIC (MOONEY-RIVLIN)  //, 00010000
   6       5         37H NUMBER OF DIFFERENT SETS OF MATERIAL/,                  00010100
   6       6         14H     CONSTANTS,13(2H .),16H( NPAR(16)). . =,I5//,       00010200
   6       7         40H NUMBER OF MATERIAL CONSTANTS PER SET. .,                00010300
   6       8         16H( NPAR(17)). . =,I5//,                                   00010400
   6       9         32H DIMENSION OF STORAGE ARRAY (WA)/,                       00010500
   6       1         26H     PER INTEGRATION POINT,7(2H .),16H( NPAR(18)). . =,00010600
   6       2         I5///)                                                      00010700
   2    2100 FORMAT (30H MATERIAL CONSTANTS SET NUMBER,6H .... ,I5//,          00010800
   6       1         1H ,4X,29HDEN ..........( DENSITY ).. =, E14.6/)          00010900
   2    2101 FORMAT (1H ,4X,29HE .............( PROP(1) ).. =, E14.6/,         00011000
   6       1         1H ,4X,29HVNU ..........( PROP(2) ).. =, E14.6///)        00011100
   2    2102 FORMAT (1H ,4X,29HE(A) ..........( PROP(1) ).. =, E14.6/,         00011200
   6       1         1H ,4X,29HE(B) ..........( PROP(2) ).. =, E14.6/,         00011300
   6       2         1H ,4X,29HE(C) ..........( PROP(3) ).. =, E14.6/,         00011400
   6       3         1H ,4X,29HVNU(AB) .......( PROP(4) ).. =, E14.6/,         00011500
   6       4         1H ,4X,29HVNU(AC) .......( PROP(5) ).. =, E14.6/,         00011600
   6       5         1H ,4X,29HVNU(BC) .......( PROP(6) ).. =, E14.6/,         00011700
   6       6         1H ,4X,29HG(AB) .........( PROP(7) ).. =, E14.6///)       00011800
   2    2103 FORMAT (1H ,4X,29HK(UNLOAD) ....( PROP(1) ).. =, E14.6/,         00011900
   6       1         1H ,4X,29HK(ZERO) .......( PROP(2) ).. =, E14.6/,         00012000
   6       2         1H ,4X,29HK (ONE) .......( PROP(3) ).. =, E14.6/,         00012100
   6       3         1H ,4X,29HK (TWO) .......( PROP(4) ).. =, E14.6/,         00012200
   6       4         1H ,4X,29HG(ZERO) .......( PROP(5) ).. =, E14.6/,         00012300
   6       5         1H ,4X,29HGAMMA(1) ......( PROP(6) ).. =, E14.6/,         00012400
   6       6         1H ,4X,29HGAMMA(2) ......( PROP(7) ).. =, E14.6///)       00012500
   2    2106 FORMAT (1H ,4X,29HE .............( PROP(1) ).. =, E14.6/,         00012600
   6       1         1H ,4X,29HVNU ..........( PROP(2) ).. =, E14.6/,         00012700
   6       2         1H ,4X,29HYIELD .........( PROP(3) ).. =, E14.6/,         00012800
   6       3         1H ,4X,29HE (HARDEN) ...( PROP(4) ).. =, E14.6///)       00012900
   2    2109 FORMAT (1H ,4X,29HC1 ............( PROP(1) ).. =, E14.6/,         00013000
   6       1         1H ,4X,29HC2 ............( PROP(2) ).. =, E14.6///)       00013100
   2    2200 FORMAT (///                                                       00013200
   6       1         19X,6HVOLUME,8X,7HLOADING,6X,9HUNLOADING,8X,7HLOADING, /  00013300
   6       2         19X,6HSTRAIN,2(3X,12HBULK MODULUS),2X,13HSHEAR MODULUS,   00013400
   6       3           / 1X)                                                    00013500
   2    2210 FORMAT (7H POINT(,I1,2H) ,4E15.4)                                 00013600
   2    2220 FORMAT (31H MATERIAL DENSITY            = (,E13.4,1H),/           00013700
   6       1         31H STIFFNESS REDUCTION FACTOR = (,E13.4,1H),/            00013800
   6       2         31H SHEAR REDUCTION FACTOR     = (,E13.4,1H),1X)          00013900
   2    2500 FORMAT (1H ,4X,5HPROP(,I2,10H) ..... =, E14.6)                    00014000
   1  C                                                                        00014100
   1  C                                                                        00014200
   7        END                                                                00014300
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7        SUBROUTINE MATWRT (N,DEN,PROP)                                        00000100
   7        IMPLICIT REAL*8(A-H,O-Z)                                              00000200
   1  C                                                                           00000300
   1  C                                                                           00000400
   1  C     PROGRAM TO PRINT MATERIAL PROPERTIES                                  00000500
   1  C               FOR THREE-DIMENSIONAL ELEMENTS                              00000600
   1  C                                                                           00000700
   7        COMMON/BBCCIO/KI01,KI02,KI03,KI04,KI05,KI06,KI07,KI08,KI09,          00000800
   6       1KI010,KI011,KI012,KI013,KI014,KI015,KI016,KI017,KI018,KI019,KI020,  00000900
   6       2KI021,KI022,KI023,KI024,KI025,KI026,KI027,KI028,KI029,KI030         00001000
   1  C                                                                           00001100
   7        COMMON /EL/ IND,ICOUNT,NPAR(20),NUMEG,NEGL,NEGNL,IMASS,IDAMP,ISTAT   00001200
   6       1           ,NDOF,KLIN,IEIG,IMASSN,IDAMPN                            00001300
   7        DIMENSION PROP(1)                                                    00001400
   7        EQUIVALENCE (NPAR(15),MODEL),(NPAR(16),NUMMAT),(NPAR(17),NCON)       00001500
   6       1           ,(NPAR(18),IDW)                                          00001600
   1  C                                                                           00001700
   1  C                                                                           00001800
   7        IF (N.EQ.1) WRITE (KI06,2000) MODEL,NUMMAT,NCON,IDW                  00001900
   7        WRITE (KI06,2100) N,DEN                                              00002000
   1  C                                                                           00002100
   7        GO TO ( 1, 2, 3, 4, 5, 6), MODEL                                     00002200
   1  C                                                                           00002300
   1  C                                                                           00002400
   1  C.... MODEL = 1    L I N E A R    I S O T R O P I C                        00002500
   1  C                                                                           00002600
   5      1 WRITE(KI06,2101) (PROP(I), I=1,NCON)                                  00002700
   7        RETURN                                                               00002800
   1  C                                                                           00002900
   1  C                                                                           00003000
   1  C.... MODEL = 2    L I N E A R    O R T H O T R O P I C                    00003100
   1  C                                                                           00003200
   5      2 WRITE(KI06,2102) (PROP(I),I=1,NCON)                                   00003300
   7        RETURN                                                               00003400
   1  C                                                                           00003500
   1  C                                                                           00003600
   1  C.... MODEL = 3    C U R V E    D E S C R I P T I O N    M O D E L        00003700
   1  C                                                                           00003800
   5      3 IP=NCON/4                                                           00003900
   7        WRITE(KI06,2200)                                                     00004000
   7        DO 20 I=1,IP                                                         00004100
   4     20 WRITE(KI06,2210) I,PROP(I),PROP(IP+I),PROP(2*IP+I),PROP(3*IP+I)     00004200
   1  C                                                                           00004300
   1  C                                                                           00004400
   1  C                                                                           00004500
   1  C                                                                           00004600
   1  C                                                                           00004700
   1  C                                                                           00004800
   7        RETURN                                                               00004900
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   1  C                                                                      00005000
   1  C                                                                      00005100
   1  C.... MODEL .GE. 4   U S E R - S U P P L I E D  M A T E R I A L       00005200
   1  C                                                                      00005300
   1  C                    M O D E L S                                       00005400
   1  C                                                                      00005500
   1  C                                                                      00005600
   5      4 GO TO 50                                                        00005700
   5      5 GO TO 50                                                        00005800
   5      6 GO TO 50                                                        00005900
   1  C                                                                      00006000
   4     50 WRITE (6,2500) (I,PROP(I), I=1,NCON)                            00006100
   7        RETURN                                                          00006200
   1  C                                                                      00006300
   1  C                                                                      00006400
   1  C                                                                      00006500
   2   2000 FORMAT (38H M A T E R I A L   D E F I N I T I O N///,            00006600
   6         1        16H MATERIAL MODEL.,12(2H .),16H( NPAR(15)). . =,I5/,  00006700
   6         2        35H     EQ.1, LINEAR ELASTIC ISOTROPIC                /,00006800
   6         4        46H     EQ.3, NONLINEAR CURVE DESCRIPTION MODEL   //,00006900
   6         5        37H NUMBER OF DIFFERENT SETS OF MATERIAL/,            00007000
   6         6        14H     CONSTANTS,13(2H .),16H( NPAR(16)). . =,I5//,  00007100
   6         7        40H NUMBER OF MATERIAL CONSTANTS PER SET. .,          00007200
   6         8        16H( NPAR(17)). . =,I5//,                             00007300
   6         9        32H DIMENSION OF STORAGE ARRAY (WA)/,                 00007400
   6         1        26H     PER INTEGRATION POINT,7(2H .),16H( NPAR(18)). . =,00007500
   6         2        I5///)                                                00007600
   2   2100 FORMAT (30H MATERIAL CONSTANTS SET NUMBER,6H .... ,I5//,        00007700
   6         1        1H ,4X,29HDEN ..........( DENSITY ).. =, E14.6/)      00007800
   2   2101 FORMAT (1H ,4X,29HE ............( PROP(1) ).. =, E14.6/,        00007900
   6         1        1H ,4X,29HVNU ..........( PROP(2) ).. =, E14.6///)    00008000
   2   2102 FORMAT (1H ,4X,29HE(A) .........( PROP(1) ).. =, E14.6/,        00008100
   6         1        1H ,4X,29HE(B) .........( PROP(2) ).. =, E14.6/,      00008200
   6         2        1H ,4X,29HE(C) .........( PROP(3) ).. =, E14.6/,      00008300
   6         3        1H ,4X,29HVNU(AB) ......( PROP(4) ).. =, E14.6/,     00008400
   6         4        1H ,4X,29HVNU(AC) ......( PROP(5) ).. =, E14.6/,     00008500
   6         5        1H ,4X,29HVNU(BC) ......( PROP(6) ).. =, E14.6/,     00008600
   6         6        1H ,4X,29HG(AB) ........( PROP(7) ).. =, E14.6///)   00008700
   2   2200 FORMAT (///                                                     00008800
   6         1        19X,6HVOLUME,8X,7HLOADING,6X,9HUNLOADING,8X,7HLOADING, /  00008900
   6         2        19X,6HSTRAIN,2(3X,12HBULK MODULUS),2X,13HSHEAR MODULUS,   00009000
   6         3           / 1X)                                             00009100
   2   2210 FORMAT (7H POINT(,I1,2H) ,4E15.4)                               00009200
   2   2500 FORMAT (1H ,4X,5HPROP(,I2,10H) ..... =, E14.6)                  00009300
   1  C                                                                      00009400
   1  C                                                                      00009500
   1  C                                                                      00009600
   7        END                                                             00009700
```

PROJECT: TSO
LIBRARY: TDA10
TYPE: FH2079
MEMBER: MAXAEX
LEVEL: 01.01
USERID: TDA10
DATE: 82/08/25
TIME: 14:23
PAGE: 01 OF 01

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8  MOD FLAGS

   1  C   SUBROUTINE FOR ALLOCATING SPCE ACCORDING TO NUMBER EQUATION (NEQ)   00000100
   1  C   PROGRAMED BY S.N. MUKHERJEE. BBC,BADEN ,AUG 1976.                  00000201   **
   1  C                                                                      00000300
   7        SUBROUTINE MAXAEX(INEW,IOLD,NEQ)                                00000400
   7        DIMENSION INEW(1),IOLD(1)                                       00000500
   7        NN=NEQ+1                                                        00000600
   7        DO 10 I=1,NN                                                    00000700
   4     10 INEW(I) = IOLD(I)                                               00000800
   7        RETURN                                                          00000900
   7        END                                                             00001000
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7          SUBROUTINE MAXMIN (STRESS,P1,P2,AG)                          00000100
   7          IMPLICIT REAL*8(A-H,O-Z)                                     00000200
   1  C                                                                    00000300
   7          DIMENSION STRESS(1)                                          00000400
   1  C                                                                    00000500
   1  C                                                                    00000600
   7          CC = (STRESS(1)+STRESS(2)) * 0.5                             00000700
   7          BB = (STRESS(1)-STRESS(2)) * 0.5                             00000800
   7          CR = DSQRT(BB**2 +STRESS(3)**2)                              00000900
   7          P1 =  CC+CR                                                  00001000
   7          P2 =  CC-CR                                                  00001100
   7          AG=45.0                                                      00001200
   7          IF(DABS(BB).LT.1.0D-8) RETURN                                00001300
   1  C                                                                    00001400
   7          AG = 28.648*DATAN2(STRESS(3),BB)                             00001500
   1  C                                                                    00001600
   7          RETURN                                                       00001700
   1  C                                                                    00001800
   7          END                                                          00001900
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7        SUBROUTINE MIDEP (TAU,DEPS,DP)                                        00000100
   7        IMPLICIT REAL*8(A-H,O-Z)                                              00000200
   1  C                                                                           00000300
   1  C         FORMS THE ELASTO-PLASTIC MATERIAL MATRIX                          00000400
   1  C                                                                           00000500
   7        COMMON /EL/ IND,ICOUNT,NPAR(20),NUMEG,NEGL,NEGNL,IMASS,IDAMP,ISTAT00000600
   6       1          ,NDOF,KLIN,IEIG,IMASSN,IDAMPN                             00000700
   7        COMMON /VMISES/A2,B2,C2,D2,A1,B1,C1,D1,YLD,BM,ISR,IST                00000800
   7        COMMON /MATMOD/ STRESS(4),STRAIN(4),C(4,4),IPT,NEL                   00000900
   7        DIMENSION TAU(1),DEPS(1),DP(1)                                       00001000
   1  C                                                                           00001100
   7        EQUIVALENCE (NPAR(5),ITYP2D)                                         00001200
   1  C                                                                           00001300
   1  C                                                                           00001400
   7        SM = (TAU(1)+TAU(2)+TAU(4))/3.                                       00001500
   7        SX = TAU(1) - SM                                                     00001600
   7        SY = TAU(2) - SM                                                     00001700
   7        SS = TAU(3)                                                          00001800
   7        SZ = TAU(4) - SM                                                     00001900
   7        HPRIME=2.*D1/3.                                                      00002000
   7        BETA=1.5/YLD/YLD/(1.+HPRIME/A1)                                      00002100
   1  C                                                                           00002200
   7        IF (ITYP2D - 1) 10,11,12                                             00002300
   4     10 WP= SX*DEPS(1) + SY*DEPS(2) + SS*DEPS(3) + SZ*DEPS(4)                00002400
   7        GO TO 15                                                             00002500
   1  C                                                                           00002600
   4     11 WP= SX*DEPS(1) + SY*DEPS(2) + SS*DEPS(3)                             00002700
   7        GO TO 15                                                             00002800
   1  C                                                                           00002900
   4     12 DP1= A1 * (C1 - BETA*SX*SZ)                                          00003000
   7        DP2= A1 * (C1 - BETA*SY*SZ)                                          00003100
   7        DP3= A1 * (   - BETA*SS*SZ)                                          00003200
   7        DP4= A1 * (B1 - BETA*SZ*SZ)                                          00003300
   1  C                                                                           00003400
   7        DEPS(4)= (-DP1*DEPS(1)-DP2*DEPS(2)-DP3*DEPS(3))/DP4                  00003500
   7        WP= SX*DEPS(1) + SY*DEPS(2) + SS*DEPS(3) + SZ*DEPS(4)                00003600
   1  C                                                                           00003700
   4     15 IF (WP.LT.0.) BETA=0.                                                00003800
   1  C                                                                           00003900
   7        DP ( 1) = A1 * (B1 - BETA*SX*SX)                                     00004000
   7        DP ( 2) = A1 * (C1 - BETA*SX*SY)                                     00004100
   7        DP ( 3) = A1 * (   - BETA*SX*SS)                                     00004200
   7        DP ( 4) = A1 * (C1 - BETA*SX*SZ)                                     00004300
   7        DP ( 5) = DP (2)                                                     00004400
   7        DP ( 6) = A1 * (B1 - BETA*SY*SY)                                     00004500
   7        DP ( 7) = A1 * (   - BETA*SY*SS)                                     00004600
   7        DP ( 8) = A1 * (C1 - BETA*SY*SZ)                                     00004700
   7        DP ( 9) = DP (3)                                                     00004800
   7        DP (10) = DP (7)                                                     00004900
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8
   7        DP (11) = A1 * (.5 - BETA*SS*SS)                                    00005000
   7        DP (12) = A1 * (   - BETA*SZ*SS)                                    00005100
   1  C                                                                         00005200
   7        IF (ITYP2D.EQ.1) RETURN                                             00005300
   1  C                                                                         00005400
   7        DP (13) = DP (4)                                                    00005500
   7        DP (14) = DP (8)                                                    00005600
   7        DP (15) = DP(12)                                                    00005700
   7        DP (16) = A1 * (B1 - BETA*SZ*SZ)                                    00005800
   1  C                                                                         00005900
   7        IF (ITYP2D.EQ.0) RETURN                                             00006000
   1  C                                                                         00006100
   1  C     PLANE STRESS / MODIFY DP MATRIX                                     00006200
   1  C                                                                         00006300
   7        DO 120 I=1,3                                                        00006400
   7        A=C(I,4)/C(4,4)                                                     00006500
   7        DO 120 J=I,3                                                        00006600
   7        C(I,J)=C(I,J) - C(4,J)*A                                            00006700
   3    120 C(J,I) = C(I,J)                                                     00006800
   7        IF (WP.LT.0.0) DEPS(4)=D2*(DEPS(1) + DEPS(2))                       00006900
   7        STRAIN(4)=STRAIN(4) + DEPS(4)                                       00007000
   1  C                                                                         00007100
   1  C                                                                         00007200
   7        RETURN                                                              00007300
   7        END                                                                 00007400
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7        SUBROUTINE MLTPLY(A,B,C,MAXA,NEQ)                                00000100
   7        IMPLICIT REAL*8(A-H,O-Z)                                         00000200
   1  C                                                                      00000300
   1  C                                                                      00000400
   1  C .   P R O G R A M                                                  . 00000500
   1  C .     . TO CALCULATE   A = B*C   ,   WHERE  B  IS STORED IN        . 00000600
   1  C .       COMPACTED FORM  ,  A  AND  C  ARE VECTORS                  . 00000700
   1  C                                                                      00000800
   1  C                                                                      00000900
   7        DIMENSION A(1),B(1),C(1),MAXA(1)                                 00001000
   1  C                                                                      00001100
   7        DO 99 I=1,NEQ                                                    00001200
   4     99 A(I)=0.                                                          00001300
   7        DO 100 I=1,NEQ                                                   00001400
   7        KL=MAXA(I)                                                       00001500
   7        KU=MAXA(I+1) - 1                                                 00001600
   7        II=I + 1                                                         00001700
   7        CC=C(I)                                                          00001800
   7        DO 100 KK=KL,KU                                                  00001900
   7        II=II - 1                                                        00002000
   3    100 A(II)=A(II) + B(KK)*CC                                           00002100
   7        DO 200 I=2,NEQ                                                   00002200
   7        KL=MAXA(I) + 1                                                   00002300
   7        KU=MAXA(I+1) - 1                                                 00002400
   7        IF(KU-KL) 200,210,210                                            00002500
   3    210 II=I                                                             00002600
   7        AA=0.                                                            00002700
   7        DO 220 KK=KL,KU                                                  00002800
   7        II=II - 1                                                        00002900
   3    220 AA=AA + B(KK)*C(II)                                              00003000
   7        A(I)=A(I) + AA                                                   00003100
   3    200 CONTINUE                                                         00003200
   1  C                                                                      00003300
   1  C                                                                      00003400
   7        RETURN                                                           00003500
   7        END                                                              00003600
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7        SUBROUTINE MOONEY (NEL,PROP,C,STRESS,STRAIN,IPT)                      00000100
   7        IMPLICIT REAL*8(A-H,O-Z)                                              00000200
   1  C                                                                           00000300
   1  C                                                                           00000400
   1  C     INCOMPRESSIBLE NONLINEAR ELASTIC MATERIAL                             00000500
   1  C         ( MOONEY-RIVLIN DESCRIPTION IN STATE OF PLANE STRESS )            00000600
   1  C                                                                           00000700
   1  C                                                                           00000800
   7        DIMENSION PROP(1),STRAIN(1),STRESS(1),C(4,1)                          00000900
   7        COMMON/BBCCIO/KIO1,KIO2,KIO3,KIO4,KIO5,KIO6,KIO7,KIO8,KIO9,           00001000
   6       1KIO10,KIO11,KIO12,KIO13,KIO14,KIO15,KIO16,KIO17,KIO18,KIO19,KIO20,00001100
   6       2KIO21,KIO22,KIO23,KIO24,KIO25,KIO26,KIO27,KIO28,KIO29,KIO30          00001200
   7        COMMON /EL/ IND,ICOUNT,NPAR(20),NUMEG,NEGL,NEGNL,IMASS,IDAMP,ISTAT00001300
   6       1           ,NDOF,KLIN,IEIG,IMASSN,IDAMPN                             00001400
   7        COMMON /VAR/ NG,KPRI,MODEX,KSTEP,ITE,ITEMAX,IREF,IEQREF,INOCMD       00001500
   7        COMMON /DISDER/DISD(5)                                                00001600
   7        DIMENSION CS(4)                                                       00001700
   7        EQUIVALENCE (CS(1),CS11),(CS(2),CS22),(CS(3),CS12),(CS(4),CS33)       00001800
   6       1           ,(NPAR(3),INDNL)                                          00001900
   1  C                                                                           00002000
   1  C                                                                           00002100
   1  C                                                                           00002200
   1  C                                                                           00002300
   1  C     1.   DEFINE MATERIAL CONSTANTS                                        00002400
   1  C                                                                           00002500
   7        C1=PROP(1)                                                            00002600
   7        C2=PROP(2)                                                            00002700
   1  C                                                                           00002800
   1  C     2.   CALCULATE DEFORMATION GRADIENT                                   00002900
   1  C                                                                           00003000
   7        C11=2.*STRAIN(1) + 1.                                                 00003100
   7        C12=STRAIN(3)                                                         00003200
   7        C22=2.*STRAIN(2) + 1.                                                 00003300
   1  C                                                                           00003400
   1  C     3. CALCULATE EXTENSION RATIO                                          00003500
   1  C                                                                           00003600
   7        DENOM=C11*C22 - C12*C12                                               00003700
   7        EX2=1./DENOM                                                          00003800
   7        EX4=EX2*EX2                                                           00003900
   7        C1122=C11 + C22                                                       00004000
   1  C                                                                           00004100
   1  C     4.   CALCULATE STRESSES                                               00004200
   1  C                                                                           00004300
   7        T1=2.*(C1 + C2*EX2)                                                   00004400
   7        T2=2.*(C1*EX4 - C2*(1. - EX4*C1122))                                  00004500
   7        STRESS(1)=T1 - T2*C22                                                 00004600
   7        STRESS(2)=T1 - T2*C11                                                 00004700
   7        STRESS(3)=T2*C12                                                      00004800
   7        STRESS(4)=0.0                                                         00004900
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   1  C                                                                    00005000
   1  C     PRINT STRESSES                                                 00005100
   1  C                                                                    00005200
   7        IF (KPRI.NE.0) GO TO 10                                        00005300
   7        IF (IPT.EQ.1 .AND. NEL.EQ.1)                                   00005400
   6       * WRITE(KIO6,2000)                                              00005500
   7        IF (IPT.NE.1) GO TO 11                                         00005600
   7        WRITE(KIO6,2034) NEL                                           00005700
   1  C                                                                    00005800
   1  C     DISD   = DERIVATIVES OF DISPLACEMENTS WITH RESPECT TO          00005900
   1  C              REFERENCE STATE COORDINATES                           00006000
   1  C                                                                    00006100
   1  C     CALCULATE THE DEFORMATION GRADIENT                             00006200
   1  C                                                                    00006300
   4     11 X11=1.+DISD(1)                                                 00006400
   7        X12=   DISD(3)                                                 00006500
   7        X22=1.+DISD(2)                                                 00006600
   7        X21=   DISD(4)                                                 00006700
   7        X33=1.                                                         00006800
   1  C                                                                    00006900
   1  C     DET    = (INITIAL DENSITY / CURRENT DENSITY)                   00007000
   1  C     INCOMPRESSIBILITY CONDITION YIELDS  DET=1.                     00007100
   1  C                                                                    00007200
   7        DET=1.                                                         00007300
   1  C                                                                    00007400
   1  C     STRESS = 2ND PIOLA-KIRCHHOFF STRESSES                          00007500
   1  C                                                                    00007600
   7        S11=STRESS(1)                                                  00007700
   7        S22=STRESS(2)                                                  00007800
   7        S12=STRESS(3)                                                  00007900
   1  C                                                                    00008000
   1  C     CS     = CAUCHY STRESSES                                       00008100
   1  C                                                                    00008200
   7        CS11=DET * (S11*X11*X11 + 2.*S12*X11*X12 + S22*X12*X12)        00008300
   7        CS22=DET * (S11*X21*X21 + 2.*S12*X21*X22 + S22*X22*X22)        00008400
   7        CS12=DET * (S11*X11*X21 + S12*(X11*X22+X12*X21) + S22*X12*X22) 00008500
   7        CS33=0.                                                        00008600
   1  C                                                                    00008700
   4     15 CALL MAXMIN (CS,P1,P2,AG)                                      00008800
   7        WRITE(KIO6,2040)                                               00008900
   6       *              IPT,(CS(I),I=1,4),P1,P2,AG                      00009000
   1  C                                                                    00009100
   1  C     RETURN IF STIFFNESS NOT TO BE CALCULATED                       00009200
   1  C                                                                    00009300
   4     10 IF (ICOUNT.GT.2) RETURN                                        00009400
   7        IF (IREF.NE.0) RETURN                                          00009500
   1  C                                                                    00009600
   1  C     5. CALCULATE STRESS-STRAIN MATRIX                              00009700
   1  C                                                                    00009800
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7        C1F=C1*EX4*4.                                                          00009900
   7        C2F=C2*EX4*4.                                                          00010000
   7        EC11=EX2*C11                                                           00010100
   7        EC22=EX2*C22                                                           00010200
   7        EC1=EC11*C1122                                                         00010300
   7        EC2=EC22*C1122                                                         00010400
   7        EC12=EC11*C22                                                          00010500
   1  C                                                                            00010600
   7        C(1,1)=2.*C22*(C1F*EC22 + C2F*(-1. + EC2))                             00010700
   7        C(1,2)=C1F*(-1. + 2.*EC12) + C2F*(1./EX4 + 2.*C1122*(-1. + EC12))      00010800
   7        C(1,3)=C12*(-2.*C1F*EC22 + C2F*(1. - 2.*EC2))                          00010900
   7        C(2,2)=2.*C11*(C1F*EC11 + C2F*(-1. + EC1))                             00011000
   7        C(2,3)=C12*(-2.*C1F*EC11 + C2F*(1. - 2.*EC1))                          00011100
   7        C(3,3)=(2.*C12*C12*EX2 + 0.5)*(C1F + C2F*C1122) - 2.*C2                00011200
   7        C(2,1)=C(1,2)                                                          00011300
   7        C(3,1)=C(1,3)                                                          00011400
   7        C(3,2)=C(2,3)                                                          00011500
   1  C                                                                            00011600
   7        RETURN                                                                 00011700
   1  C                                                                            00011800
   1  C                                                                            00011900
   2   2000 FORMAT (//24H  ELEMENT   INTEGRATION /19H  NUMBER      POINT ,4X,     00012000
   6       1         50H  SIGMA-X1  SIGMA-X2   TAU-X12  SIGMA-X3  SIGMA-P+,      00012100
   6       2         40H  SIGMA-P-  ANGLE                                 /)      00012200
   2   2034 FORMAT (/I9)                                                          00012300
   2   2040 FORMAT (9X,I10,4X,6E10.3,F7.2)                                        00012400
   1  C                                                                            00012500
   1  C                                                                            00012600
   7        END                                                                    00012700
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7        SUBROUTINE MULT (A,B,C,MAXA,NEQ)                                00000100
   7        IMPLICIT REAL*8(A-H,O-Z)                                        00000200
   1  C                                                                     00000300
   1  C . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 00000400
   1  C .                                                                 . 00000500
   1  C .   P R O G R A M                                                 . 00000600
   1  C .      . TO CALCULATE   A = B*C    ,   WHERE  B  IS STORED IN     . 00000700
   1  C .        COMPACTED FORM  ,  A  AND  C  ARE VECTORS                 . 00000800
   1  C .                                                                 . 00000900
   1  C . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 00001000
   1  C                                                                     00001100
   7        DIMENSION A(1),B(1),C(1),MAXA(1)                                00001200
   1  C                                                                     00001300
   7        DO 100 I=1,NEQ                                                  00001400
   7        KL=MAXA(I)                                                      00001500
   7        KU=MAXA(I+1) - 1                                                00001600
   7        II=I + 1                                                        00001700
   7        CC=C(I)                                                         00001800
   7        DO 100 KK=KL,KU                                                 00001900
   7        II=II - 1                                                       00002000
   3    100 A(II)=A(II) - B(KK)*CC                                          00002100
   7        IF (NEQ.EQ.1) RETURN                                            00002200
   7        DO 200 I=2,NEQ                                                  00002300
   7        KL=MAXA(I) + 1                                                  00002400
   7        KU=MAXA(I+1) - 1                                                00002500
   7        IF (KU-KL) 200,210,210                                          00002600
   3    210 II=I                                                            00002700
   7        AA=0.                                                           00002800
   7        DO 220 KK=KL,KU                                                 00002900
   7        II=II - 1                                                       00003000
   3    220 AA=AA + B(KK)*C(II)                                             00003100
   7        A(I)=A(I) - AA                                                  00003200
   3    200 CONTINUE                                                        00003300
   1  C                                                                     00003400
   7        RETURN                                                          00003500
   7        END                                                             00003600
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7        SUBROUTINE NEWDAV (DISP,VEL,ACC,DISPI)                          00000100
   7        IMPLICIT REAL*8(A-H,O-Z)                                        00000200
   1  C                                                                     00000300
   1  C . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 00000400
   1  C .                                                                   . 00000500
   1  C .   P R O G R A M                                                   . 00000600
   1  C .      . TO CALCULATE NEW DISPLACEMENTS, VELOCITIES, AND            . 00000700
   1  C .        ACCELERATIONS AT TIME T+DELTA(T)                           . 00000800
   1  C .                                                                   . 00000900
   1  C .                                                                   . 00001000
   1  C . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 00001100
   1  C                                                                     00001200
   7        COMMON /CONST/ DT,DTA,A0,A1,A2,A3,A4,A5,A6,A7,A8,A9,A10,A11     00001300
   6       1              ,A12,A13,A14,A15,A16,A17,A18,A19,A20,IOPE        00001400
   7        COMMON /SOL/ NUMNP,NEQ,NWK,NWM,NWC,NUMEST,MIDEST,MAXEST,NSTE,MA 00001500
   7        COMMON /EL/ IND,ICOUNT,NPAR(20),NUMEG,NEGL,NEGNL,IMASS,IDAMP,ISTAT00001600
   6       1           ,NDOF,KLIN,IEIG,IMASSN,IDAMPN                       00001700
   7        DIMENSION DISP(1),VEL(1),ACC(1),DISPI(1)                        00001800
   1  C                                                                     00001900
   7        IF (ISTAT.GE.1) GO TO 10                                        00002000
   1  C                                                                     00002100
   1  C                                                                     00002200
   1  C     S T A T I C S                                                   00002300
   1  C                                                                     00002400
   1  C                                                                     00002500
   7        IF (KLIN.GT.0) GO TO 60                                         00002600
   1  C                                                                     00002700
   1  C     LINEAR ANALYSIS                                                 00002800
   1  C                                                                     00002900
   7        DO 50 I=1,NEQ                                                   00003000
   4     50 DISP(I)=DISPI(I)                                                00003100
   7        GO TO 70                                                        00003200
   1  C                                                                     00003300
   1  C     NONLINEAR ANALYSIS                                              00003400
   1  C                                                                     00003500
   4     60 DO 40 I=1,NEQ                                                   00003600
   4     40 DISP(I)=DISP(I) + DISPI(I)                                      00003700
   4     70 RETURN                                                          00003800
   1  C                                                                     00003900
   1  C                                                                     00004000
   1  C     D Y N A M I C S                                                 00004100
   1  C                                                                     00004200
   1  C                                                                     00004300
   4     10 IF (KLIN.GT.0) GO TO 90                                         00004400
   1  C                                                                     00004500
   1  C     LINEAR ANALYSIS                                                 00004600
   1  C                                                                     00004700
   7        DO 80 I=1,NEQ                                                   00004800
   4     80 DISPI(I)=DISPI(I) - DISP(I)                                     00004900
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   4     90 CONTINUE                                                          00005000
   1  C                                                                       00005100
   1  C     NONLINEAR ANALYSIS                                               00005200
   1  C                                                                       00005300
   7        IF (IOPE.NE.1) GO TO 101                                         00005400
   1  C                                                                       00005500
   1  C     THETA .NE. 1.                                                    00005600
   1  C                                                                       00005700
   7        DO 100 I=1,NEQ                                                   00005800
   7        UTT=ACC(I)                                                       00005900
   7        UUT=VEL(I)                                                       00006000
   7        ACC(I) =A6 *DISPI(I)+A7 *UUT+A8 *UTT                             00006100
   7        VEL(I) =UUT + A9*(ACC(I) + UTT)                                  00006200
   7        DISP(I)=DISP(I) + DT*UUT + A10*(ACC(I) + 2.*UTT)                 00006300
   3    100 CONTINUE                                                         00006400
   7        GOTO 105                                                         00006500
   3    101 CONTINUE                                                         00006600
   1  C                                                                       00006700
   1  C     THETA .EQ. 1.                                                    00006800
   1  C                                                                       00006900
   7        DO 102 I=1,NEQ                                                   00007000
   7        UTT=ACC(I)                                                       00007100
   7        UUT=VEL(I)                                                       00007200
   7        ACC(I) =A6 *DISPI(I)+A7 *UUT+A8 *UTT                             00007300
   7        VEL(I) =UUT + A9*UTT + A10*ACC(I)                                00007400
   7        DISP(I)=    DISPI(I)+DISP(I)                                     00007500
   3    102 CONTINUE                                                         00007600
   3    105 CONTINUE                                                         00007700
   1  C                                                                       00007800
   7        RETURN                                                           00007900
   7        END                                                              00008000
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7        SUBROUTINE NODMAS  (ID,XMN,NDOF)                                    00000100
   7        IMPLICIT REAL*8(A-H,O-Z)                                            00000200
   1  C                                                                         00000300
   1  C                                                                         00000400
   1  C . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 00000500
   1  C .                                                                     . 00000600
   1  C .   READS CONCENTRATED NODAL MASSES AND DAMPERS                       . 00000700
   1  C .                                                                     . 00000800
   1  C .   FINDS NODAL MASS VECTOR OR NODAL DAMPING VECTOR  ( XMN )          . 00000900
   1  C .                                                                     . 00001000
   1  C .   WRITES NODAL MASS AND DAMPING VECTORS ON TAPE 21                  . 00001100
   1  C .                                                                     . 00001200
   1  C . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 00001300
   1  C                                                                         00001400
   1  C                                                                         00001500
   7        DIMENSION ID(NDOF,1),XMN(1),XMASS(6),XDAMP(6)                       00001600
   7        COMMON/BBCCIO/KIO1,KIO2,KIO3,KIO4,KIO5,KIO6,KIO7,KIO8,KIO9,        00001700
   6       1KIO10,KIO11,KIO12,KIO13,KIO14,KIO15,KIO16,KIO17,KIO18,KIO19,KIO20,00001800
   6       2KIO21,KIO22,KIO23,KIO24,KIO25,KIO26,KIO27,KIO28,KIO29,KIO30       00001900
   7        COMMON /EL/ IND,ICOUNT,NPAR(20),NUMEG,NEGL,NEGNL,IMASS,IDAMP,ISTAT00002000
   6       1           ,NDOFDM,KLIN,IEIG,IMASSN,IDAMPN                        00002100
   7        COMMON /SOL/ NUMNP,NEQ,NWK,NWM,NWC,NUMEST,MIDEST,MAXEST,NSTE,MA   00002200
   1  C                                                                         00002300
   1  C                                                                         00002400
   1  C     N O D A L   M A S S E S                                             00002500
   1  C                                                                         00002600
   1  C                                                                         00002700
   7        REWIND KIO21                                                        00002800
   7        IF (IMASSN.EQ.0) GO TO 200                                          00002900
   1  C                                                                         00003000
   7        NN=3                                                                00003100
   7        IF (NDOF.GT.3) NN=6                                                 00003200
   7        DO 80 J=1,NEQ                                                       00003300
   4     80 XMN(J)=0.                                                           00003400
   7        WRITE(KIO6,2000)                                                    00003500
   7        DO 100 IN=1,IMASSN                                                  00003600
   7        READ(KIO5,1000)                                                     00003700
   6       *              N,(XMASS(I),I=1,NN)                                  00003800
   7        DO 90 J=1,NDOF                                                      00003900
   7        JJ=ID(J,N)                                                          00004000
   7        IF (JJ.EQ.0) GO TO 90                                               00004100
   7        JM=J                                                                00004200
   7        IF (NDOF.EQ.2) JM=J + 1                                             00004300
   7        XMN(JJ)=XMN(JJ) + XMASS(JM)                                         00004400
   4     90 CONTINUE                                                            00004500
   3    100  WRITE(KIO6,2001)                                                   00004600
   6       *               N,(XMASS(I),I=1,NN)                                 00004700
   1  C                                                                         00004800
   1  C     STORE NODAL MASS VECTOR ON TAPE 21                                  00004900
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   1  C                                                                      00005000
   7        WRITE(KIO21) (XMN(I) , I=1,NEQ)                                 00005100
   1  C                                                                      00005200
   1  C                                                                      00005300
   1  C     N O D A L   D A M P E R S                                        00005400
   1  C                                                                      00005500
   1  C                                                                      00005600
   3    200 IF (IDAMPN.EQ.0) RETURN                                         00005700
   1  C                                                                      00005800
   7        NN=3                                                            00005900
   7        IF (NDOF.GT.3) NN=6                                             00006000
   7        DO 210 I=1,NEQ                                                  00006100
   3    210 XMN(I)=0.0                                                      00006200
   7        WRITE(KIO6,2010)                                                00006300
   7        DO 300 IN=1,IDAMPN                                              00006400
   7        READ(KIO5,1000)                                                 00006500
   6       *              N,(XDAMP(I),I=1,NN)                               00006600
   7        DO 290 J=1,NDOF                                                 00006700
   7        JJ=ID(J,N)                                                      00006800
   7        IF (JJ.EQ.0) GO TO 290                                          00006900
   7        JM=J                                                            00007000
   7        IF (NDOF.EQ.2) JM=J + 1                                         00007100
   7        XMN(JJ)=XMN(JJ) + XDAMP(JM)                                     00007200
   3    290 CONTINUE                                                        00007300
   3    300 WRITE(KIO6,2001)                                                00007400
   6       *                N,(XDAMP(I),I=1,NN)                            00007500
   1  C     STORE NODAL DAMPER VECTOR ON TAPE 21                            00007600
   1  C                                                                      00007700
   1  C                                                                      00007800
   7        WRITE(KIO21) (XMN(I),I=1,NEQ)                                   00007900
   1  C                                                                      00008000
   1  C                                                                      00008100
   7        RETURN                                                          00008200
   1  C                                                                      00008300
   1  C                                                                      00008400
   2   1000 FORMAT (I10,6F10.0)                                             00008500
   2   2000 FORMAT (1H1,31H N O D A L   M A S S   D A T A   ///             00008600
   6      1        ,7H   NODE,11H     X-MASS,11H     Y-MASS,11H     Z-MASS//)00008700
   2   2001 FORMAT (I7,6E11.3)                                              00008800
   2   2010 FORMAT (1H1,37H N O D A L   D A M P E R S   D A T A       ///    00008900
   6      1        ,7H   NODE,11H   X-DAMPER,11H   Y-DAMPER,11H   Z-DAMPER//)00009000
   1  C                                                                      00009100
   1  C                                                                      00009200
   7        END                                                             00009300
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7        SUBROUTINE NONSPI                                                      00000100
   7        IMPLICIT REAL*8(A-H,O-Z)                                               00000200
                                                                                   00000300
   1  C
   7        COMMON/BBCCIO/KIO1,KIO2,KIO3,KIO4,KIO5,KIO6,KIO7,KIO8,KIO9,            00000400
   6       1KIO10,KIO11,KIO12,KIO13,KIO14,KIO15,KIO16,KIO17,KIO18,KIO19,KIO20,     00000500
   6       2KIO21,KIO22,KIO23,KIO24,KIO25,KIO26,KIO27,KIO28,KIO29,KIO30           00000600
                                                                                   00000700
   1  C
   7        COMMON /SOL/ NUMNP,NEQ,NWK,NWM,NWC,NUMEST,MIDEST,MAXEST,NSTE,MA        00000800
   7        COMMON /DIM/ N1,N2,N3,N4,N5,N6,N7,N8,N9,N10,N11,N12,N13,N14,N15        00000900
   7        COMMON /EL/ IND,ICOUNT,NPAR(20),NUMEG,NEGL,NEGNL,IMASS,IDAMP,ISTAT    00001000
   6       1           ,NDOF,KLIN,IEIG,IMASSN,IDAMPN                           00001100
   7        COMMON /CONST/ DT,DTA,A0,A1,A2,A3,A4,A5,A6,A7,A8,A9,A10,A11           00001200
   6       1              ,A12,A13,A14,A15,A16,A17,A18,A19,A20,IOPE             00001300
   7        COMMON /LOA/ NLOAD,NPTM                                              00001400
   7        COMMON /JUNK/ HED(12),MTOT                                           00001500
   7        COMMON /VAR/ NG,KPRI,MODEX,KSTEP,ITE,ITEMAX,IREF,IEQREF,INOCMD       00001600
   7        COMMON /TAPES/ NREAD,NWRITE                                          00001700
   7        COMMON /NORMS/ RNORM,RENORM,RTOL                                     00001800
   7        COMMON /PRCON/ NPB,IDC,IVC,IAC,IPC,IPNODE(2,8)                       00001900
   7        COMMON /DAMPNG/ ADAMP,BDAMP                                          00002000
   7        COMMON /NSAPI/ OPVAR(7),TSTART,IEQUIT,NUMREF,IPRI,ISREF              00002100
   7        COMMON A(1)                                                          00002200
   1  C                                                                         00002300
   7        DIMENSION IDOF(6)                                                    00002400
   1  C                                                                         00002500
   1  C                                                                         00002600
   1  C     R E A D   C O N T R O L   I N F O R M A T I O N                      00002700
   1  C                                                                         00002800
   1  C                                                                         00002900
   7        READ (KIO5,1000,END=100)                                             00003000
   6       *             HED,NUMNP,(IDOF(I),I=1,6),NEGL,NEGNL,MODEX,NSTE,      00003100
   6       1             DT,TSTART,IPRI                                       00003200
   7        IF (NUMNP.EQ.0) STOP                                                 00003300
   7        READ(KIO5,1001)                                                      00003400
   6       *             IMASS,IDAMP,IMASSN,IDAMPN                              00003500
   7        READ(KIO5,1002) IEIG                                                 00003600
   7        READ(KIO5,1005)                                                      00003700
   6       *             ISREF,NUMREF,IEQUIT,ITEMAX,RTOL                        00003800
   7        READ(KIO5,1020) IOPE,OPVAR                                           00003900
   7        READ(KIO5,1010) NPB,IDC,IVC,IAC                                      00004000
   7        READ(KIO5,1010) IPNODE                                               00004100
   1  C                                                                         00004200
   1  C     VERIFY AND INITIALIZE SOLUTION VARIABLES                             00004300
   1  C                                                                         00004400
   7        NUMEG=NEGL + NEGNL                                                   00004500
   7        NDOF=6                                                               00004600
   7        DO 1 I=1,6                                                           00004700
   5      1 NDOF=NDOF - IDOF(I)                                                  00004800
   7        IF (IPRI.EQ.0) IPRI=1                                                00004900
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7        ISTAT=1                                                                00005000
   7        IF (IMASS.EQ.0) ISTAT=0                                                00005100
   7        IF (IDAMP.EQ.1 .AND. IMASS.EQ.0) GO TO 5                               00005200
   7        IF (IDAMPN.NE.0 .AND. IMASS.EQ.0) GO TO 5                              00005300
   7        IF (IMASSN.NE.0 .AND. IMASS.EQ.0) GO TO 5                              00005400
   7        GO TO 3                                                                00005500
   5      5 WRITE(KIO6,3002)                                                       00005600
   7        STOP                                                                   00005700
   5      3 TH=1.0                                                                 00005800
   7        DTA=DT                                                                 00005900
   7        KLIN=1                                                                 00006000
   7        IF (NEGNL.EQ.0) KLIN=0                                                 00006100
   7        IF (IEIG.LE.1) GO TO 4                                                 00006200
   7        WRITE(KIO6,3003)                                                       00006300
   7        STOP                                                                   00006400
   5      4 IF (ISREF.EQ.0) ISREF=1                                                00006500
   7        IF (KLIN.EQ.0) ISREF=NSTE + 1                                          00006600
   7        IF (IEQUIT.EQ.0) IEQUIT=1                                              00006700
   7        IF (KLIN.EQ.0) IEQUIT=NSTE + 1                                         00006800
   7        IF (ITEMAX.EQ.0) ITEMAX=15                                             00006900
   7        IF (RTOL.EQ.0) RTOL=1.0E-03                                            00007000
   7        NUMREF=0                                                               00007100
   7        IPC=IDC+IVC+IAC                                                        00007200
   7        IF (NPB.NE.0) GO TO 6                                                  00007300
   7        IPC=1                                                                  00007400
   7        NPB=1                                                                  00007500
   7        IDC=1                                                                  00007600
   7        IVC=1                                                                  00007700
   7        IAC=1                                                                  00007800
   7        IPNODE(1,1)=1                                                          00007900
   7        IPNODE(2,1)=NUMNP                                                      00008000
   5      6 CONTINUE                                                               00008100
   7        IF (IOPE.LE.0) IOPE=1                                                  00008200
   1  C                                                                            00008300
   7        WRITE(KIO6,2000)                                                       00008400
   6       *               HED,NUMNP,(IDOF(I),I=1,6),NEGL,NEGNL,MODEX              00008500
   7        WRITE(KIO6,2002)                                                       00008600
   6       *               NSTE,DT,TSTART,IPRI                                     00008700
   7        WRITE(KIO6,2008)                                                       00008800
   6       *               IMASS,IDAMP,IMASSN,IDAMPN                               00008900
   7        WRITE(KIO6,2001) IEIG                                                  00009000
   7        WRITE(KIO6,2003)                                                       00009100
   6       *               ISREF,NUMREF,IEQUIT,ITEMAX,RTOL                         00009200
   7        WRITE(KIO6,2004) ISTAT,KLIN                                            00009300
   1  C                                                                            00009400
   7        INOCMD=1                                                               00009500
   7        IF (IDAMP.EQ.0 .AND. IMASS.NE.2) INOCMD=0                              00009600
   7        IF (IDAMPN.NE.0) INOCMD=1                                              00009700
   7        RTOL=RTOL * RTOL                                                       00009800
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8
    7         IF (ISTAT.EQ.0) GO TO 9                                          00009900
                                                                               00010000
    1  C
    1  C      SET TIME INTEGRATION COEFFICIENTS IN CASE OF DYNAMIC PROBLEM      00010100
                                                                               00010200
    1  C
    7         CALL OPCOEF (OPVAR)                                              00010300
                                                                               00010400
    1  C
    7         WRITE(KIO6,2005) IOPE                                            00010500
    7         IF (IOPE.EQ.1) TH=OPVAR(1)                                       00010600
    7         IF (IOPE.EQ.1)                                                   00010700
    6        *WRITE(KIO6,2006) OPVAR(1)                                        00010800
    7         IF (IOPE.EQ.2) WRITE(KIO6,2007) OPVAR(1),OPVAR(2)                00010900
                                                                               00011000
    1  C
    1  C      PRINT DISPL/VEL/ACC OUTPUT INFORMATION                           00011100
                                                                               00011200
    1  C
    5       9 WRITE(KIO6,2012)                                                 00011300
    6        *               NPB,IDC,IVC,IAC                                   00011400
    7         DO 7 L=1,NPB                                                     00011500
    5       7 WRITE(KIO6,2013)                                                 00011600
    6        *               L,L,IPNODE(1,L),L,IPNODE(2,L)                     00011700
                                                                               00011800
    1  C
                                                                               00011900
    1  C      R E A D   N O D A L   P O I N T   D A T A                        00012000
                                                                               00012100
    1  C
                                                                               00012200
    1  C
    7         N1= 1 + NUMEST                                                   00012300
    7         N2=N1 + NDOF*NUMNP                                               00012400
    7         N3=N2 + NUMNP                                                    00012500
    7         N4=N3 + NUMNP                                                    00012600
    7         N5=N4 + NUMNP                                                    00012700
    7         IF (N5.GT.MTOT) CALL ERROR (N5-MTOT)                             00012800
                                                                               00012900
    1  C
    7         CALL INPUT (A(N1),A(N2),A(N3),A(N4),NUMNP,NDOF,NEQ)              00013000
                                                                               00013100
    1  C
                                                                               00013200
    1  C      C A L C U L A T E   A N D   S T O R E   L O A D   V E C T O R S  00013300
                                                                               00013400
    1  C
                                                                               00013500
    1  C
    7         READ(KIO5,1010) NLOAD,NLCUR,NPTM                                 00013600
    7         WRITE(KIO6,2010) NLOAD,NLCUR,NPTM                                00013700
                                                                               00013800
    1  C
    7         N6 =N5  + NEQ                                                    00013900
    7         N7 =N6  + NPTM                                                   00014000
    7         N8 =N7  + NPTM                                                   00014100
    7         N9 =N8  + NLCUR*NSTE                                             00014200
    7         N10=N9  + NLOAD                                                  00014300
    7         N11=N10 + NLOAD                                                  00014400
    7         N12=N11 + NLOAD                                                  00014500
    7         N13=N12 + NLOAD                                                  00014600
    7         IF (N13.GT.MTOT) CALL ERROR (N13-MTOT)                           00014700
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8
```

```
   1  C                                                                          00014800
   7        CALL LOADS (A(N5),A(N6),A(N7),A(N8),A(N9),A(N10),A(N11),A(N12),     00014900
   6       1           A(N1),NLCUR,NDOF)                                        00015000
   1  C                                                                          00015100
   1  C                                                                          00015200
   1  C     I N P U T   R A Y L E I G H   D A M P I N G                          00015300
   1  C                          C O E F F I C I E N T S                         00015400
   1  C                                                                          00015500
   1  C                                                                          00015600
   7        IF (IDAMP.EQ.0) GO TO 10                                            00015700
   7        READ(KIO5,1030) ADAMP,BDAMP                                         00015800
   7        WRITE(KIO6,2030) ADAMP,BDAMP                                        00015900
   7        IF (ADAMP.NE.0.0 .OR. BDAMP.NE.0.0) GO TO 11                        00016000
   7        WRITE(KIO6,3001)                                                    00016100
   7        STOP                                                                00016200
   4     10 ADAMP=0.0                                                           00016300
   7        BDAMP=0.0                                                           00016400
   1  C                                                                          00016500
   1  C                                                                          00016600
   1  C     E S T A B L I S H  C O N C E N T R A T E D  N O D A L                00016700
   1  C          M A S S  A N D  D A M P I N G  V E C T O R S                    00016800
   1  C                                                                          00016900
   1  C                                                                          00017000
   4     11 CALL NODMAS  (A(N1),A(N5),NDOF)                                     00017100
   1  C                                                                          00017200
   1  C                                                                          00017300
   1  C     F I N D  I N I T I A L  C O N D I T I O N S                          00017400
   1  C                                                                          00017500
   1  C                                                                          00017600
   7        N6=N5 + NEQ                                                         00017700
   7        N7=N6 + NEQ                                                         00017800
   7        N8=N7 + NEQ                                                         00017900
   7        IF (ISTAT.EQ.0) N8=N6                                               00018000
   7        IF (N8.GT.MTOT) CALL ERROR (N8 - MTOT)                              00018100
   7        CALL INITAL (A(N5),A(N6),A(N7),A(N1),NDOF)                          00018200
   1  C                                                                          00018300
   7        GO TO 102                                                           00018400
   3    100 WRITE(KIO6,103)                                                     00018500
   3    103 FORMAT(' END OF DATA')                                              00018600
   7        STOP                                                                00018700
   3    102 RETURN                                                              00018800
   1  C                                                                          00018900
   1  C                                                                          00019000
   2   1000 FORMAT (12A6/I5,6I1,I4,3I5,2F10.0,I5)                               00019100
   2   1001 FORMAT (4I5)                                                        00019200
   2   1002 FORMAT (I5)                                                         00019300
   2   1005 FORMAT (4I5,E10.4)                                                  00019400
   2   1010 FORMAT (16I5)                                                       00019500
   2   1020 FORMAT (I10,7F10.0)                                                 00019600
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   2   1030 FORMAT (2F10.0)                                                    00019700
   1  C                                                                        00019800
   2   2000 FORMAT (1H1,12A6 ///                                              00019900
   6       155H C O N T R O L  I N F O R M A T I O N                    //5X,00020000
   6       255HNUMBER OF NODAL POINTS . . . . . . . . . . (NUMNP)   =,I5//5X,00020100
   6       355HMASTER X-TRANSLATION CODE . . . . . . . . (IDOF(1)) =,I5//5X,00020200
   6       355HMASTER Y-TRANSLATION CODE . . . . . . . . (IDOF(2)) =,I5//5X,00020300
   6       355HMASTER Z-TRANSLATION CODE . . . . . . . . (IDOF(3)) =,I5//5X,00020400
   6       355HMASTER X-ROTATION    CODE . . . . . . . . (IDOF(4)) =,I5//5X,00020500
   6       355HMASTER Y-ROTATION    CODE . . . . . . . . (IDOF(5)) =,I5//5X,00020600
   6       355HMASTER Z-ROTATION    CODE . . . . . . . . (IDOF(6)) =,I5//5X,00020700
   6       455HNUMBER OF LINEAR ELEMENT GROUPS . . . . . (NEGL)    =,I5//5X,00020800
   6       555HNUMBER OF NONLINEAR ELEMENT GROUPS  . . . (NEGNL)   =,I5//5X,00020900
   6       D55HSOLUTION MODE . . . . . . . . . . . . . . (MODEX)   =,I5 /5X,00021000
   6       E55H   EQ.0, DATA CHECK                                      /5X,00021100
   6       F55H   EQ.1, EXECUTION                                       /5X,00021200
   6       G55H   EQ.2, RESTART                                         )  00021300
   2   2001 FORMAT (/5X,                                                      00021400
   6       H55HFREQUENCIES SOLUTION CODE . . . . . . . . (IEIG)    =,I5 /5X,00021500
   6       I55H   EQ.0, NO FREQUENCIES SOLUTION                         /5X,00021600
   6       J55H   EQ.1, FREQUENCIES AND MODE SHAPES                     /5X,00021700
   6       K55H          ARE DETERMINED                                  )  00021800
   2   2002 FORMAT (/5X,                                                      00021900
   6       Q55HNUMBER OF TIME STEPS  . . . . . . . . . . (NSTE)    =,I5//5X,00022000
   6       R51HTIME STEP INCREMENT . . . . . . . . . . . (DT)  =,E11.4 //5X,00022100
   6       S51HTIME AT SOLUTION START  . . . . . . . . . (TSTART)=, E11.4//5X,00022200
   6       S55HPRINTING INTERVAL . . . . . . . . . . . . (IPRI)    =,I5 )  00022300
   2   2003 FORMAT (/5X,                                                      00022400
   6       T55HNUMBER OF TIME STEPS BETWEEN REFORMING                    /5X,00022500
   6       U55H  EFFECTIVE STIFFNESS MATRIX  . . . . . . (ISREF)   =,I5//5X,00022600
   6       A55HNUMBER OF ALLOWABLE STIFFNESS REFORMATIONS                /5X,00022700
   6       B55H  IN EACH TIME STEP . . . . . . . . . . . (NUMREF)  =,I5//5X,00022800
   6       V55HNUMBER OF TIME STEPS BETWEEN                              /5X,00022900
   6       W55H  EQUILIBRIUM ITERATIONS  . . . . . . . . (IEQUIT)  =,I5//5X,00023000
   6       X55HMAXIMUM NUMBER OF EQUILIBRIUM                             /5X,00023100
   6       Y55H  ITERATIONS PERMITTED  . . . . . . . . . (ITEMAX)  =,I5//5X,00023200
   6       Z51HCONVERGENCE TOLERANCE . . . . . . . . . . (RTOL)=,E9.2   )  00023300
   2   2004 FORMAT (//                                                        00023400
   6       A55H A N A L Y S I S  T Y P E                                //5X,00023500
   6       B55HTIME DEPENDENCY CODE  . . . . . . . . . . (ISTAT)   =,I5 /5X,00023600
   6       C55H   EQ.0, STATIC ANALYSIS                                 /5X,00023700
   6       D55H   EQ.1, DYNAMIC ANALYSIS                               //5X,00023800
   6       E55HNONLINEARITY CODE . . . . . . . . . . . . (KLIN)    =,I5 /5X,00023900
   6       F55H   EQ.0, LINEAR ANALYSIS                                 /5X,00024000
   6       G55H   EQ.1, NONLINEAR ANALYSIS                              )  00024100
   2   2005 FORMAT (/5X,                                                      00024200
   6       155HTIME INTEGRATION CODE . . . . . . . . . . (IOPE)    =,I5 /5X,00024300
   6       255H   EQ.1, WILSONS THETA METHOD                            ,/,5X,  00024400
   6       355H   EQ.2, NEWMARKS METHOD                                 ,/,5X)  00024500
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   2   2006 FORMAT (5X,                                                          00024600
   6       155HINTEGRATION PARAMETER . . . . . (THETA)   =,F5.2)  00024700
   2   2007 FORMAT (5X,                                                          00024800
   6       155HINTEGRATION PARAMETERS  . . . . . (DELTA)   =,F5.2/5X00024900
   6       255H                                  (ALPHA)   =,F5.2)  00025000
   2   2012 FORMAT (//                                                           00025100
   6       A55H D I S P / V E L / A C C   P R I N T O U T   C O D E      //5X,00025200
   6       B55HNUMBER OF BLOCKS OF NODAL PRINTOUT  . . . . . (NPB) =,I5//5X,00025300
   6       C55HDISPLACEMENT PRINTOUT CODE  . . . . . (IDC) =,I5 /5X,00025400
   6       D55H   EQ.0, NO PRINTING OF DISPLACEMENTS                  /5X,00025500
   6       E55H   EQ.1, PRINT DISPLACEMENTS                          //5X,00025600
   6       F55HVELOCITY PRINTOUT CODE  . . . . . (IVC) =,I5 /5X,00025700
   6       G55H   EQ.0, NO PRINTING OF VELOCITIES                     /5X,00025800
   6       H55H   EQ.1, PRINT VELOCITIES                             //5X,00025900
   6       I55HACCELERATION PRINTOUT CODE  . . . . . (IAC) =,I5 /5X,00026000
   6       J55H   EQ.0, NO PRINTING OF ACCELERATIONS                  /5X,00026100
   6       K55H   EQ.1, PRINT ACCELERATIONS                           )  00026200
   2   2013 FORMAT (/5X,                                                         00026300
   6       A6HBLOCK ,I2                                              //8X,00026400
   6       B48HFIRST NODE OF THIS BLOCK  . . . . .  (IPNODE(1,I1,3H) =I5//8X,00026500
   6       C48HLAST NODE OF THIS BLOCK  . . . . . (IPNODE(2,I1,3H) =I5)    00026600
   2   2008 FORMAT (/5X,                                                         00026700
   6       155HMASS MATRIX CODE  . . . . . (IMASS)   =,I5 /5X,00026800
   6       255H   EQ.0, NO MASS EFFECTS                               /5X,00026900
   6       355H   EQ.1, LUMPED MASS                                   /5X,00027000
   6       455H   EQ.2, CONSISTENT MASS                              //5X,00027100
   6       555HDAMPING MATRIX CODE . . . . . (IDAMP)   =,I5 /5X,00027200
   6       655H   EQ.0, NO RAYLEIGH DAMPING                           /5X,00027300
   6       755H   EQ.1, RAYLEIGH DAMPING                             //5X,00027400
   6       855HNUMBER OF NODAL MASSES  . . . . . (IMASSN)  =,I5//5X,00027500
   6       955HNUMBER OF NODAL DAMPERS . . . . . (IDAMPN)  =,I5  )  00027600
   2   2010 FORMAT (1H1,12H L O A D S   // 4X,                                   00027700
   6       1  38H NUMBER OF LOADS                    = ,I5 / 4X,               00027800
   6       2  38H NUMBER OF LOAD CURVES             = ,I5 / 4X,               00027900
   6       3  38H MAX NUMBER OF POINTS IN LOAD CURVES = ,I5 /   )              00028000
   2   2030 FORMAT (1H1,51H R A Y L E I G H   D A M P I N G   C O E F F I C I 00028100
   6       1        10HE N T S   ///                                          00028200
   6       2        40H     ALPHA . . . . . (ADAMP)   =,E13.4/               00028300
   6       3        40H     BETA  . . . . . (BDAMP)   =,E13.4//)             00028400
   2   3001 FORMAT (///51H **STOP - NO RAYLEIGH DAMPING COEFFICIENTS INPUT **)00028500
   2   3002 FORMAT (///45H **STOP - IF RAYLEIGH DAMPING OR CONCENTRATED        00028600
   6      1        /43HMASSES/DAMPERS, THEN IMASS CANNOT EQUAL 0     )       00028700
   2   3003 FORMAT (///42H **STOP - IEIG.GT.1 NOT PERMITTED IN THIS            00028800
   6       1        19HVERSION OF NONSAP**   )                               00028900
   1   C                                                                      00029000
   7         END                                                              00029100
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7          SUBROUTINE OPCOEF (OPVAR)                                        00000100
   7          IMPLICIT REAL*8(A-H,O-Z)                                         00000200
   1  C                                                                        00000300
   1  C       CALCULATES COEFFICIENTS OF TIME INTEGRATION OPERATORS            00000400
   1  C       FOR THE FOLLOWING METHODS                                        00000500
   1  C       IOPE = 1 .....WILSONS THETA METHOD                               00000600
   1  C       IOPE = 2 .....NEWMARK METHOD                                     00000700
   1  C                                                                        00000800
   7          COMMON /CONST/ DT,DTA,A0,A1,A2,A3,A4,A5,A6,A7,A8,A9,A10,A11      00000900
   6         1              ,A12,A13,A14,A15,A16,A17,A18,A19,A20,IOPE         00001000
   7          DIMENSION OPVAR(1)                                               00001100
   1  C                                                                        00001200
   7          DATA IOUT/6/                                                     00001300
   7          DATA IOPMAX/2/                                                   00001400
   7          DATA ZERO/1.E-9/                                                 00001500
   7          DATA THMN/1.39/,THMX/2.01/                                       00001600
   1  C                                                                        00001700
   1  C                                                                        00001800
   7          IF (IOPE.LE.IOPMAX) GO TO 99                                     00001900
   7          WRITE (IOUT,1001)                                                00002000
   7          STOP                                                             00002100
   4       99 CONTINUE                                                         00002200
   7          IF (DT.GT.ZERO) GO TO 100                                        00002300
   7          WRITE (IOUT,1003)                                                00002400
   7          STOP                                                             00002500
   3      100 CONTINUE                                                         00002600
   1  C                                                                        00002700
   7          GO TO (101, 111, 121, 131) ,IOPE                                 00002800
   1  C                                                                        00002900
   1  C       W I L S O N S   T H E T A   M E T H O D                          00003000
   1  C                                                                        00003100
   3      101 IF (OPVAR(1).LE.0.) OPVAR(1)=1.4                                 00003200
   7          TH=OPVAR(1)                                                      00003300
   7          IF ((TH.GT.THMN).AND.(TH.LT.THMX)) GO TO 102                     00003400
   7          WRITE (IOUT,1002)                                                00003500
   7          STOP                                                             00003600
   3      102 CONTINUE                                                         00003700
   7          DTA=DT*TH                                                        00003800
   7          THSQ=TH*TH                                                       00003900
   7          A0 = 6./(DTA*DTA)                                                00004000
   7          A1 = 3./DTA                                                      00004100
   7          A2 = 2.*A1                                                       00004200
   7          A3 = 2.                                                          00004300
   7          A4 = 2.                                                          00004400
   7          A5 = .5*DTA                                                      00004500
   7          A6 = A0/TH                                                       00004600
   7          A7 =-A2/TH                                                       00004700
   7          A8 = 1.-3./TH                                                    00004800
   7          A9 = .5*DT                                                       00004900
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7        A10= DT*DT/6.                                                    00005000
   7        GO TO 999                                                        00005100
   3    111 CONTINUE                                                         00005200
   1  C                                                                      00005300
   1  C     N E W M A R K S   M E T H O D                                    00005400
   1  C                                                                      00005500
   7        IF (OPVAR(1).LE.0.0) OPVAR(1)=0.5                                00005600
   7        IF (OPVAR(2).LE.0.0) OPVAR(2)=0.25*((OPVAR(1) + 0.50)**2)        00005700
   7        DELT=OPVAR(1)                                                    00005800
   7        ALFA=OPVAR(2)                                                    00005900
   7        IF (DELT.LT.0.5) GO TO 113                                       00006000
   7        IF (DELT.GT.0.55) GO TO 114                                      00006100
   7        D1=0.5*(DELT+0.5)                                                00006200
   7        D2=D1*D1                                                         00006300
   7        IF (ALFA.LE.ZERO) GO TO 113                                      00006400
   7        IF (ALFA.LT.D2) GO TO 115                                        00006500
   7        IF (ALFA.GT.D1) GO TO 114                                        00006600
   7        GO TO 116                                                        00006700
   3    113 CONTINUE                                                         00006800
   7        WRITE (IOUT,1002)                                                00006900
   7        STOP                                                             00007000
   3    114 CONTINUE                                                         00007100
   7        WRITE (IOUT,1004)                                                00007200
   7        GO TO 116                                                        00007300
   3    115 CONTINUE                                                         00007400
   7        WRITE (IOUT,1005)                                                00007500
   3    116 CONTINUE                                                         00007600
   7        DEAL=DELT/ALFA                                                   00007700
   7        A0 = 1./(ALFA*DT*DT)                                             00007800
   7        A1 = DEAL/DT                                                     00007900
   7        A2 = 1./(ALFA*DT)                                                00008000
   7        A3 = .5/ALFA-1.                                                  00008100
   7        A4 = DEAL-1.                                                     00008200
   7        A5 = DT*(.5*DEAL-1.)                                             00008300
   7        A6 = A0                                                          00008400
   7        A7 =-A2                                                          00008500
   7        A8 =-A3                                                          00008600
   7        A9 = DT*(1. - DELT)                                              00008700
   7        A10= DELT*DT                                                     00008800
   7        GO TO 999                                                        00008900
   3    121 CONTINUE                                                         00009000
   3    131 CONTINUE                                                         00009100
   3    999 CONTINUE                                                         00009200
   7        RETURN                                                           00009300
   1  C                                                                      00009400
   2   1001 FORMAT(//,10X,42HINTEGRATION METHOD INDICATOR OUT OF RANGE )     00009500
   2   1002 FORMAT(//,10X,41HTIME INTEGRATION PARAMETER NOT ADMISSIBLE)      00009600
   2   1003 FORMAT(//,10X,19HTIME STEP TOO SMALL)                            00009700
   2   1004 FORMAT(//,10X,45HTIME INTEGRATION PARAMETER SUSPICIOUS        )  00009800
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   2   1005 FORMAT(//,10X,45HCONDITIONALLY STABLE ALGORITHM               )  00009900
   1  C                                                                      00010000
   7        END                                                              00010100
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7         SUBROUTINE PNORM (S,MAXA,NEQ,ANORM)                          00000100
   7         IMPLICIT REAL*8(A-H,O-Z)                                     00000200
   1  C                                                                   00000300
   1  C      FINDS A NORM OF THE MATRIX S                                 00000400
   1  C                                                                   00000500
   7         DIMENSION S(1),MAXA(1)                                       00000600
   1  C                                                                   00000700
   7         SUM=0.                                                       00000800
   7         DO 10 I=1,NEQ                                                00000900
   7         II=MAXA(I)                                                   00001000
   4     10  SUM=SUM + S(II)                                             00001100
   7         ANORM=(SUM/NEQ)*1.E-08                                       00001200
   7         RETURN                                                       00001300
   7         END                                                          00001400
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7          SUBROUTINE POSINV (A,NMAX,NDD)                                  00000100
   7          IMPLICIT REAL*8(A-H,O-Z)                                        00000200
   1  C                                                                       00000300
   7          DIMENSION A(NDD,NDD)                                            00000400
   1  C                                                                       00000500
   7          DO 200 N=1,NMAX                                                 00000600
   1  C                                                                       00000700
   7          D=A(N,N)                                                        00000800
   7          DO 100 J=1,NMAX                                                 00000900
   3    100   A(N,J)=-A(N,J)/D                                                00001000
   1  C                                                                       00001100
   7          DO 150 I=1,NMAX                                                 00001200
   7          IF(N-I) 110,150,110                                             00001300
   3    110   DO 140 J=1,NMAX                                                 00001400
   7          IF(N-J) 120,140,120                                             00001500
   3    120   A(I,J)=A(I,J)+A(I,N)*A(N,J)                                     00001600
   3    140   CONTINUE                                                        00001700
   3    150   A(I,N)=A(I,N)/D                                                 00001800
   1  C                                                                       00001900
   7          A(N,N)=1.0/D                                                    00002000
   1  C                                                                       00002100
   3    200   CONTINUE                                                        00002200
   1  C                                                                       00002300
   7          RETURN                                                          00002400
   7          END                                                             00002500
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7        SUBROUTINE PRAGER (TAU,DEPS,DP)                                        00000100
   7        IMPLICIT REAL*8(A-H,O-Z)                                               00000200
   1  C                                                                            00000300
   1  C        FORMS THE ELASTO-PLASTIC MATERIAL MATRIX                            00000400
   1  C        (FOR THE DRUCKER-PRAGER YIELD CRITERION)                            00000500
   1  C                                                                            00000600
   7        COMMON /EL    / IND,ICOUNT,NPAR(20),NUMEG,NEGL,NEGNL,IMASS,IDAMP,     00000700
   6       1                ISTAT,NDOF,KLIN,IEIG,IMASSN,IDAMPN                    00000800
   7        COMMON /MATMOD/  STRESS(4),STRAIN(4),C(4,4),IPT,NEL                    00000900
   7        COMMON /DRPRAG/ A1,B1,C1,A2,B2,C2,D2,G,BM,ALFA,YLD,ISR,IST             00001000
   1  C                                                                            00001100
   7        EQUIVALENCE (NPAR(5),ITYP2D)                                           00001200
   7        DIMENSION TAU(1),DEPS(1),DP(1)                                         00001300
   1  C                                                                            00001400
   1  C                                                                            00001500
   7        SM = (TAU(1)+TAU(2)+TAU(4))/3.                                         00001600
   7        SX = TAU(1) - SM                                                       00001700
   7        SY = TAU(2) - SM                                                       00001800
   7        SS = TAU(3)                                                            00001900
   7        SZ = TAU(4) - SM                                                       00002000
   7        SBAR=DSQRT( .5*(SX*SX+SY*SY+SZ*SZ) +SS*SS )                            00002100
   7        QDQ=DSQRT (G +9.*BM*ALFA*ALFA)                                         00002200
   7        AA=G/SBAR/QDQ                                                          00002300
   7        BB=3.*BM*ALFA/QDQ                                                      00002400
   1  C                                                                            00002500
   7        IF (ITYP2D - 1) 10,11,12                                               00002600
   4     10 DLAMDA= BB * (DEPS(1)+DEPS(2)+DEPS(4)) +                               00002700
   6       1        AA * (SX*DEPS(1)+SY*DEPS(2)+SS*DEPS(3)+SZ*DEPS(4))            00002800
   7        GO TO 15                                                               00002900
   1  C                                                                            00003000
   4     11 DLAMDA= BB* (DEPS(1)+DEPS(2)) +                                        00003100
   6       1        AA * (SX*DEPS(1)+SY*DEPS(2)+SS*DEPS(3))                       00003200
   7        GO TO 15                                                               00003300
   1  C                                                                            00003400
   4     12 SA= AA*SX + BB                                                         00003500
   7        SB= AA*SY + BB                                                         00003600
   7        SC= AA*SS                                                              00003700
   7        SD= AA*SZ + BB                                                         00003800
   1  C                                                                            00003900
   7        DP1= B2 - SA*SD                                                        00004000
   7        DP2= B2 - SB*SD                                                        00004100
   7        DP3=    - SC*SD                                                        00004200
   7        DP4= A2 - SD*SD                                                        00004300
   1  C                                                                            00004400
   7        DEPS(4)=(-DP1*DEPS(1)-DP2*DEPS(2)-DP3*DEPS(3))/DP4                     00004500
   1  C                                                                            00004600
   7        DLAMDA= BB * (DEPS(1)+DEPS(2)+DEPS(4)) +                               00004700
   6       1        AA * (SX*DEPS(1)+SY*DEPS(2)+SS*DEPS(3)+SZ*DEPS(4))            00004800
   1  C                                                                            00004900
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7        IF (DLAMDA.LT.0.) GO TO 16                                      00005000
   7        SX=SA                                                           00005100
   7        SY=SB                                                           00005200
   7        SS=SC                                                           00005300
   7        SZ=SD                                                           00005400
   7        GO TO 25                                                        00005500
   1  C                                                                     00005600
   4     15 IF (DLAMDA.GT.0.) GO TO 20                                      00005700
   4     16 AA=0.                                                           00005800
   7        BB=0.                                                           00005900
   7        IF (ITYP2D.EQ.2)  DEPS(4)=D2*(DEPS(1)+DEPS(2))                  00006000
   1  C                                                                     00006100
   4     20 SX= AA*SX + BB                                                  00006200
   7        SY= AA*SY + BB                                                  00006300
   7        SS= AA*SS                                                       00006400
   7        SZ= AA*SZ + BB                                                  00006500
   1  C                                                                     00006600
   4     25 DP( 1)= A2 - SX*SX                                              00006700
   7        DP( 2)= B2 - SX*SY                                              00006800
   7        DP( 3)=    - SX*SS                                              00006900
   7        DP( 4)= B2 - SX*SZ                                              00007000
   7        DP( 5)= DP(2)                                                   00007100
   7        DP( 6)= A2 - SY*SY                                              00007200
   7        DP( 7)=    - SY*SS                                              00007300
   7        DP( 8)= B2 - SY*SZ                                              00007400
   7        DP( 9)= DP(3)                                                   00007500
   7        DP(10)= DP(7)                                                   00007600
   7        DP(11)= C2 - SS*SS                                              00007700
   7        DP(12)=    - SS*SZ                                              00007800
   1  C                                                                     00007900
   7        IF (ITYP2D.EQ.1) RETURN                                         00008000
   1  C                                                                     00008100
   7        DP(13)= DP(4)                                                   00008200
   7        DP(14)= DP(8)                                                   00008300
   7        DP(15)= DP(12)                                                  00008400
   7        DP(16)= A2 - SZ*SZ                                              00008500
   1  C                                                                     00008600
   7        IF (ITYP2D.EQ.0) RETURN                                         00008700
   1  C                                                                     00008800
   1  C     PLANE STRESS / MODIFY DP MATRIX                                 00008900
   1  C                                                                     00009000
   7        DO 120 I=1,3                                                    00009100
   7        A=C(I,4)/C(4,4)                                                 00009200
   7        DO 120 J=I,3                                                    00009300
   7        C(I,J)=C(I,J) - C(4,J)*A                                        00009400
   3    120 C(J,I) = C(I,J)                                                 00009500
   7        STRAIN(4)=STRAIN(4) + DEPS(4)                                   00009600
   1  C                                                                     00009700
   1  C                                                                     00009800
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7        RETURN                                                        00009900
   7        END                                                           00010000
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7        SUBROUTINE QUADM (N,ND,XM,CM,XX,NOD5)                                  00000100
   7        IMPLICIT REAL*8(A-H,O-Z)                                               00000200
   1  C                                                                            00000300
   1  C                                                                            00000400
   1  C       SUBROUTINE TO CALCULATE THE MASS MATRIX OF                           00000500
   1  C           A QUADRILATERAL ELEMENT.                                         00000600
   1  C                                                                            00000700
   1  C                                                                            00000800
   7        COMMON /EL/ IND,ICOUNT,NPAR(20),NUMEG,NEGL,NEGNL,IMASS,IDAMP,ISTAT   00000900
   6       1           ,NDOF,KLIN,IEIG,IMASSN,IDAMPN                            00001000
   7        COMMON /TODIM/ BET,THIC,DE,IEL,NND5                                  00001100
   7        DIMENSION CM(1),XM(8),D(16),XG(4,4),WGT(4,4),XX(2,8),NOD5(1)         00001200
   7        DIMENSION H(8),P(2,8),IPERM(4),XJ(2,2)                               00001300
   1  C                                                                            00001400
   7        EQUIVALENCE (NPAR(5),ITYP2D)                                         00001500
   1  C                                                                            00001600
   7        DATA XG /     0.,              0.,              0.,              0., 00001700
   6       1 -.5773502691896, .5773502691896,              0.,              0., 00001800
   6       2 -.7745966692415, .0000000000000, .7745966692415,              0., 00001900
   6       3 -.8611363115941,-.3399810435849, .3399810435849, .8611363115941/ 00002000
   7        DATA WGT / 2.000,              0.,              0.,              0., 00002100
   6       1 1.0000000000000,1.0000000000000,              0.,              0., 00002200
   6       2  .5555555555556, .8888888888889, .5555555555556,              0., 00002300
   6       3  .3478548451375, .6521451548625, .6521451548625, .3478548451375/ 00002400
   7        DATA IPERM/2,3,4,1/                                                  00002500
   1  C                                                                            00002600
   1  C       INTEGRATE --- CONSISTENT OR LUMPED MASS MATRIX                       00002700
   1  C                                                                            00002800
   7        IF (IMASS.EQ.1) GO TO 9                                              00002900
   7        DO 8 I=1,136                                                         00003000
   5       8 CM(I)=0.                                                            00003100
   5       9 DO 7 I=1,ND                                                         00003200
   5       7 XM(I)=0.                                                            00003300
   1  C                                                                            00003400
   7        DO 100 LX=1,3                                                        00003500
   7        R=XG(LX,3)                                                           00003600
   7        DO 100 LY=1,3                                                        00003700
   7        S=XG(LY,3)                                                           00003800
   7        WT=WGT(LX,3)*WGT(LY,3)                                               00003900
   1  C                                                                            00004000
   1  C                                                                            00004100
   1  C     FIND INTERPOLATION FUNCTIONS AND JACOBIAN                              00004200
   1  C                                                                            00004300
   7        CALL FUNCT2 (R,S,H,P,NOD5,XJ,DET,XX,NEL)                             00004400
   1  C                                                                            00004500
   1  C     COMPUTE THE RADIUS AT POINT (R,S)                                      00004600
   1  C                                                                            00004700
   7        IF (ITYP2D.EQ.0) GO TO 40                                            00004800
   7        IF (ITYP2D.GT.0) XBAR=THIC                                           00004900
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7         GO TO 60                                                      00005000
   4      40 XBAR=0.0                                                      00005100
   7         DO 50 K=1,IEL                                                 00005200
   4      50 XBAR=XBAR + H(K)*XX(1,K)                                      00005300
   1  C                                                                    00005400
   4      60 FAC=WT*XBAR*DET*DE                                            00005500
   1  C                                                                    00005600
   1  C      CONSISTENT MASS                                               00005700
   1  C                                                                    00005800
   7         IF (IMASS.LT.2) GO TO 320                                     00005900
   7         DO 200 I = 1,IEL                                              00006000
   7         D(2*I-1) = H(I)                                               00006100
   3    200  D(2*I) = H(I)                                                 00006200
   7         KL=1                                                          00006300
   7         DO 300 I=1,ND,2                                               00006400
   7         DO 301 J=I,ND,2                                               00006500
   7         CM(KL)=CM(KL) + D(I)*D(J)*FAC                                 00006600
   3    301  KL=KL + 2                                                    00006700
   3    300  KL=KL + ND - I                                               00006800
   7         GO TO 100                                                     00006900
   1  C                                                                    00007000
   1  C      LUMPED MASS                                                   00007100
   1  C                                                                    00007200
   3    320  DO 325 I=1,ND,2                                              00007300
   7         FACM=FAC/IEL                                                  00007400
   3    325  XM(I)=XM(I) + FACM                                            00007500
   1  C                                                                    00007600
   3    100  CONTINUE                                                     00007700
   1  C                                                                    00007800
   7         IF (IMASS.EQ.1) GO TO 335                                     00007900
   1  C                                                                    00008000
   7         KL=1                                                          00008100
   7         DO 401 I=1,ND,2                                               00008200
   7         KS=KL + ND - I + 1                                            00008300
   7         DO 400 J=I,ND,2                                               00008400
   7         CM(KS)=CM(KL)                                                 00008500
   7         KS=KS + 2                                                     00008600
   3    400  KL=KL + 2                                                    00008700
   3    401  KL=KL + ND - I                                               00008800
   1  C                                                                    00008900
   7         RETURN                                                        00009000
   1  C                                                                    00009100
   1  C                                                                    00009200
   1  C                                                                    00009300
   3    335  DO 340 I=1,ND,2                                              00009400
   3    340  XM(I+1)=XM(I)                                                00009500
   1  C                                                                    00009600
   7         RETURN                                                        00009700
   1  C                                                                    00009800
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   1  C                                                                    00009900
   7        END                                                            00010000
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7        SUBROUTINE QUADM3  (N,ND,NDM2,XM,CM,XX,NOD9)                          00000100
   7        IMPLICIT REAL*8(A-H,O-Z)                                              00000200
   1  C                                                                           00000300
   1  C                                                                           00000400
   1  C . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .  00000500
   1  C .                                                                     .   00000600
   1  C .   P R O G R A M                                                     .   00000700
   1  C .                                                                     .   00000800
   1  C .      EVALUATES MASS MATRIX                                          .   00000900
   1  C .                                                                     .   00001000
   1  C .      CURVILINEAR HEXAHEDRON   8 TO 21 NODES                         .   00001100
   1  C .                                                                     .   00001200
   1  C . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .  00001300
   1  C                                                                           00001400
   1  C                                                                           00001500
   1  C                                                                           00001600
   7        COMMON /EL/ IND,ICOUNT,NPAR(20),NUMEG,NEGL,NEGNL,IMASS,IDAMP,ISTAT00001700
   6       1           ,NDOFDM,KLIN,IEIG,IMASSN,IDAMPN                          00001800
   7        COMMON /THREED/ DE,IELD,IELX,IEL,NPT,IDW,NND9                       00001900
   7        DIMENSION XM(1),CM(1),XX(3,1),D(63),XG(4,4),WGT(4,4),NOD9(1)        00002000
   7        DIMENSION H(21),P(3,21),XJ(3,3)                                     00002100
   1  C                                                                           00002200
   1  C                                                                           00002300
   7        DATA XG /     0.,              0.,              0.,              0., 00002400
   6       1 -.5773502691896, .5773502691896,              0.,              0., 00002500
   6       2 -.7745966692415, .0000000000000, .7745966692415,              0., 00002600
   6       3 -.8611363115941,-.3399810435849, .3399810435849, .8611363115941/ 00002700
   7        DATA WGT / 2.000,              0.,              0.,              0., 00002800
   6       1 1.0000000000000,1.0000000000000,              0.,              0., 00002900
   6       2  .5555555555556, .8888888888889, .5555555555556,              0., 00003000
   6       3  .3478548451375, .6521451548625, .6521451548625, .3478548451375/ 00003100
   1  C                                                                           00003200
   1  C                                                                           00003300
   1  C     INTEGRATE USING GAUSS QUADRATURE                                      00003400
   1  C                                                                           00003500
   1  C                                                                           00003600
   7        NINTM=3                                                             00003700
   7        NINTZM=3                                                            00003800
   7        IF (IMASS.EQ.1) GO TO 9                                             00003900
   7        DO 8 I=1,NDM2                                                       00004000
   5      8 CM(I)=0.0                                                           00004100
   7        GO TO 10                                                            00004200
   5      9 DO 7 I=1,ND                                                         00004300
   5      7 XM(I)=0.                                                            00004400
   1  C                                                                           00004500
   4     10 DO 900 LX=1,NINTM                                                   00004600
   7        R=XG(LX,NINTM)                                                      00004700
   7        DO 900 LY=1,NINTM                                                   00004800
   7        S=XG(LY,NINTM)                                                      00004900
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7        DO 900 LZ=1,NINTZM                                                  00005000
   7        T=XG(LZ,NINTZM)                                                     00005100
   7        WT=WGT(LX,NINTM)*WGT(LY,NINTM)*WGT(LZ,NINTZM)                       00005200
   1  C                                                                         00005300
   1  C                                                                         00005400
   1  C     FIND INTERPOLATION FUNCTIONS AND THEIR DERIVATIVES                  00005500
   1  C     FIND JACOBIAN MATRIX AND ITS DETERMINANT                            00005600
   1  C                                                                         00005700
   1  C                                                                         00005800
   7        CALL FUNCT (R,S,T,H,P,NOD9,XJ,DET,XX,NEL)                           00005900
   1  C                                                                         00006000
   1  C                                                                         00006100
   1  C     CONSISTENT MASS MATRIX                                              00006200
   1  C                                                                         00006300
   1  C                                                                         00006400
   7        FAC=WT*DET*DE                                                       00006500
   7        IF (IMASS.LT.2) GO TO 320                                           00006600
   7        DO 200 I=1,IEL                                                      00006700
   7        D(3*I - 2)=H(I)                                                     00006800
   7        D(3*I - 1)=H(I)                                                     00006900
   3    200 D(3*I)=H(I)                                                         00007000
   7        KL=1                                                                00007100
   7        DO 300 I=1,ND,3                                                     00007200
   7        DO 301 J=I,ND,3                                                     00007300
   7        CM(KL)=CM(KL) + D(I)*D(J)*FAC                                       00007400
   3    301 KL=KL + 3                                                           00007500
   3    300 KL=KL + 2*(ND-I) - 1                                                00007600
   7        GO TO 900                                                           00007700
   1  C                                                                         00007800
   1  C                                                                         00007900
   1  C     LUMPED MASS VECTOR                                                  00008000
   1  C                                                                         00008100
   1  C                                                                         00008200
   3    320 DO 325 I=1,ND,3                                                     00008300
   7        FACM=FAC/IEL                                                        00008400
   3    325 XM(I)=XM(I) + FACM                                                  00008500
   1  C                                                                         00008600
   3    900 CONTINUE                                                            00008700
   1  C                                                                         00008800
   7        IF (IMASS.EQ.1) GO TO 335                                           00008900
   7        KL=1                                                                00009000
   7        DO 450 I=1,ND,3                                                     00009100
   7        KS1=KL + ND - I + 1                                                 00009200
   7        KS2=KS1 + ND - I                                                    00009300
   7        DO 451 J=I,ND,3                                                     00009400
   7        CM(KS1)=CM(KL)                                                      00009500
   7        CM(KS2)=CM(KL)                                                      00009600
   7        KL=KL + 3                                                           00009700
   7        KS1=KS1 + 3                                                         00009800
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   3     451 KS2=KS2 + 3                                                    00009900
   3     450 KL=KL + 2*(ND-I) - 1                                           00010000
   7         RETURN                                                         00010100
   1   C                                                                    00010200
   3     335 DO 340 I=1,ND,3                                                00010300
   7         XM(I+1)=XM(I)                                                  00010400
   3     340 XM(I+2)=XM(I)                                                  00010500
   7         RETURN                                                         00010600
   1   C                                                                    00010700
   1   C                                                                    00010800
   7         END                                                            00010900
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7        SUBROUTINE QUADS (ND,B,S,YZ,PROP,RE,EDIS,IDW,WA,NOD5)                00000100
   1  C                                                                          00000200
   1  C                                                                          00000300
   1  C . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 00000400
   1  C                                                                          00000500
   1  C        ISOPARAMETRIC FORMULATION OF QUADRILATERAL ELEMENT STIFFNESS       00000600
   1  C        FOR AXISYMMETRIC GEOMETRY (PLANE STRESS AND PLANE STRAIN          00000700
   1  C                                                  INCLUDED)               00000800
   1  C                                                                          00000900
   1  C . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 00001000
   1  C                                                                          00001100
   1  C                                                                          00001200
   7        IMPLICIT REAL*8(A-H,O-Z)                                            00001300
   7        COMMON /EL/ IND,ICOUNT,NPAR(20),NUMEG,NEGL,NEGNL,IMASS,IDAMP,ISTAT00001400
   6       1          ,NDOF,KLIN,IEIG,IMASSN,IDAMPN                           00001500
   7        COMMON /VAR/ NG,KPRI,MODEX,KSTEP,ITE,ITEMAX,IREF,IEQREF,INOCMD     00001600
   7        COMMON /TODIM/ BET,THIC,DE,IEL,NND5                                00001700
   7        COMMON /MATMOD/ STRESS(4),STRAIN(4),D(4,4),IPT,NEL                 00001800
   7        COMMON /DISDER/ DISD(5)                                            00001900
   7        DIMENSION B(4,1),S(1),YZ(1),RE(1),EDIS(1),PROP(1),WA(1),NOD5(1)    00002000
   7        DIMENSION XG(4,4),WGT(4,4),DB(4),XX(16),BS(4,16),DI(4,4)           00002100
   1  C                                                                          00002200
   7        EQUIVALENCE (NPAR(10),NINT),(NPAR(5),ITYP2D),(NPAR(3),INDNL)       00002300
   7        EQUIVALENCE (NPAR(15),MODEL)                                       00002400
   1  C                                                                          00002500
   1  C                                                                          00002600
   7        DATA XG /     0.,             0.,             0.,             0.,  00002700
   6       1 -.5773502691896, .5773502691896,             0.,             0.,  00002800
   6       2 -.7745966692415, .0000000000000, .7745966692415,             0.,  00002900
   6       3 -.8611363115941,-.3399810435849, .3399810435849, .8611363115941/  00003000
   7        DATA WGT / 2.000,             0.,             0.,             0.,  00003100
   6       1 1.0000000000000,1.0000000000000,             0.,             0.,  00003200
   6       2  .5555555555556, .8888888888889, .5555555555556,             0.,  00003300
   6       3  .3478548451375, .6521451548625, .6521451548625, .3478548451375/  00003400
   1  C                                                                          00003500
   1  C                                                                          00003600
   7        NPT=NINT*NINT                                                      00003700
   7        IST=4                                                              00003800
   7        IF (ITYP2D.NE.0) IST=3                                             00003900
   7        KST=IST-1                                                          00004000
   1  C                                                                          00004100
   7        IF (IND.GE.4) GO TO 100                                            00004200
   1  C                                                                          00004300
   1  C                                                                          00004400
   1  C     F I N D   S T I F F N E S S   O F   L I N E A R   E L E M E N T      00004500
   1  C                                                                          00004600
   1  C                                                                          00004700
   7        CALL STSTL (NEL,YZ,PROP,D)                                         00004800
   1  C                                                                          00004900
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7        DO 10 LX=1,NINT                                                  00005000
   7        E1=XG(LX,NINT)                                                   00005100
   7        DO 10 LY=1,NINT                                                  00005200
   7        E2=XG(LY,NINT)                                                   00005300
   7        WT=WGT(LX,NINT)*WGT(LY,NINT)                                     00005400
   1  C                                                                      00005500
   1  C     EVALUATE DERIVATIVE OPERATOR AND THE JACOBIAN DETERMINANT        00005600
   1  C                                                                      00005700
   7        CALL DERIQ (NEL,YZ,B,DET,E1,E2,XBAR,NOD5)                        00005800
   1  C                                                                      00005900
   1  C     ADD CONTRIBUTION TO ELEMENT STIFFNESS                            00006000
   1  C                                                                      00006100
   7        IF (IST.EQ.3) XBAR=THIC                                          00006200
   7        FAC=WT*XBAR*DET                                                  00006300
   1  C                                                                      00006400
   7        KL=1                                                             00006500
   7        DO 50 J=1,ND,2                                                   00006600
   7        DO 52 K=1,3                                                      00006700
   7        DB(K)=D(K,1)*B(1,J) + D(K,3)*B(3,J)                              00006800
   4     52 DB(K)=DB(K)*FAC                                                  00006900
   7        DO 51 I=J,ND,2                                                   00007000
   7        S(KL)=S(KL) + B(1,I)*DB(1) + B(3,I)*DB(3)                        00007100
   7        KL=KL + 1                                                        00007200
   7        S(KL)=S(KL) + B(2,I+1)*DB(2) + B(3,I+1)*DB(3)                    00007300
   4     51 KL=KL + 1                                                        00007400
   4     50 KL=KL + ND - J                                                   00007500
   1  C                                                                      00007600
   7        KL=ND + 1                                                        00007700
   7        DO 54 J=2,ND,2                                                   00007800
   7        DO 56 K=1,3                                                      00007900
   7        DB(K)=D(K,2)*B(2,J) + D(K,3)*B(3,J)                              00008000
   4     56 DB(K)=DB(K)*FAC                                                  00008100
   7        KS=KL                                                            00008200
   7        DO 55 I=J,ND,2                                                   00008300
   7        S(KS)=S(KS) + B(2,I)*DB(2) + B(3,I)*DB(3)                        00008400
   4     55 KS=KS + 2                                                        00008500
   7        IF (J-ND) 57,54,54                                               00008600
   4     57 K=J + 1                                                          00008700
   7        KS=KL + 1                                                        00008800
   7        DO 58 II=K,ND,2                                                  00008900
   7        S(KS)=S(KS) + B(1,II)*DB(1) + B(3,II)*DB(3)                      00009000
   4     58 KS=KS + 2                                                        00009100
   4     54 KL=KL + 2*ND - 2*J + 1                                           00009200
   1  C                                                                      00009300
   7        IF (IST.EQ.3) GO TO 10                                           00009400
   7        KL=1                                                             00009500
   7        DO 60 J=1,ND,2                                                   00009600
   7        DB(1)=D(1,4)*B(4,J)*FAC                                          00009700
   7        DB(2)=D(2,4)*B(4,J)*FAC                                          00009800
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7        DB(3)=D(3,4)*B(4,J)*FAC                                              00009900
   7        DB(4)=D(4,1)*B(1,J) + D(4,3)*B(3,J) + D(4,4)*B(4,J)                  00010000
   7        DB(4)=DB(4)*FAC                                                      00010100
   7        DO 61 I=J,ND,2                                                       00010200
   7        S(KL)=S(KL) + B(1,I)*DB(1) + B(3,I)*DB(3) + B(4,I)*DB(4)             00010300
   7        KL=KL + 1                                                            00010400
   7        S(KL)=S(KL) + B(2,I+1)*DB(2) + B(3,I+1)*DB(3)                        00010500
   4     61 KL=KL + 1                                                            00010600
   4     60 KL=KL + ND - J                                                       00010700
   7        KL=ND + 1                                                            00010800
   7        DO 59 J=2,ND,2                                                       00010900
   7        DB(4)=D(4,2)*B(2,J) + D(4,3)*B(3,J)                                  00011000
   7        DB(4)=DB(4)*FAC                                                      00011100
   7        DO 62 I=J,ND                                                         00011200
   7        S(KL)=S(KL) + B(4,I)*DB(4)                                           00011300
   4     62 KL=KL + 1                                                            00011400
   4     59 KL=KL + ND - J                                                       00011500
   1  C                                                                          00011600
   4     10 CONTINUE                                                             00011700
   1  C                                                                          00011800
   7        RETURN                                                               00011900
   1  C                                                                          00012000
   1  C                                                                          00012100
   1  C     F I N D  N O N L I N E A R  E L E M E N T  M A T R I C E S            00012200
   1  C                                                                          00012300
   1  C                                                                          00012400
   1  C     UPDATE ELEMENT COORDINATES                                           00012500
   1  C                                                                          00012600
   3    100 IF (INDNL.LE.2) GO TO 122                                           00012700
   7        DO 120 J=1,ND                                                        00012800
   3    120 XX(J) = YZ(J) + EDIS(J)                                             00012900
   1  C                                                                          00013000
   1  C     EVALUATE STRESS STRAIN LAW IN CASE MATERIAL VARIABLES ARE            00013100
   1  C     CONSTANT , I.E. , FOR MODEL.EQ.1  AND MODEL.EQ.2                     00013200
   1  C                                                                          00013300
   3    122 IF (MODEL.GT.2) GO TO 125                                           00013400
   7        IF (INDNL.LE.2) GO TO 124                                            00013500
   7        CALL STSTL (NEL,XX,PROP,D)                                           00013600
   7        GO TO 125                                                            00013700
   3    124 CALL STSTL (NEL,YZ,PROP,D)                                          00013800
   1  C                                                                          00013900
   1  C                                                                          00014000
   1  C     INTEGRATE STIFFNESS MATRIX AND ELEMENT NODAL FORCE EXPRESSION        00014100
   1  C                                                                          00014200
   1  C                                                                          00014300
   3    125 DO 300 LX=1,NINT                                                    00014400
   7        E1=XG(LX,NINT)                                                       00014500
   7        DO 300 LY=1,NINT                                                     00014600
   7        E2=XG(LY,NINT)                                                       00014700
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7        WT=WGT(LX,NINT)*WGT(LY,NINT)                                    00014800
   7        IPT=(LX-1)*NINT + LY                                            00014900
   7        IF (INDNL.EQ.3) GO TO 200                                       00015000
   1  C                                                                     00015100
   1  C                                                                     00015200
   1  C     T O T A L   L A G R A N G I A N   F O R M U L A T I O N         00015300
   1  C                                                                     00015400
   1  C                                                                     00015500
   1  C     EVALUATE THE DERIVATIVE OPERATOR B                              00015600
   1  C                                                                     00015700
   7        CALL DERIQ (NEL,YZ,B,DET,E1,E2,XBAR,NOD5)                       00015800
   1  C                                                                     00015900
   1  C     CALCULATE DISPLACEMENT DERIVATIVES                              00016000
   1  C                                                                     00016100
   7        DO 130 I=1,5                                                    00016200
   3    130 DISD(I)=0.0                                                     00016300
   7        DO 140 J=2,ND,2                                                 00016400
   7        I=J - 1                                                         00016500
   7        DISD(1)=DISD(1) + B(1,I)*EDIS(I)                                00016600
   7        DISD(2)=DISD(2) + B(2,J)*EDIS(J)                                00016700
   7        DISD(3)=DISD(3) + B(3,I)*EDIS(I)                                00016800
   3    140 DISD(4)=DISD(4) + B(3,J)*EDIS(J)                                00016900
   7        IF (IST.EQ.3) GO TO 160                                         00017000
   7        DO 150 I=1,ND,2                                                 00017100
   3    150 DISD(5)=DISD(5) + B(4,I)*EDIS(I)                                00017200
   1  C                                                                     00017300
   1  C     EVALUATE STRESS-STRAIN LAW AND CURRENT STRESSES                 00017400
   1  C                                                                     00017500
   3    160 CALL STSTN (YZ,PROP,DISD,IDW,WA)                                00017600
   1  C                                                                     00017700
   7        IF (INDNL.EQ.1) GO TO 221                                       00017800
   1  C                                                                     00017900
   1  C     EVALUATE DERIVATIVE OPERATOR INCLUDING THE INITIAL              00018000
   1  C     DISPLACEMENT EFFECTS                                            00018100
   1  C                                                                     00018200
   7        DO 164 J=2,ND,2                                                 00018300
   7        I=J - 1                                                         00018400
   7        BS(1,I)=B(1,I) + B(1,I)*DISD(1)                                 00018500
   7        BS(1,J)=B(1,I)*DISD(4)                                          00018600
   7        BS(2,I)=B(2,J)*DISD(3)                                          00018700
   7        BS(2,J)=B(2,J) + B(2,J)*DISD(2)                                 00018800
   7        BS(3,I)=B(3,I) + B(3,I)*DISD(1) + B(3,J)*DISD(3)                00018900
   3    164 BS(3,J)=B(3,J) + B(3,I)*DISD(4) + B(3,J)*DISD(2)                00019000
   7        IF (IST.EQ.3) GO TO 167                                         00019100
   7        DO 166 I=1,ND,2                                                 00019200
   7        J=I + 1                                                         00019300
   7        BS(4,J)=0.0                                                     00019400
   3    166 BS(4,I)=B(4,I) + B(4,I)*DISD(5)                                 00019500
   1  C                                                                     00019600
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   1  C     ADD STRESS CONTRIBUTION TO ELEMENT FORCE VECTOR                    00019700
   1  C                                                                        00019800
   3    167 IF (IST.EQ.3) XBAR=THIC                                           00019900
   7        FAC=WT*XBAR*DET                                                   00020000
   7        TAU11=STRESS(1)*FAC                                               00020100
   7        TAU22=STRESS(2)*FAC                                               00020200
   7        TAU12=STRESS(3)*FAC                                               00020300
   7        TAU33=STRESS(4)*FAC                                               00020400
   7        DO 170 I=1,ND                                                     00020500
   3    170 RE(I)=RE(I) + BS(1,I)*TAU11 + BS(2,I)*TAU22 + BS(3,I)*TAU12       00020600
   7        IF (IST.EQ.3) GO TO 176                                           00020700
   7        DO 174 J=1,ND,2                                                   00020800
   3    174 RE(J)=RE(J) + BS(4,J)*TAU33                                       00020900
   1  C                                                                       00021000
   3    176 IF (ICOUNT - 2) 178,178,300                                       00021100
   3    178 IF (IREF) 300,179,300                                             00021200
   1  C                                                                       00021300
   1  C     ADD LINEAR CONTRIBUTION TO ELEMENT STIFFNESS MATRIX               00021400
   1  C                                                                       00021500
   3    179 DO 183 I=1,IST                                                    00021600
   7        DO 183 J=I,IST                                                    00021700
   7        DI(I,J)=D(I,J)*FAC                                                00021800
   3    183 DI(J,I)=DI(I,J)                                                   00021900
   7        KL=0                                                              00022000
   7        DO 180 J=1,ND                                                     00022100
   7        DO 182 K=1,IST                                                    00022200
   7        DB(K)=0.                                                          00022300
   7        DO 184 L=1,IST                                                    00022400
   3    184 DB(K)=DB(K) + DI(K,L)*BS(L,J)                                     00022500
   3    182 CONTINUE                                                          00022600
   1  C                                                                       00022700
   7        DO 180 I=J,ND                                                     00022800
   7        KL=KL + 1                                                         00022900
   7        DUM=0.                                                            00023000
   7        DO 186 K=1,IST                                                    00023100
   3    186 DUM=DUM + BS(K,I)*DB(K)                                           00023200
   3    180 S(KL)=S(KL) + DUM                                                 00023300
   1  C                                                                       00023400
   7        GO TO 365                                                         00023500
   1  C                                                                       00023600
   1  C                                                                       00023700
   1  C     U P D A T E D   L A G R A N G I A N   F O R M U L A T I O N       00023800
   1  C                                                                       00023900
   1  C                                                                       00024000
   1  C     EVALUATE THE DERIVATIVE OPERATOR B                                00024100
   1  C                                                                       00024200
   3    200 CALL DERIQ (NEL,XX,B,DET,E1,E2,XBAR,NOD5)                         00024300
   1  C                                                                       00024400
   1  C     CALCULATE DISPLACEMENT DERIVATIVES                                00024500
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   1  C                                                                      00024600
   7        DO 210 I=1,5                                                     00024700
   3    210 DISD(I)=0.                                                       00024800
   7        DO 212 J=2,ND,2                                                  00024900
   7        I=J - 1                                                          00025000
   7        DISD(1)=DISD(1) + B(1,I)*EDIS(I)                                 00025100
   7        DISD(2)=DISD(2) + B(2,J)*EDIS(J)                                 00025200
   7        DISD(3)=DISD(3) + B(3,I)*EDIS(I)                                 00025300
   3    212 DISD(4)=DISD(4) + B(3,J)*EDIS(J)                                 00025400
   7        IF (IST.EQ.3) GO TO 216                                          00025500
   7        DO 214 I=1,ND,2                                                  00025600
   3    214 DISD(5)=DISD(5) + B(4,I)*EDIS(I)                                 00025700
   1  C                                                                      00025800
   1  C     EVALUATE STRESS-STRAIN LAW AND CURRENT STRESSES                  00025900
   1  C                                                                      00026000
   3    216 CALL STSTN (XX,PROP,DISD,IDW,WA)                                 00026100
   1  C                                                                      00026200
   3    221 IF (IST.EQ.3) XBAR=THIC                                          00026300
   7        IF (INDNL.EQ.1) GO TO 222                                        00026400
   1  C                                                                      00026500
   7        IF (ITYP2D.LT.2) GO TO 222                                       00026600
   7        E11=DISD(1) - 0.5*(DISD(1)*DISD(1) + DISD(4)*DISD(4))            00026700
   7        E22=DISD(2) - 0.5*(DISD(2)*DISD(2) + DISD(3)*DISD(3))            00026800
   7        E12=DISD(3) + DISD(4) - DISD(3)*DISD(1) - DISD(4)*DISD(2)        00026900
   7        E33=-(D(4,1)*E11 + D(4,2)*E22 + D(4,3)*E12)/D(4,4)               00027000
   7        EXT=1. - 2.*E33                                                  00027100
   7        EXT=1. -DSQRT(EXT)                                               00027200
   7        XBAR=THIC*(1. - EXT)                                             00027300
   1  C                                                                      00027400
   3    222 FAC=WT*XBAR*DET                                                  00027500
   1  C                                                                      00027600
   1  C     ADD STRESS CONTRIBUTION TO ELEMENT FORCE VECTOR                  00027700
   1  C                                                                      00027800
   7        TAU11=STRESS(1)*FAC                                              00027900
   7        TAU22=STRESS(2)*FAC                                              00028000
   7        TAU12=STRESS(3)*FAC                                              00028100
   7        TAU33=STRESS(4)*FAC                                              00028200
   7        DO 340 J=2,ND,2                                                  00028300
   7        I=J - 1                                                          00028400
   7        RE(I)=RE(I) + B(1,I)*TAU11 + B(3,I)*TAU12                        00028500
   3    340 RE(J)=RE(J) + B(2,J)*TAU22 + B(3,J)*TAU12                        00028600
   7        IF (IST.EQ.3) GO TO 350                                          00028700
   7        DO 345 J=1,ND,2                                                  00028800
   3    345 RE(J)=RE(J) + B(4,J)*TAU33                                       00028900
   1  C                                                                      00029000
   3    350 IF (ICOUNT-2) 220,220,300                                        00029100
   3    220 IF (IREF) 300,230,300                                            00029200
   1  C                                                                      00029300
   1  C     ADD LINEAR CONTRIBUTION TO STIFFNESS MATRIX                      00029400
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   1  C                                                                      00029500
   3    230 DO 232 I=1,IST                                                   00029600
   7        DO 232 J=I,IST                                                   00029700
   7        DI(I,J)=D(I,J)*FAC                                               00029800
   3    232 DI(J,I)=DI(I,J)                                                  00029900
   7        KL=1                                                             00030000
   7        DO 250 J=1,ND,2                                                  00030100
   7        DO 252 K=1,3                                                     00030200
   3    252 DB(K)=DI(K,1)*B(1,J) + DI(K,3)*B(3,J)                            00030300
   7        DO 251 I=J,ND,2                                                  00030400
   7        S(KL)=S(KL) + B(1,I)*DB(1) + B(3,I)*DB(3)                        00030500
   7        KL=KL + 1                                                        00030600
   7        S(KL)=S(KL) + B(2,I+1)*DB(2) + B(3,I+1)*DB(3)                    00030700
   3    251 KL=KL + 1                                                        00030800
   3    250 KL=KL + ND - J                                                   00030900
   7        KL=ND + 1                                                        00031000
   1  C                                                                      00031100
   7        DO 254 J=2,ND,2                                                  00031200
   7        DO 256 K=1,3                                                     00031300
   3    256 DB(K)=DI(K,2)*B(2,J) + DI(K,3)*B(3,J)                            00031400
   7        KS=KL                                                            00031500
   7        DO 255 I=J,ND,2                                                  00031600
   7        S(KS)=S(KS) + B(2,I)*DB(2) + B(3,I)*DB(3)                        00031700
   3    255 KS=KS + 2                                                        00031800
   7        IF (J-ND) 257,254,254                                            00031900
   3    257 K=J + 1                                                          00032000
   7        KS=KL + 1                                                        00032100
   7        DO 258 II=K,ND,2                                                 00032200
   7        S(KS)=S(KS) + B(1,II)*DB(1) + B(3,II)*DB(3)                      00032300
   3    258 KS=KS + 2                                                        00032400
   3    254 KL=KL + 2*ND - 2*J + 1                                           00032500
   1  C                                                                      00032600
   7        IF (IST.EQ.3) GO TO 365                                          00032700
   7        KL=1                                                             00032800
   7        DO 260 J=1,ND,2                                                  00032900
   7        DB(1)=DI(1,4)*B(4,J)                                             00033000
   7        DB(2)=DI(2,4)*B(4,J)                                             00033100
   7        DB(3)=DI(3,4)*B(4,J)                                             00033200
   7        DB(4)=DI(4,1)*B(1,J) + DI(4,3)*B(3,J) + DI(4,4)*B(4,J)           00033300
   7        DO 261 I=J,ND,2                                                  00033400
   7        S(KL)=S(KL) + B(1,I)*DB(1) + B(3,I)*DB(3) + B(4,I)*DB(4)         00033500
   7        KL=KL + 1                                                        00033600
   7        S(KL)=S(KL) + B(2,I+1)*DB(2) + B(3,I+1)*DB(3)                    00033700
   3    261 KL=KL + 1                                                        00033800
   3    260 KL=KL + ND - J                                                   00033900
   7        KL=ND + 1                                                        00034000
   7        DO 259 J=2,ND,2                                                  00034100
   7        DB(4)=DI(4,2)*B(2,J) + DI(4,3)*B(3,J)                            00034200
   7        DO 262 I=J,ND                                                    00034300
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7        S(KL)=S(KL) + B(4,I)*DB(4)                                      00034400
   3     262 KL=KL + 1                                                      00034500
   3     259 KL=KL + ND - J                                                 00034600
   1  C                                                                     00034700
   1  C                                                                     00034800
   1  C     T O T A L   A N D   U P D A T E D   L A G R A N G I A N         00034900
   1  C                              F O R M U L A T I O N                  00035000
   1  C                                                                     00035100
   1  C                                                                     00035200
   1  C     ADD NONLINEAR CONTRIBUTION TO STIFFNESS MATRIX                  00035300
   1  C                                                                     00035400
   3     365 IF (INDNL.EQ.1) GO TO 300                                      00035500
   1  C                                                                     00035600
   7        KL=1                                                           00035700
   7        DO 400 J=1,ND,2                                                00035800
   7        DB1=TAU11*B(1,J) + TAU12*B(3,J)                                00035900
   7        DB2=TAU12*B(1,J) + TAU22*B(3,J)                                00036000
   1  C                                                                     00036100
   7        KS=KL                                                          00036200
   7        DO 401 I=J,ND,2                                                00036300
   7        KSS=KS + ND - J + 1                                            00036400
   7        DUM=B(1,I)*DB1 + B(3,I)*DB2                                    00036500
   7        S(KS)=S(KS) + DUM                                              00036600
   7        S(KSS)=S(KSS) + DUM                                            00036700
   3     401 KS=KS + 2                                                     00036800
   3     400 KL=KL + 2*ND - 2*J + 1                                        00036900
   1  C                                                                     00037000
   7        IF (IST.EQ.3) GO TO 300                                        00037100
   7        KL=1                                                           00037200
   7        DO 420 J=1,ND,2                                                00037300
   7        DB3=TAU33*B(4,J)                                               00037400
   7        DO 421 I=J,ND,2                                                00037500
   7        S(KL)=S(KL) + DB3*B(4,I)                                       00037600
   3     421 KL=KL + 2                                                     00037700
   3     420 KL=KL + ND - J                                                00037800
   1  C                                                                     00037900
   3     300 CONTINUE                                                      00038000
   1  C                                                                     00038100
   1  C                                                                     00038200
   7        RETURN                                                         00038300
   7        END                                                            00038400
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7        SUBROUTINE QUADS3  (ND,B,S,XYZ,PROP,RE,EDIS,WA,NOD9)                  00000100
   7        IMPLICIT REAL*8(A-H,O-Z)                                              00000200
   1  C                                                                           00000300
   1  C                                                                           00000400
   1  C . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 00000500
   1  C .                                                                       . 00000600
   1  C .                                                                       . 00000700
   1  C .      HEXAHEDRAL CURVILINEAR THREE-DIMENSIONAL ELEMENTS                . 00000800
   1  C .                                                                       . 00000900
   1  C .      ISOPARAMETRIC OR SUBPARAMETRIC                                   . 00001000
   1  C .                                                                       . 00001100
   1  C .                                                                       . 00001200
   1  C . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 00001300
   1  C                                                                           00001400
   1  C                                                                           00001500
   1  C                                                                           00001600
   7        COMMON /EL/ IND,ICOUNT,NPAR(20),NUMEG,NEGL,NEGNL,IMASS,IDAMP,ISTAT00001700
   6       1           ,NDOFDM,KLIN,IEIG,IMASSN,IDAMPN                          00001800
   7        COMMON /VAR/ NG,KPRI,MODEX,KSTEP,ITE,ITEMAX,IREF,IEQREF,INOCMD     00001900
   7        COMMON /THREED/ DE,IELD,IELX,IEL,NPT,IDW,NND9                      00002000
   7        COMMON /MTMD3D/  D(6,6),STRESS(6),STRAIN(6),IPT,NEL                00002100
   1  C                                                                        00002200
   7        DIMENSION B(1),S(1),XYZ(1),PROP(1),RE(1),EDIS(1),                  00002300
   6       1          WA(1),NOD9(1)                                           00002400
   7        DIMENSION XG(4,4),WGT(4,4),BB(9),DISD(9),TAU(6)                    00002500
   1  C                                                                        00002600
   7        EQUIVALENCE (NPAR(15),MODEL),(NPAR(10),NINT),(NPAR(11),NINTZ)      00002700
   1  C                                                                        00002800
   7        DATA XG /     0.,              0.,              0.,              0., 00002900
   6       1 -.5773502691896, .5773502691896,              0.,              0., 00003000
   6       2 -.7745966692415, .0000000000000, .7745966692415,              0., 00003100
   6       3 -.8611363115941,-.3399810435849, .3399810435849, .8611363115941/ 00003200
   7        DATA WGT / 2.000,              0.,              0.,              0., 00003300
   6       1 1.0000000000000,1.0000000000000,              0.,              0., 00003400
   6       2  .5555555555556, .8888888888889, .5555555555556,              0., 00003500
   6       3  .3478548451375, .6521451548625, .6521451548625, .3478548451375/ 00003600
   1  C                                                                        00003700
   1  C                                                                        00003800
   7        IF (IND.GE.4) GO TO 100                                            00003900
   1  C                                                                        00004000
   1  C                                                                        00004100
   1  C     F I N D  S T I F F N E S S  O F  L I N E A R  E L E M E N T        00004200
   1  C                                                                        00004300
   1  C                                                                        00004400
   1  C                                                                        00004500
   1  C     INTEGRATE B(TRANSPOSED) * B                                        00004600
   1  C                                                                        00004700
   7        DO 10 LX=1,NINT                                                    00004800
   7        E1=XG(LX,NINT)                                                     00004900
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7        DO 10 LY=1,NINT                                                    00005000
   7        E2=XG(LY,NINT)                                                     00005100
   7        DO 10 LZ=1,NINTZ                                                   00005200
   7        E3=XG(LZ,NINTZ)                                                    00005300
   7        WT=WGT(LX,NINT)*WGT(LY,NINT)*WGT(LZ,NINTZ)                         00005400
   1  C                                                                        00005500
   1  C     EVALUATE STRAIN-DISPLACEMENT MATRIX B AND JACOBIAN DETERMINANT     00005600
   1  C     AT THIS INTEGRATION POINT                                          00005700
   1  C                                                                        00005800
   7        CALL DERIQ3  (NEL,XYZ,B,DET,E1,E2,E3,NOD9)                         00005900
   1  C                                                                        00006000
   7        FAC=WT*DET                                                         00006100
   7        FAC=DSQRT(FAC)                                                     00006200
   7        DO 40 I=1,ND                                                       00006300
   4     40 B(I)=FAC*B(I)                                                      00006400
   7        KL=0                                                               00006500
   7        DO 45 I=1,ND                                                       00006600
   7        DO 45 J=I,ND                                                       00006700
   7        KL=KL + 1                                                          00006800
   4     45 S(KL)=S(KL) + B(I)*B(J)                                            00006900
   4     10 CONTINUE                                                           00007000
   1  C                                                                        00007100
   1  C                                                                        00007200
   1  C     MULTIPLY D BY THE INTEGRATED B(TRANSPOSED) * B                     00007300
   1  C                                                                        00007400
   7        CALL STST3L  (NEL,XYZ,PROP,D)                                      00007500
   1  C                                                                        00007600
   1  C     ISOTROPIC CASE                                                     00007700
   1  C                                                                        00007800
   7        IF (MODEL.GT.1) GO TO 70                                           00007900
   7        KL=1                                                               00008000
   7        D1=D(1,1)                                                          00008100
   7        D2=D(1,2)                                                          00008200
   7        D3=D(4,4)                                                          00008300
   7        DO 60 II=1,IEL                                                     00008400
   7        I0=3*(II-1)                                                        00008500
   7        DO 61 JJ=II,IEL                                                    00008600
   7        J0=3*(JJ-1)                                                        00008700
   1  C                                                                        00008800
   7        KS=KL                                                              00008900
   7        IC=0                                                               00009000
   7        DO 63 I=1,3                                                        00009100
   7        DO 62 J=1,3                                                        00009200
   7        IC=IC + 1                                                          00009300
   7        BB(IC)=S(KS)                                                       00009400
   4     62 KS=KS + 1                                                          00009500
   4     63 KS=KS + ND - I0 - I - 3                                            00009600
   1  C                                                                        00009700
   7        KS1=KL                                                             00009800
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7        KS2=KS1 + ND - I0 - 1                                                00009900
   7        KS3=KS2 + ND - I0 - 2                                                00010000
   7        S(KS1)=BB(1)*D1 + (BB(5) + BB(9))*D3                                 00010100
   7        S(KS2+1)=BB(5)*D1 + (BB(1) + BB(9))*D3                               00010200
   7        S(KS3+2)=BB(9)*D1 + (BB(1) + BB(5))*D3                               00010300
   7        IF (II.EQ.JJ) GO TO 64                                               00010400
   7        S(KS1+1)=BB(2)*D2 + BB(4)*D3                                         00010500
   7        S(KS2)=BB(4)*D2 + BB(2)*D3                                           00010600
   7        S(KS1+2)=BB(3)*D2 + BB(7)*D3                                         00010700
   7        S(KS3)=BB(7)*D2 + BB(3)*D3                                           00010800
   7        S(KS2+2)=BB(6)*D2 + BB(8)*D3                                         00010900
   7        S(KS3+1)=BB(8)*D2 + BB(6)*D3                                         00011000
   7        GO TO 61                                                             00011100
   4     64 S(KS1+1)=BB(2)*(D2 + D3)                                             00011200
   7        S(KS1+2)=BB(3)*(D2 + D3)                                             00011300
   7        S(KS2+2)=BB(6)*(D2 + D3)                                             00011400
   4     61 KL=KL + 3                                                            00011500
   4     60 KL=KL + 2*(ND-I0) - 3                                                00011600
   7        RETURN                                                               00011700
   1  C                                                                          00011800
   1  C     ANISOTROPIC CASE                                                     00011900
   1  C                                                                          00012000
   4     70 CONTINUE                                                             00012100
   7        RETURN                                                               00012200
   1  C                                                                          00012300
   1  C                                                                          00012400
   1  C     F I N D  N O N L I N E A R  E L E M E N T  M A T R I C E S            00012500
   1  C                                                                          00012600
   1  C                                                                          00012700
   1  C     EVALUATE STRESS-STRAIN LAW IF LINEAR MATERIAL MODEL                  00012800
   1  C     USED IN THIS ELEMENT                                                 00012900
   1  C                                                                          00013000
   3    100 IF (MODEL.GT.2) GO TO 125                                            00013100
   7        CALL STST3L  (NEL,XYZ,PROP,D)                                        00013200
   1  C                                                                          00013300
   1  C                                                                          00013400
   1  C     INTEGRATE STIFFNESS MATRIX AND ELEMENT NODAL FORCE EXPRESSION        00013500
   1  C                                                                          00013600
   1  C                                                                          00013700
   3    125 IPT=0                                                               00013800
   7        DO 300 LX=1,NINT                                                     00013900
   7        E1=XG(LX,NINT)                                                       00014000
   7        DO 300 LY=1,NINT                                                     00014100
   7        E2=XG(LY,NINT)                                                       00014200
   7        DO 300 LZ=1,NINTZ                                                    00014300
   7        E3=XG(LZ,NINTZ)                                                      00014400
   7        WT=WGT(LX,NINT)*WGT(LY,NINT)*WGT(LZ,NINTZ)                           00014500
   7        IPT=IPT + 1                                                          00014600
   1  C                                                                          00014700
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   1  C                                                                      00014800
   1  C     M A T E R I A L   N O N L I N E A R I T Y   F O R M U L A T I O N 00014900
   1  C                                                                      00015000
   1  C          ( NO KINEMATIC NONLINEARITIES ACCOUNTED FOR )               00015100
   1  C                                                                      00015200
   1  C                                                                      00015300
   1  C     EVALUATE DERIVATIVE OPERATOR B (IN COMPACTED FORM)               00015400
   1  C                                                                      00015500
   7        CALL DERIQ3  (NEL,XYZ,B,DET,E1,E2,E3,NOD9)                      00015600
   1  C                                                                      00015700
   1  C     CALCULATE DISPLACEMENT DERIVATIVES                               00015800
   1  C                                                                      00015900
   7        DO 130 I=1,9                                                    00016000
   3     130 DISD(I)=0.0                                                    00016100
   7        DO 140 J=3,ND,3                                                 00016200
   7        I=J - 1                                                         00016300
   7        K= J - 2                                                        00016400
   7        DISD(1)=DISD(1) + B(K)*EDIS(K)                                  00016500
   7        DISD(2)=DISD(2) + B(I)*EDIS(I)                                  00016600
   7        DISD(3)=DISD(3) + B(J)*EDIS(J)                                  00016700
   7        DISD(4)=DISD(4) + B(I)*EDIS(K)                                  00016800
   7        DISD(5)=DISD(5) + B(J)*EDIS(K)                                  00016900
   7        DISD(6)=DISD(6) + B(K)*EDIS(I)                                  00017000
   7        DISD(7)=DISD(7) + B(J)*EDIS(I)                                  00017100
   7        DISD(8)=DISD(8) + B(K)*EDIS(J)                                  00017200
   3     140 DISD(9)=DISD(9) + B(I)*EDIS(J)                                 00017300
   1  C                                                                      00017400
   1  C     EVALUATE STRESS-STRAIN LAW AND CURRENT STRESSES                  00017500
   1  C                                                                      00017600
   7        CALL STST3N (DISD)                                              00017700
   1  C                                                                      00017800
   1  C     ADD STRESS CONTRIBUTION TO ELEMENT FORCE VECTOR                  00017900
   1  C                                                                      00018000
   7        FAC=WT*DET                                                      00018100
   7        DO 215 I=1,6                                                    00018200
   3     215 TAU(I)=STRESS(I)*FAC                                           00018300
   7        DO 216 K=3,ND,3                                                 00018400
   7        I=K - 2                                                         00018500
   7        J=K - 1                                                         00018600
   7        RE(I)=RE(I) + B(I)*TAU(1) + B(J)*TAU(4) + B(K)*TAU(5)           00018700
   7        RE(J)=RE(J) + B(J)*TAU(2) + B(I)*TAU(4) + B(K)*TAU(6)           00018800
   3     216 RE(K)=RE(K) + B(K)*TAU(3) + B(I)*TAU(5) + B(J)*TAU(6)          00018900
   1  C                                                                      00019000
   7        IF (ICOUNT - 2) 220,220,300                                     00019100
   3     220 IF (IREF) 300,230,300                                          00019200
   1  C                                                                      00019300
   1  C     ADD LINEAR CONTRIBUTION TO ELEMENT STIFFNESS MATRIX              00019400
   1  C                                                                      00019500
   1  C        ISOTROPIC VERSION OF B(TRANSPOSED)*D*B                       00019600
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   1  C                                                                      00019700
   3    230 IF (MODEL.EQ.1 .OR. MODEL.EQ.3) GO TO 232                        00019800
   7        GO TO 320                                                        00019900
   3    232 D11=D(1,1)*FAC                                                   00020000
   7        D12=D(1,2)*FAC                                                   00020100
   7        D44=D(4,4)*FAC                                                   00020200
   7        KL=1                                                             00020300
   7        DO 250 J=1,ND,3                                                  00020400
   7        DB1=D11*B(J)                                                     00020500
   7        DB2=D12*B(J)                                                     00020600
   7        DB3=D12*B(J)                                                     00020700
   7        DB4=D44*B(J+1)                                                   00020800
   7        DB5=D44*B(J+2)                                                   00020900
   7        DO 251 I=J,ND,3                                                  00021000
   7        S(KL)=S(KL) + B(I)*DB1 + B(I+1)*DB4 + B(I+2)*DB5                 00021100
   7        KL=KL + 1                                                        00021200
   7        S(KL)=S(KL) + B(I+1)*DB2 + B(I)*DB4                              00021300
   7        KL=KL + 1                                                        00021400
   7        S(KL)=S(KL) + B(I+2)*DB3 + B(I)*DB5                              00021500
   3    251 KL=KL + 1                                                        00021600
   3    250 KL=KL + 2*(ND-J) - 1                                             00021700
   1  C                                                                      00021800
   7        KL=ND + 1                                                        00021900
   7        DO 254 J=2,ND,3                                                  00022000
   7        DB1=D12*B(J)                                                     00022100
   7        DB2=D11*B(J)                                                     00022200
   7        DB3=D12*B(J)                                                     00022300
   7        DB4=D44*B(J-1)                                                   00022400
   7        DB6=D44*B(J+1)                                                   00022500
   7        S(KL)=S(KL) + B(J)*DB2 + B(J-1)*DB4 + B(J+1)*DB6                 00022600
   7        KL=KL + 1                                                        00022700
   7        S(KL)=S(KL) + B(J+1)*DB3 + B(J)*DB6                              00022800
   7        KL=KL + 1                                                        00022900
   7        JM=J + 2                                                         00023000
   7        IF (ND-JM) 254,253,253                                           00023100
   3    253 DO 255 I=JM,ND,3                                                 00023200
   7        S(KL)=S(KL) + B(I)*DB1 + B(I+1)*DB4                              00023300
   7        KL=KL + 1                                                        00023400
   7        S(KL)=S(KL) + B(I+1)*DB2 + B(I)*DB4 + B(I+2)*DB6                 00023500
   7        KL=KL + 1                                                        00023600
   7        S(KL)=S(KL) + B(I+2)*DB3 + B(I+1)*DB6                            00023700
   3    255 KL=KL + 1                                                        00023800
   3    254 KL=KL + 2*(ND-J) - 1                                             00023900
   1  C                                                                      00024000
   7        KL=2*ND                                                          00024100
   7        DO 256 J=3,ND,3                                                  00024200
   7        DB1=D12*B(J)                                                     00024300
   7        DB2=D12*B(J)                                                     00024400
   7        DB3=D11*B(J)                                                     00024500
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7        DB5=D44*B(J-2)                                                    00024600
   7        DB6=D44*B(J-1)                                                    00024700
   7        S(KL)=S(KL) + B(J)*DB3 + B(J-2)*DB5 + B(J-1)*DB6                  00024800
   7        KL=KL + 1                                                         00024900
   7        JM=J + 1                                                          00025000
   7        IF (ND-JM) 256,257,257                                            00025100
   3    257 DO 258 I=JM,ND,3                                                  00025200
   7        S(KL)=S(KL) + B(I)*DB1 + B(I+2)*DB5                               00025300
   7        KL=KL + 1                                                         00025400
   7        S(KL)=S(KL) + B(I+1)*DB2 + B(I+2)*DB6                             00025500
   7        KL=KL + 1                                                         00025600
   7        S(KL)=S(KL) + B(I+2)*DB3 + B(I)*DB5 + B(I+1)*DB6                  00025700
   3    258 KL=KL + 1                                                         00025800
   3    256 KL=KL + 2*(ND-J) - 1                                              00025900
   7        GO TO 300                                                         00026000
   1  C                                                                       00026100
   1  C         ORTHOTROPIC VERSION OF B(TRANSPOSED)*D*B                      00026200
   1  C                                                                       00026300
   3    320 CONTINUE                                                          00026400
   1  C                                                                       00026500
   1  C                                                                       00026600
   3    300 CONTINUE                                                          00026700
   7        RETURN                                                            00026800
   1  C                                                                       00026900
   1  C                                                                       00027000
   1  C                                                                       00027100
   7        END                                                               00027200
```

```
START                                                                    MOD
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8  FLAGS

   7        SUBROUTINE READID(ID,NDOF,NUMNP)                                 00000100
   7        IMPLICIT REAL*8(A-H,O-Z)                                         00000200
   1  C                                                                      00000300
   1  C     PROGRAM FOR GENERATING IDENTIFICATION ARRAY OF BOUNDARY CONDITION 00000400
   1  C     PROGRAMED BY S.N.MUKHERJEE. BBC,BADEN.AUG.1976.                  00000501   **
   1  C                                                                      00000600
   7        COMMON/BBCCIO/KIO1,KIO2,KIO3,KIO4,KIO5,KIO6,KIO7,KIO8,KIO9,      00000700
   6       1KIO10,KIO11,KIO12,KIO13,KIO14,KIO15,KIO16,KIO17,KIO18,KIO19,KIO20,00000800
   6       2KIO21,KIO22,KIO23,KIO24,KIO25,KIO26,KIO27,KIO28,KIO29,KIO30      00000900
   7        DIMENSION ID(NDOF,1)                                             00001000
   7        REWIND KIO18                                                     00001100
   7        READ (KIO18)                                                     00001200
   6       *          ((ID(I,J),I=1,NDOF),J=1,NUMNP)                        00001300
   7        RETURN                                                           00001400
   7        END                                                              00001500
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7        SUBROUTINE RUSS (ID,X,Y,Z,HT,E,DEN,AREA,STRAI,LM,XYZ,MATP,PINIT,   00000100
   6       1                 IPS,PROP,NCON,NDOF)                               00000200
   7        IMPLICIT REAL*8(A-H,O-Z)                                           00000300
   1  C                                                                        00000400
   7        COMMON/BBCCIO/KIO1,KIO2,KIO3,KIO4,KIO5,KIO6,KIO7,KIO8,KIO9,       00000500
   6       1KIO10,KIO11,KIO12,KIO13,KIO14,KIO15,KIO16,KIO17,KIO18,KIO19,KIO20,00000600
   6       2KIO21,KIO22,KIO23,KIO24,KIO25,KIO26,KIO27,KIO28,KIO29,KIO30       00000700
   7        COMMON /SOL/ NUMNP,NEQ,NWK,NWM,NWC,NUMEST,MIDEST,MAXEST,NSTE,MA   00000800
   7        COMMON /EL/ IND,ICOUNT,NPAR(20),NUMEG,NEGL,NEGNL,IMASS,IDAMP,ISTAT00000900
   6       1           ,NDOFDM,KLIN,IEIG,IMASSN,IDAMPN                       00001000
   7        COMMON /DIM/ N1,N2,N3,N4,N5,N6,N7,N8,N9,N10,N11,N12,N13,N14,N15   00001100
   7        COMMON /CONST/ DT,DTA,A0,A1,A2,A3,A4,A5,A6,A7,A8,A9,A10,A11       00001200
   6       1             ,A12,A13,A14,A15,A16,A17,A18,A19,A20,IOPE           00001300
   7        COMMON /DAMPNG/ ADAMP,BDAMP                                       00001400
   7        COMMON /EM/ S(21),XM(6),ST(6),D(3),AEM(464)                       00001500
   7        COMMON /VAR/ NG,KPRI,MODEX,KSTEP,ITE,ITEMAX,IREF,IEQREF,INOCMD    00001600
   7        COMMON A(1)                                                       00001700
   1  C                                                                        00001800
   7        DIMENSION X(1),Y(1),Z(1),ID(NDOF,1),E(1),DEN(1),AREA(1),LM(6,1),  00001900
   6       1          XYZ(6,1),MATP(1),HT(1),STRAI(1),PINIT(1),PROP(NCON,1)  00002000
   7        DIMENSION DR(3),IPS(1)                                            00002100
   1  C                                                                        00002200
   7        EQUIVALENCE (NPAR(1),NPAR1),(NPAR(2),NUME),(NPAR(16),NUMMAT)      00002300
   6       1           ,(NPAR(15),MODEL),(NPAR(3),INDNL)                     00002400
   1  C                                                                        00002500
   1  C     ** NOTE ** DURING THE TIME INTEGRATION, X=DISP, Y=VEL, Z=ACC, HT=R00002600
   1  C                                                                        00002700
   7        ND=6                                                              00002800
   1  C                                                                        00002900
   7        IF (KPRI.EQ.0) GO TO 800                                          00003000
   7        IF (IND.GT.0) GO TO 420                                           00003100
   1  C                                                                        00003200
   1  C                                                                        00003300
   1  C     R E A D   A N D   G E N E R A T E   E L E M E N T                  00003400
   1  C     I N F O R M A T I O N                                              00003500
   1  C                                                                        00003600
   1  C                                                                        00003700
   7        IF (INDNL.EQ.0) INDNL=1                                           00003800
   7        WRITE(KIO6,2030)                                                  00003900
   6       *               NPAR1,NUME,INDNL                                   00004000
   7        IF (MODEL.EQ.0) MODEL=1                                           00004100
   7        IF (NUMMAT.EQ.0) NUMMAT=1                                         00004200
   7        WRITE(KIO6,2000)                                                  00004300
   6       *               MODEL,NUMMAT,NCON                                  00004400
   7        GO TO (1,2), MODEL                                                00004500
   1  C                                                                        00004600
   5      1 WRITE(KIO6,2001)                                                  00004700
   7        DO 10 I=1,NUMMAT                                                  00004800
   7        READ(KIO5,1000) N                                                 00004900
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7        READ(KIO5,1001)                                                  00005000
   6       *                E(N),AREA(N),DEN(N),STRAI(N)                     00005100
   4     10 WRITE(KIO6,2002)                                                00005200
   6       *               N,E(N),AREA(N),DEN(N),STRAI(N)                   00005300
   7        GO TO 20                                                        00005400
   1  C                                                                     00005500
   5      2 IP=NCON/2                                                       00005600
   7        DO 15 I=1,NUMMAT                                                00005700
   7        READ(KIO5,1000) N                                               00005800
   7        READ (KIO5,1006)                                                00005900
   6       *              AREA(N),DEN(N),STRAI(N)                           00006000
   7        READ(KIO5,1005)                                                 00006100
   6       *              (PROP(J,N),J=1,NCON)                              00006200
   7        WRITE(KIO6,2006) I                                              00006300
   7        DO 16 J=1,IP                                                    00006400
   4     16 WRITE(KIO6,2007)                                                00006500
   6       *               J,PROP(J,N),PROP(J+IP,N)                         00006600
   7        WRITE(KIO6,2008)                                                00006700
   6       *               AREA(N),DEN(N),STRAI(N)                          00006800
   4     15 CONTINUE                                                        00006900
   1  C                                                                     00007000
   1  C     ELEMENT INFORMATION                                             00007100
   1  C                                                                     00007200
   4     20 WRITE(KIO6,2005)                                                00007300
   7        N=1                                                             00007400
   3    100 READ(KIO5,1004)                                                00007500
   6       *              M,II,JJ,MTYP,PIN,IS,KG                            00007600
   7        IF (KG.EQ.0) KG=1                                               00007700
   3    120 IF(M.NE.N) GO TO 200                                           00007800
   7        I=II                                                            00007900
   7        J=JJ                                                            00008000
   7        MTYPE=MTYP                                                      00008100
   7        KKK=KG                                                          00008200
   7        PINT=PIN                                                        00008300
   7        IPST=IS                                                         00008400
   1  C                                                                     00008500
   1  C     SAVE ELEMENT INFORMATION                                        00008600
   1  C                                                                     00008700
   3    200 XYZ(1,N)=X(I)                                                  00008800
   7        XYZ(2,N)=Y(I)                                                   00008900
   7        XYZ(3,N)=Z(I)                                                   00009000
   1  C                                                                     00009100
   7        XYZ(4,N)=X(J)                                                   00009200
   7        XYZ(5,N)=Y(J)                                                   00009300
   7        XYZ(6,N)=Z(J)                                                   00009400
   1  C                                                                     00009500
   1  C                                                                     00009600
   7        MATP(N)=MTYPE                                                   00009700
   7        PINIT(N)=PINT                                                   00009800
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7          IPS(N)=IPST                                                    00009900
   1  C                                                                      00010000
   7          DO 390 L=1,6                                                   00010100
   3    390   LM(L,N)=0.                                                     00010200
   7          DO 400 L=1,NDOF                                                00010300
   7          LL=L                                                           00010400
   7          IF (NDOF.EQ.2) LL=L + 1                                        00010500
   7          LM(LL,N)=ID(L,I)                                               00010600
   3    400   LM(LL+3,N)=ID(L,J)                                             00010700
   1  C                                                                      00010800
   1  C       UPDATE COLUMN HEIGHTS AND BANDWIDTH                            00010900
   1  C                                                                      00011000
   7          CALL COLHT (HT,ND,LM(1,N))                                     00011100
   1  C                                                                      00011200
   7          WRITE(KIO6,2004)                                               00011300
   6         *                N,I,J,MTYPE,PINT,IPST                         00011400
   7          IF (N.EQ.NUME) RETURN                                          00011500
   7          N=N+1                                                          00011600
   7          I=I+KKK                                                        00011700
   7          J=J+KKK                                                        00011800
   7          IF(N.GT.M) GO TO 100                                           00011900
   7          GO TO 120                                                      00012000
   1  C                                                                      00012100
   3    420   GO TO (440,560,560,700),IND                                    00012200
   1  C                                                                      00012300
   1  C                                                                      00012400
   1  C       A S S E M B L E   E F F E C T I V E   L I N E A R              00012500
   1  C       S T R U C T U R E   S T I F F N E S S                          00012600
   1  C                                                                      00012700
   1  C                                                                      00012800
   3    440   GO TO (520,540),ICOUNT                                         00012900
   1  C                                                                      00013000
   3    520   DO 500 N=1,NUME                                                00013100
   7          MTYPE=MATP(N)                                                  00013200
   7          XL2=0.                                                         00013300
   7          DO 505 L=1,3                                                   00013400
   7          D(L)=XYZ(L,N) - XYZ(L+3,N)                                     00013500
   3    505   XL2=XL2 + D(L)*D(L)                                            00013600
   7          XL=DSQRT(XL2)                                                  00013700
   7          XX=E(MTYPE)*AREA(MTYPE)*XL                                     00013800
   7          DO 510 L=1,3                                                   00013900
   7          ST(L)=D(L)/XL2                                                 00014000
   3    510   ST(L+3)=-ST(L)                                                 00014100
   1  C                                                                      00014200
   7          KL=0                                                           00014300
   7          DO 600 L=1,6                                                   00014400
   7          YY=ST(L)*XX                                                    00014500
   7          DO 600 K=L,6                                                   00014600
   7          KL=KL + 1                                                      00014700
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   3     600 S(KL)=ST(K)*YY                                                    00014800
   7         CALL ADDBAN (A(N2),A(N1),S,LM(1,N),ND)                            00014900
   3     500 CONTINUE                                                          00015000
   7         RETURN                                                            00015100
                                                                               00015200
   1  C                                                                        00015300
   1  C      ADD MASS AND ADAMP*MASS PORTION OF RAYLEIGH DAMPING               00015400
   1  C                                                                        00015500
   3     540 GO TO (545,550), IMASS                                            00015600
   1  C                                                                        00015700
   1  C      LUMPED MASS DISCRETIZATION                                        00015800
   1  C                                                                        00015900
   3     545 DO 640 N=1,NUME                                                   00016000
   7         MTYPE=MATP(N)                                                     00016100
   7         XL2=0.                                                            00016200
   7         DO 645 L=1,3                                                      00016300
   7         D(L)=XYZ(L,N) - XYZ(L+3,N)                                        00016400
   3     645 XL2=XL2 + D(L)*D(L)                                               00016500
   7         XL=DSQRT(XL2)                                                     00016600
   7         XMM=XL*AREA(MTYPE)*DEN(MTYPE)/2.                                  00016700
   7         DO 620 L=1,6                                                      00016800
   3     620 XM(L)=XMM                                                         00016900
   7         IST=0                                                             00017000
   7         FAC=A0 + ADAMP*A1                                                 00017100
   7         CALL ADDMA (A(N5),XM,LM(1,N),ND,FAC,IST)                          00017200
   7         XMM=XMM*FAC                                                       00017300
   7         DO 621 L=1,6                                                      00017400
   3     621 XM(L)=XMM                                                         00017500
   7         CALL ADDDI (A(N2),A(N1),XM,LM(1,N),ND)                            00017600
   3     640 CONTINUE                                                          00017700
   1  C                                                                        00017800
   7         RETURN                                                            00017900
   1  C                                                                        00018000
   1  C      CONSISTENT MASS DISCRETIZATION                                    00018100
   1  C                                                                        00018200
   3     550 DO 650 N=1,NUME                                                   00018300
   7         MTYPE=MATP(N)                                                     00018400
   7         XL2=0.                                                            00018500
   7         DO 652 L=1,3                                                      00018600
   7         D(L)=XYZ(L,N) - XYZ(L+3,N)                                        00018700
   3     652 XL2=XL2 + D(L)*D(L)                                               00018800
   7         XL=DSQRT(XL2)                                                     00018900
   7         XMM=DEN(MTYPE)*AREA(MTYPE)*XL*(A0 + ADAMP*A1)/6.                  00019000
   7         DO 653 L=1,20                                                     00019100
   3     653 S(L)=0.                                                           00019200
   7         KL=1                                                              00019300
   7         DO 654 L=1,3                                                      00019400
   7         S(KL)=XMM*2.                                                      00019500
   7         S(KL+3)=XMM                                                       00019600
   3     654 KL=KL + 7 - L
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7        S(16)=XMM*2.                                                          00019700
   7        S(19)=S(16)                                                           00019800
   7        S(21)=S(16)                                                           00019900
   7        CALL ADDBAN (A(N2),A(N1),S,LM(1,N),ND)                                00020000
   3     650 CONTINUE                                                             00020100
   1  C                                                                           00020200
   7        RETURN                                                                00020300
   1  C                                                                           00020400
   1  C                                                                           00020500
   1  C     A S S E M B L E  M A S S  M A T R I X                                 00020600
   1  C                                                                           00020700
   1  C                                                                           00020800
   1  C     CONSISTENT MASS DISCRETIZATION                                        00020900
   1  C                                                                           00021000
   3     560 DO 670 N=1,NUME                                                      00021100
   7        MTYPE=MATP(N)                                                         00021200
   7        XL2=0.                                                                00021300
   7        DO 675 L=1,3                                                          00021400
   7        D(L)=XYZ(L,N) - XYZ(L+3,N)                                            00021500
   3     675 XL2=XL2 + D(L)*D(L)                                                  00021600
   7        XL=DSQRT(XL2)                                                         00021700
   7        XMM=DEN(MTYPE)*AREA(MTYPE)*XL/6.                                      00021800
   7        DO 676 L=1,20                                                         00021900
   3     676 S(L)=0.                                                              00022000
   7        KL=1                                                                  00022100
   7        DO 677 L=1,3                                                          00022200
   7        S(KL)=XMM*2.                                                          00022300
   7        S(KL+3)=XMM                                                           00022400
   3     677 KL=KL + 7 - L                                                        00022500
   7        S(16)=XMM*2.                                                          00022600
   7        S(19)=S(16)                                                           00022700
   7        S(21)=S(16)                                                           00022800
   7        CALL ADDBAN (A(N2),A(N1),S,LM(1,N),ND)                                00022900
   3     670 CONTINUE                                                             00023000
   1  C                                                                           00023100
   7        RETURN                                                                00023200
   1  C                                                                           00023300
   1  C                                                                           00023400
   1  C     A S S E M B L E  N O N L I N E A R  F I N A L  S T R U C T U R        00023500
   1  C     S T I F F N E S S  A N D  E F F E C T I V E  L O A D S                 00023600
   1  C                                                                           00023700
   1  C                                                                           00023800
   3     700 DO 710 N=1,NUME                                                      00023900
   7        MTYPE=MATP(N)                                                         00024000
   7        PINT=PINIT(N)                                                         00024100
   7        XL2=0.                                                                00024200
   7        XLO=0.                                                                00024300
   7        DO 720 L=1,3                                                          00024400
   7        DR(L)=XYZ(L+3,N) - XYZ(L,N)                                           00024500
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7        XLO=XLO + DR(L)*DR(L)                                          00024600
   7        I=LM(L,N)                                                      00024700
   7        D(L)=DR(L)                                                     00024800
   7        IF (I) 702,702,704                                             00024900
   3    704 D(L)=D(L) - X(I)                                               00025000
   3    702 J=LM(L+3,N)                                                    00025100
   7        IF (J) 720,720,708                                             00025200
   3    708 D(L)=D(L) + X(J)                                               00025300
   3    720 XL2=XL2 + D(L)*D(L)                                            00025400
   7        XO=DSQRT(XLO)                                                  00025500
   7        XL=DSQRT(XL2)                                                  00025600
   1  C                                                                    00025700
   1  C     FIND STRESS CORRESPONDING TO THIS STRAIN LEVEL                 00025800
   1  C                                                                    00025900
   7        IP=NCON/2                                                      00026000
   7        EPST=(XL-XO)/XO+STRAI(MTYPE)                                   00026100
   7        CALL MATBAR (E,AREA,MODEL,IP,N,MTYPE,XL,EPST,P,EAL,STR,PINT,   00026200
   6       1             PROP(1,MTYPE),PROP(1+IP,MTYPE))                  00026300
   1  C                                                                    00026400
   3    725 IF (INDNL.NE.1) GO TO 729                                      00026500
   7        DO 726 L=1,3                                                   00026600
   3    726 D(L)=DR(L)/XO                                                  00026700
   7        GO TO 732                                                      00026800
   3    729 DO 730 L=1,3                                                   00026900
   3    730 D(L)=D(L)/XL                                                   00027000
   3    732 DO 740 L=1,3                                                   00027100
   7        I=LM(L,N)                                                      00027200
   7        IF (I) 735,735,736                                             00027300
   3    736 HT(I)=HT(I) + P*D(L)                                           00027400
   3    735 J=LM(L+3,N)                                                    00027500
   7        IF (J) 740,740,737                                             00027600
   3    737 HT(J)=HT(J) - P*D(L)                                           00027700
   3    740 CONTINUE                                                       00027800
   3    742 IF (ICOUNT-2) 745,745,710                                      00027900
   1  C                                                                    00028000
   1  C     ADD ELEMENT STIFFNESS                                          00028100
   1  C                                                                    00028200
   3    745 IF (IREF) 710,750,710                                          00028300
   3    750 C1=D(1)                                                        00028400
   7        C2=D(2)                                                        00028500
   7        C3=D(3)                                                        00028600
   7        D1=C1*C1                                                       00028700
   7        D2=C2*C2                                                       00028800
   7        D3=C3*C3                                                       00028900
   7        B=P/XL                                                         00029000
   7        IF (INDNL.EQ.1) B=0.0                                          00029100
   7        S(1)=EAL*D1    + B                                             00029200
   7        S(2)=EAL*C1*C2                                                 00029300
   7        S(3)=EAL*C1*C3                                                 00029400
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7        S(7)=EAL*D2    + B                                              00029500
   7        S(8)=EAL*C2*C3                                                  00029600
   7        S(12)=EAL*D3   + B                                              00029700
   7        DO 770 I=1,3                                                    00029800
   3    770 S(I+3)=-S(I)                                                    00029900
   7        S(9)=-S(2)                                                      00030000
   7        S(10)=-S(7)                                                     00030100
   7        S(11)=-S(8)                                                     00030200
   7        S(13)=-S(3)                                                     00030300
   7        S(14)=-S(8)                                                     00030400
   7        S(15)=-S(12)                                                    00030500
   7        DO 780 I=1,3                                                    00030600
   7        S(15+I)=S(I)                                                    00030700
   3    780 S(18+I)=S(6+I)                                                  00030800
   7        S(21)=S(12)                                                     00030900
   1  C                                                                     00031000
   7        CALL ADDBAN (A(N4),A(N1),S,LM(1,N),ND)                          00031100
   1  C                                                                     00031200
   3    710 CONTINUE                                                        00031300
   7        RETURN                                                          00031400
   1  C                                                                     00031500
   1  C                                                                     00031600
   1  C     S T R E S S   C A L C U L A T I O N S                           00031700
   1  C                                                                     00031800
   1  C                                                                     00031900
   3    800 IF (NG.GT.NEGL) GO TO 805                                       00032000
   1  C                                                                     00032100
   1  C     GEOMETRICALLY AND MATERIALLY LINEAR STRESS CALCULATION          00032200
   1  C                                                                     00032300
   7        IPRNT=0                                                         00032400
   7        DO 830 N=1,NUME                                                 00032500
   7        IPST=IPS(N)                                                     00032600
   7        IF (IPST.EQ.0) GO TO 830                                        00032700
   7        IPRNT=IPRNT + 1                                                 00032800
   7        IF (IPRNT.NE.1) GO TO 801                                       00032900
   7        WRITE(KIO6,2010) NG                                             00033000
   3    801 MTYPE=MATP(N)                                                   00033100
   7        XL2=0.                                                          00033200
   7        DO 820 L=1,3                                                    00033300
   7        D(L) = XYZ(L,N) - XYZ(L+3,N)                                    00033400
   3    820 XL2=XL2+D(L)*D(L)                                               00033500
   7        DO 814 L=1,3                                                    00033600
   7        ST(L)=(D(L)/XL2)*E(MTYPE)                                       00033700
   3    814 ST(L+3)=-ST(L)                                                  00033800
   7        STR=STRAI(MTYPE)*E(MTYPE)                                       00033900
   7        DO 806 L=1,3                                                    00034000
   7        I=LM(L,N)                                                       00034100
   7        IF (I.LE.0) GO TO 807                                           00034200
   7        STR=STR+ST(L)*X(I)                                              00034300
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   3    807 J=LM(L+3,N)                                                    00034400
   7        IF (J.LE.0) GO TO 806                                          00034500
   7        STR=STR+ST(L+3)*X(J)                                           00034600
   3    806 CONTINUE                                                       00034700
   7        P=STR*AREA(MTYPE)                                              00034800
   7        P=P + PINIT(N)                                                 00034900
   7        STR=STR + (PINIT(N)/AREA(MTYPE))                               00035000
   7        WRITE(KIO6,2020) N,P,STR                                       00035100
   3    830 CONTINUE                                                       00035200
   1  C                                                                    00035300
   7        RETURN                                                         00035400
   1  C                                                                    00035500
   1  C                                                                    00035600
   1  C     GEOMETRICALLY NONLINEAR STRESS CALCULATION                     00035700
   1  C                                                                    00035800
   3    805 IPRNT=0                                                        00035900
   7        DO 810 N=1,NUME                                                00036000
   7        IPST=IPS(N)                                                    00036100
   7        IF (IPST.EQ.0) GO TO 810                                       00036200
   7        IPRNT=IPRNT + 1                                                00036300
   7        IF (IPRNT.NE.1) GO TO 803                                      00036400
   7        WRITE (6,2010) NG                                              00036500
   3    803 MTYPE=MATP(N)                                                  00036600
   7        PINT=PINIT(N)                                                  00036700
   7        XLO=0.                                                         00036800
   7        XL2=0.                                                         00036900
   7        DO 825 L=1,3                                                   00037000
   7        D(L)=XYZ(L,N) - XYZ(L+3,N)                                     00037100
   7        XLO=XLO + D(L)*D(L)                                            00037200
   7        I=LM(L,N)                                                      00037300
   7        IF (I) 802,802,804                                             00037400
   3    804 D(L)=D(L) + X(I)                                               00037500
   3    802 J=LM(L+3,N)                                                    00037600
   7        IF (J) 825,825,808                                             00037700
   3    808 D(L)=D(L) - X(J)                                               00037800
   3    825 XL2=XL2 + D(L)*D(L)                                            00037900
   7        XO=DSQRT(XLO)                                                  00038000
   7        XL=DSQRT(XL2)                                                  00038100
   7        IP=NCON/2                                                      00038200
   7        EPST=(XL-XO)/XO +STRAI(MTYPE)                                  00038300
   7        CALL MATBAR (E,AREA,MODEL,IP,N,MTYPE,XL,EPST,P,EAL,STR,PINT,   00038400
   6       1             PROP(1,MTYPE),PROP(1+IP,MTYPE))                  00038500
   7        WRITE(KIO6,2020) N,P,STR                                       00038600
   1  C                                                                    00038700
   3    810 CONTINUE                                                       00038800
   1  C                                                                    00038900
   7        RETURN                                                         00039000
   1  C                                                                    00039100
   2   1000 FORMAT (I5)                                                    00039200
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   2   1001 FORMAT (5F10.0)                                                          00039300
   2   1004 FORMAT (4I5,F10.0,2I5)                                                   00039400
   2   1005 FORMAT (8F10.0)                                                          00039500
   2   1006 FORMAT (3F10.0)                                                          00039600
   2   2000 FORMAT (42H M A T E R I A L   D E F I N I T I O N     ///,               00039700
   6        1        16H MATERIAL MODEL.,12(2H .),16H( NPAR(15)). . =,I5/,          00039800
   6        2        40H    EQ.1,   LINEAR ELASTIC                    /,           00039900
   6        3        40H    EQ.2,   NONLINEAR ELASTIC                 /,           00040000
   6        4        42H            (STRESS-STRAIN LAW SPECIFIED)     //,          00040100
   6        5        37H NUMBER OF DIFFERENT SETS OF MATERIAL          /,           00040200
   6        6        14H     CONSTANTS,13(2H .),16H( NPAR(16)). . =,I5//,          00040300
   6        7        40H NUMBER OF MATERIAL CONSTANTS PER SET. .  ,                00040400
   6        8        16H( NPAR(17)). . =,I5)                                       00040500
   2   2001 FORMAT (///1X,4HTYPE,14X,1HE,10X,5HAREA ,12X,3HDEN,10X,5HSTRAI)         00040600
   2   2002 FORMAT (/I5,5E15.7)                                                      00040700
   2   2004 FORMAT (4I6,E25.5,I10)                                                   00040800
   2   2005 FORMAT (///40H E L E M E N T   I N F O R M A T I O N  ///,              00040900
   6        1           49H     N     I     J  TYPE       PRESTRESSING FORCE,      00041000
   6        2           10H       IPS/)                                           00041100
   2   2006 FORMAT (///28H MATERIAL CONSTANTS SET NO.  ,I5//                         00041200
   6        1           10H     POINT,15H         STRAIN,15H         STRESS )      00041300
   2   2007 FORMAT (I10,2E15.4)                                                      00041400
   2   2008 FORMAT (/20H                  AREA,20H              DENSITY,           00041500
   6        1        20H      INITIAL STRAIN//3E20.7)                             00041600
   2   2010 FORMAT (////,75H S T R E S S   C A L C U L A T I O N S   F O R         00041700
   6       1E L E M E N T   G R O U P  ,I5,15H  ( TRUSSES )  //,4X,               00041800
   6       27HELEMENT,8X,5HFORCE,12X,6HSTRESS /)                                  00041900
   2   2020 FORMAT (4X,I5,6X,E13.6,4X,E13.6)                                         00042000
   2   2030 FORMAT (36H E L E M E N T   D E F I N I T I O N ///,                     00042100
   6        1        14H ELEMENT TYPE ,13(2H .),16H( NPAR(1) ). . =,I5/,           00042200
   6        2        25H     EQ.1, TRUSS ELEMENTS/,                               00042300
   6        3        25H     EQ.2, 2-DIM ELEMENTS/,                               00042400
   6        3        25H     EQ.3, 3-DIM ELEMENTS//,                              00042500
   6        4        20H NUMBER OF ELEMENTS.,10(2H .),16H( NPAR(2) ). . =,        00042600
   6        5        I5//,                                                        00042700
   6        6        28H TYPE OF NONLINEAR ANALYSIS.,6(2H .),15H( NPAR(3) ). .    00042800
   6        7        1H=,I5/,                                                     00042900
   6        8        38H     EQ.1, MATERIALLY NONLINEAR ONLY   /,                 00043000
   6        8        45H     EQ.2, UPDATED LAGRANGIAN FORMULATION         ///)    00043100
   1   C                                                                          00043200
   7         END                                                                  00043300
```

```
START                                                                                                  MOD
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8  FLAGS

   7        SUBROUTINE SECANT (A,AVEC,B,V,MAXA,W,WW,ROOT,TIM,ERRVL,ERRVR,NITE,00000100
   6       1                   N,MA,NWK,NWM,NROOT,NC,IFPR,ANORM,COFQ,IEX)     00000200
   7        IMPLICIT REAL*8(A-H,O-Z)                                          00000300
   7        INTEGER MAXA(1),NITE(1)                                           00000400
   1  C                                                                       00000500
   1  C                                                                       00000600
   1  C     P R O G R A M  T O  F I N D  E I G E N S Y S T E M                00000700
   1  C                                                                       00000800
   1  C        EXECUTION MODES                                                00000900
   1  C*         IEX=1     FIND SMALLEST EIGENVALUES                         00001000
   1  C          IEX=2     FIND LOWER BOUND ON EIGENVALUES                   00001100
   1  C*         IEX=3     FIND LARGEST EIGENVALUES                          00001200
   1  C*   FOR NUMERICAL EFFICIENCY,UPPER AND LOWER BOUND TOLERENCES          00001300
   1  C    ARE DEFINED.(S.N.MUKHERJEE,BBC,BADEN. AUG.1976.)                  00001400
   7        DIMENSION A(1),AVEC(N,1),B(1),V(1),W(1),WW(N,1),ROOT(1),          00001500
   6       1  TIM(1),ERRVL(1),ERRVR(1)                                       00001600
   7        COMMON/BBCCIO/KIO1,KIO2,KIO3,KIO4,KIO5,KIO6,KIO7,KIO8,KIO9,       00001700
   6       1KIO10,KIO11,KIO12,KIO13,KIO14,KIO15,KIO16,KIO17,KIO18,KIO19,KIO20,00001800
   6       2KIO21,KIO22,KIO23,KIO24,KIO25,KIO26,KIO27,KIO28,KIO29,KIO30       00001900
   7        COMMON /TAPEIG/  NSTIF,NT                                         00002000
   1  C                                                                       00002100
   1  C                                                                       00002200
   1  C     FOLLOWING TOLERANCES ARE SET FOR THE IBM 370 (S.N.MUKHERJEE.BBC,  00002300
   1  C     BADEN.AUG.1976.                                                   00002400
   7        ACTOL=1.0E-04                                                     00002500
   7        RCBTOL=1.E-05                                                     00002600
   7        RITOL=1.E-06                                                      00002700
   7        RTOL=1.0E-10                                                      00002800
   7        RQTOL=1.0E-12                                                     00002900
   1  C                                                                       00003100
   7        NTF=5                                                             00003200
   7        IITEM=10                                                          00003300
   7        NITEM=39                                                          00003400
   7        NITEMM=59                                                         00003500
   7        NVM=6                                                             00003600
   1  C                                                                       00003700
   7        REWIND NT                                                         00003800
   1  C                                                                       00003900
   7        NOV=0                                                             00004100
   7        JR=1                                                              00004200
   7        NSK=0                                                             00004300
   7        ISC=1000                                                          00004400
   1  C                                                                       00004500
   1  C     FIND LOCATIONS FOR NEGATIVE ELEMENTS IN STARTING                  00004600
   1  C     ITERATION VECTORS                                                 00004700
   1  C                                                                       00004800
   7        REWIND NSTIF                                                      00004900
   7        READ (NSTIF) (A(I),I=1,NWK)                                       00005000
   7        NC1=NC + 1                                                        00005100
```

```
START                                                                                                                  MOD
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8  FLAGS

   7        DO 1 I=1,N                                                                                        00005200
   7        II=MAXA(I)                                                                                        00005300
   7        AA=A(II)                                                                                          00005400
   7        IF (AA.GT.0.0) GO TO 4                                                                            00005500
   7        WRITE(KIO6,1000) I,AA                                                                             00005600
   7        STOP                                                                                              00005700
   5     4  IJ=I                                                                                              00005800
   7        IF (NWM.EQ.NWK) IJ=II                                                                             00005900
   5     1  V(I)=B(IJ)/AA                                                                                     00006000
   7        DO 2 J=3,NC1                                                                                      00006100
   7        IMAX=0                                                                                            00006200
   7        RMAX=0.                                                                                           00006300
   7        DO 3 I=1,N                                                                                        00006400
   7        IF (V(I).LT.RMAX) GO TO 3                                                                         00006500
   7        RMAX=V(I)                                                                                         00006600
   7        IMAX=I                                                                                            00006700
   5     3  CONTINUE                                                                                          00006800
   7        NITE(J)=IMAX                                                                                      00006900
   5     2  V(IMAX)=0.                                                                                        00007000
   1  C                                                                                                       00007100
   1  C     CHECK FOR SINGLE DEGREE-OF-FREEDOM SYSTEM                                                         00007200
   1  C                                                                                                       00007300
   7        IF (N.GT.1) GO TO 5                                                                               00007400
   7        IF (B(1).GT.0) GO TO 7                                                                            00007500
   7        WRITE(KIO6,1180)                                                                                  00007600
   7        STOP                                                                                              00007700
   5     7  ROOT(1)=A(1)/B(1)                                                                                 00007800
   7        NSCH=1                                                                                            00007900
   7        A(1)=1./DSQRT(B(1))                                                                               00008000
   7        GO TO 950                                                                                         00008100
   1  C                                                                                                       00008200
   1  5     CALL TTIME (TIM1)                                                                                 00008300
   7        RA=0.0                                                                                            00008400
   7        RR=0.0                                                                                            00008500
   7        KA=0                                                                                              00008600
   7        KB=0                                                                                              00008700
   7        KR=0                                                                                              00008800
   7        CALL BANDET(A,B,V,MAXA,N,MA,NWK,NWM,RA,NSCH,DETA,ISC,1)                                           00008900
   7        FA=DETA                                                                                           00009000
   7        FR=FA                                                                                             00009100
   7        DETR=DETA                                                                                         00009200
   1  C                                                                                                       00009300
   1  C   CHECK FOR ZERO EIGENVALUE(S)                                                                        00009400
   1  C                                                                                                       00009500
   7        N1=MAXA(N)                                                                                        00009600
   7        IF(A(N1).GT.ANORM) GO TO 10                                                                       00009700
   7        WRITE(KIO6,1009)                                                                                  00009800
   7        STOP                                                                                              00009900
   1  C                                                                                                       00010000
```

```
START                                                                                                    MOD
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8                    FLAGS
   1  C   FIND LOWER BOUND ON SMALLEST EIGENVALUE                                                        00010100
   1  C                                                                                                  00010200
   4     10 IF (IFPR.EQ.1)                                                                             00010300
   6       *WRITE(KIO6,1010)                                                                           00010400
   7        IF(NWM.EQ.N) GO TO 99                                                                      00010500
   7        DO 98 I=1,N                                                                                00010600
   4     98 V(I)=1.                                                                                    00010700
   7        CALL MLTPLY(W,B,V,MAXA,N)                                                                  00010800
   7        GO TO 101                                                                                  00010900
   4     99 DO 100 I=1,N                                                                               00011000
   3    100 W(I)=B(I)                                                                                  00011100
   3    101 RT=0.0                                                                                     00011200
   7        IITE=0                                                                                     00011300
   3    110 IITE=IITE+1                                                                                00011400
   7        DO 120 I=1,N                                                                               00011500
   3    120 V(I)=W(I)                                                                                  00011600
   7        CALL BANDET (A,B,V,MAXA,N,MA,NWK,NWM,RA,NSCH,DETA,ISC,2)                                   00011700
   7        RQT=0.0                                                                                    00011800
   7        DO 130 I=1,N                                                                               00011900
   3    130 RQT=RQT+W(I)*V(I)                                                                          00012000
   7        IF (NWM-N) 179,179,178                                                                     00012100
   3    178 CALL MLTPLY (W,B,V,MAXA,N)                                                                 00012200
   7        GO TO 181                                                                                  00012300
   3    179 DO 180 I=1,N                                                                               00012400
   3    180 W(I)=B(I)*V(I)                                                                             00012500
   3    181 RQB=0.0                                                                                    00012600
   7        DO 140 I=1,N                                                                               00012700
   3    140 RQB=RQB+W(I)*V(I)                                                                          00012800
   7        RQ=RQT/RQB                                                                                 00012900
   7        IF (IFPR.EQ.1)                                                                             00013000
   6       * WRITE(KIO6,1004) RQ                                                                       00013100
   7        BS=DSQRT(RQB)                                                                              00013200
   7        TOL=DABS(RQ-RT)/RQ                                                                         00013300
   7        IF (TOL.LT.RCBTOL) GO TO 150                                                               00013400
   7        DO 160 I=1,N                                                                               00013500
   3    160 W(I)=W(I)/BS                                                                               00013600
   7        RT=RQ                                                                                      00013700
   7        IF (IITE.LT.IITEM) GO TO 110                                                               00013800
   1  C                                                                                                00013900
   3    150 RB=RQ*(1.0D0-DMIN1(0.1D0,100*TOL))                                                         00014000
   7        IS=0                                                                                       00014100
   3    230 CALL BANDET(A,B,V,MAXA,N,MA,NWK,NWM,RB,NSCH,DETB,ISC,1)                                    00014200
   7        IF (IFPR.EQ.1)                                                                             00014300
   6       *WRITE(KIO6,1020) RB,NSCH                                                                   00014400
   7        FB=DETB                                                                                    00014500
   1  C     IF (NSCH.EQ.0) GO TO 300  (NOT NECESSARY)                                                  00014600
   7        IF (NSCH.EQ.0) GO TO 299                                                                   00014700
   7        IS=IS+1                                                                                    00014800
   7        IF (IS.LE.NTF) GO TO 240                                                                   00014900
```

```
START                                                                                                  MOD
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8  FLAGS
   7        WRITE(KIO6,1030)                                                          00015000
   7        STOP                                                                      00015100
   3    240 RB=RB/(NSCH+1)                                                            00015200
   7        GO TO 230                                                                 00015300
   1  C                                                                               00015400
   1  C                                                                               00015500
   1  C     I T E R A T I O N  F O R  I N D I V I D U A L  E I G E N P A I R S       00015600
   1  C                                                                               00015700
   1  C                                                                               00015800
   3    299 ETA=2.D0                                                                  00015900
   1  C                                                                               00016000
   3    300 IF (IEX.EQ.2) ROOT(1)=RB                                                  00016100
   7        IF (IEX.EQ.2) RETURN                                                      00016200
   7        IF (IFPR.EQ.1)                                                            00016300
   6       *WRITE(KIO6,1040)                                                          00016400
   7        NITE(JR)=-1                                                               00016500
   7        IF (IFPR.EQ.1)                                                            00016600
   6       *WRITE(KIO6,1050)                                                          00016700
   6        *                  JR,NITE(JR),RA,DETA,FA,ETA,ISC                        00016800
   7        NITE(JR)=0                                                                00016900
   7        IF (IFPR.EQ.1)                                                            00017000
   6        * WRITE(KIO6,1050)                                                       00017100
   6        *                  JR,NITE(JR),RB,DETB,FB,ETA,ISC                        00017200
   1  C                                                                               00017300
   1  C   WE STOP WHEN WE HAVE THE REQUIRED NO OF ROOTS SMALLER THAN RC AND          00017400
   1  C   NOV=0                                                                      00017500
   1  C                                                                               00017600
   3    310 IF (NSCH.GE.NROOT) GO TO 900                                            00017700
   7        IF (RB.GT.COFQ) GO TO 900                                                 00017800
   7        IF (KB-KA) 301,303,302                                                    00017900
   3    301 FB=FB*1.D50                                                             00018000
   7        KB=KB+1                                                                   00018100
   1  C                                                                               00018200
   7        GO TO 303                                                                 00018300
   3    302 FA=FA*1.D50                                                             00018400
   7        KA=KA+1                                                                   00018500
   1  C     DIF=FB-FA                                                                 00018600
   3    303 DIF = FB-FA                                                             00018700
   7        IF (DIF.NE.0.0) GO TO 320                                                 00018800
   7        WRITE(KIO6,1060)                                                          00018900
   7        GO TO 900                                                                 00019000
   3    320 DEL=FB*(RB-RA)/DIF                                                      00019100
   7        RC=RB-ETA*DEL                                                             00019200
   7        TOL=RCBTOL*RC                                                             00019300
   7        IF (DABS(RC-RB).GT.TOL) GO TO 330                                         00019400
   7        IF (IFPR.EQ.1)                                                            00019500
   6       *WRITE(KIO6,1070)                                                          00019600
   7        ROOT(JR)=RB                                                               00019700
   7        GO TO 400                                                                 00019800
```

```
START                                                                                                  MOD
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8               FLAGS
   1  C                                                                                          00019900
   3    330 CALL BANDET(A,B,V,MAXA,N,MA,NWK,NWM,RC,NSCH,DETC,ISC,1)                              00020000
   7        FC=DETC                                                                            00020100
   7        KC=0                                                                               00020200
   7        NITE(JR)=NITE(JR)+1                                                                00020300
   7        IF (JR.EQ.1) GO TO 340                                                             00020400
   7        JJ=JR-1                                                                            00020500
   7        DO 350 K=1,JJ                                                                      00020600
   7        IF (DABS(FC).GT.1.D-50) GO TO 350                                                  00020700
   7        FC = FC*1.D50                                                                      00020800
   7        KC = KC+1                                                                          00020900
   3    350 FC=FC/(RC-ROOT(K))                                                                 00021000
   3    340 IF (IFPR.EQ.1)                                                                     00021100
   6       *WRITE(KIO6,1050)                                                                   00021200
   6       *                JR,NITE(JR),RC,DETC,FC,ETA,ISC                                     00021300
   1  C                                                                                        00021400
   1  C   IF WE HAVE MORE SIGNCHANGES THAN EIGENVALUES SMALLER THAN RC WE                      00021500
   1  C   START INVERSE ITERATION                                                             00021600
   1  C                                                                                        00021700
   7        NES=0                                                                            00021800
   7        IF (JR.EQ.1) GO TO 380                                                           00021900
   7        DO 360 I=1,JJ                                                                    00022000
   3    360 IF (ROOT(I).LT.RC) NES=NES+1                                                     00022100
   3    380 NOV=NSCH-NES                                                                     00022200
   7        IF (NOV.EQ.0) GO TO 370                                                          00022300
   7        IF (IFPR.EQ.1)                                                                   00022400
   6       *WRITE(KIO6,1080) NOV                                                             00022500
   7        ROOT(JR)=RC                                                                      00022600
   7        IF (NOV.GT.1) NSK=1                                                              00022700
   1  C                                                                                      00022800
   7        GO TO 400                                                                        00022900
   3    370 RR=RA                                                                            00023000
   7        FR=FA                                                                            00023100
   7        DETR=DETA                                                                        00023200
   7        RA=RB                                                                            00023300
   7        FA=FB                                                                            00023400
   7        DETA=DETB                                                                        00023500
   7        RB=RC                                                                            00023600
   7        FB=FC                                                                            00023700
   7        DETB=DETC                                                                        00023800
   7        KR=KA                                                                            00023900
   7        KA=KB                                                                            00024000
   7        KB=KC                                                                            00024100
   1  C                                                                                      00024200
   1  C     WE RESET ETA IF WE CAN ACCELERATE THE SECANT ITERATION STILL MORE                 00024300
   1  C                                                                                      00024400
   7        TOL=RB*ACTOL                                                                     00024500
   7        IF (DABS(RA-RB).LT.TOL)ETA=ETA*2                                                 00024600
   7        IF (NITE(JR).LE.NITEM) GO TO 310                                                 00024700
```

```
START                                                                                                    MOD
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8  FLAGS
    7         WRITE(KIO6,1015) JR,NITE(JR)                                            00024800
    7         GO TO 900                                                               00024900
    1   C                                                                             00025000
    1   C   CHECK FOR STORAGE                                                         00025100
    1   C                                                                             00025200
    3     400 IF (JR.LE.NC) GO TO 405                                                 00025300
    7         WRITE(KIO6,1090)                                                        00025400
    7         GO TO 900                                                               00025500
    1   C                                                                             00025600
    1   C     INITIALIZE STARTING INVERSE ITERATION VECTOR                            00025700
    1   C                                                                             00025800
    3     405 NOR=JR-1                                                                00025900
    7         IF (NOR.GT.NVM) NOR=NVM                                                 00026000
    7         CALL TTIME (TIM3)                                                       00026100
    7         IF (IFPR.EQ.1)                                                          00026200
    6        *WRITE(KIO6,1100)NOR                                                     00026300
    7         IF (JR.EQ.1) GO TO 410                                                  00026400
    7         DO 420 I=1,N                                                            00026500
    3     420 V(I)=1.0                                                                00026600
    7         I=NITE(JR+1)                                                            00026700
    7         V(I)=-1.0                                                               00026800
    3     410 IF (NWM-N) 429,429,428                                                  00026900
    3     428 CALL MLTPLY (W,B,V,MAXA,N)                                              00027000
    7         GO TO 431                                                               00027100
    3     429 DO 430 I=1,N                                                            00027200
    3     430 W(I)=B(I)*V(I)                                                          00027300
    3     431 IS=0                                                                    00027400
    7         GO TO 510                                                               00027500
    1   C                                                                             00027600
    1   C     INVERSE ITERATION                                                       00027700
    1   C                                                                             00027800
    3     440 NITE(JR)=NITE(JR)+1                                                     00027900
    7         DO 450 I=1,N                                                            00028000
    3     450 V(I)=W(I)                                                               00028100
    7         CALL BANDET (A,B,V,MAXA,N,MA,NWK,NWM,RC,NSCH,DETC,ISC,2)                00028200
    7         IF (IS.EQ.1) GO TO 460                                                  00028300
    7         RQT=0.0                                                                 00028400
    7         DO 470 I=1,N                                                            00028500
    3     470 RQT=RQT+W(I)*V(I)                                                       00028600
    7         IF (NWM-N) 474,474,473                                                  00028700
    3     473 CALL MLTPLY (W,B,V,MAXA,N)                                              00028800
    7         GO TO 476                                                               00028900
    3     474 DO 475 I=1,N                                                            00029000
    3     475 W(I)=B(I)*V(I)                                                          00029100
    3     476 RQB=0.0                                                                 00029200
    7         DO 480 I=1,N                                                            00029300
    3     480 RQB=RQB+W(I)*V(I)                                                       00029400
    7         RQ=RQT/RQB                                                              00029500
    7         RT=ROOT(JR)+RQ                                                          00029600
```

```
START                                                                                                                  MOD
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8  FLAGS

   7        IF (IFPR.EQ.1)                                                                 00029700
   6       *WRITE(KIO6,1110)                                                               00029800
   6       *                    JR,NITE(JR),RT,RQ                                          00029900
   7        TOL=RT*RQTOL                                                                   00030000
   7        EIGDIF=DABS(RT-RTA)                                                            00030100
   7        IF (EIGDIF.GT.TOL) GO TO 510                                                   00030200
   7        IS=1                                                                           00030300
   7        GO TO 440                                                                      00030400
   1  C                                                                                    00030500
   3    510 RTA=RT                                                                         00030600
   7        BS=DSQRT(RQB)                                                                  00030700
   7        IF (NOR.EQ.0) GO TO 545                                                        00030800
   7        DO 520 K=1,NOR                                                                 00030900
   7        AL=0.0                                                                         00031000
   7        DO 530 I=1,N                                                                   00031100
   3    530 AL=AL + WW(I,K)*V(I)                                                           00031200
   7        DO 540 I=1,N                                                                   00031300
   3    540 W(I)=W(I)-AL*WW(I,K)                                                           00031400
   3    520 CONTINUE                                                                       00031500
   3    545 DO 490 I=1,N                                                                   00031600
   3    490 W(I)=W(I)/BS                                                                   00031700
   1  C                                                                                    00031800
   1  C      PERFORM RAYLEIGH QUOTIENT SHIFT IF CONVERGENCE IS SLOW                        00031900
   1  C                                                                                    00032000
   7        IF (NITE(JR).LE.NITEM) GO TO 440                                               00032100
   7        IF (NITE(JR).GT.(NITEM+1)) GO TO 552                                           00032200
   7        TOL=RT*RITOL                                                                   00032300
   7        IF (EIGDIF.LT.TOL) GO TO 554                                                   00032400
   7        WRITE(KIO6,1015)  JR,NITEM                                                     00032500
   7        GO TO 900                                                                      00032600
   3    554 IF (IFPR.EQ.1)                                                                 00032700
   6       *WRITE(KIO6,1014) JR                                                            00032800
   7        WRITE (NSTIF) (A(I),I=1,NWK)                                                   00032900
   7        CALL BANDET (A,B,V,MAXA,N,MA,NWK,NWM,RT,NSCHT,DETT,ISC,1)                      00033000
   7        ROOT(JR)=RT                                                                    00033100
   7        GO TO 440                                                                      00033200
   3    552 IF (NITE(JR).LE.NITEMM) GO TO 440                                              00033300
   7        WRITE(KIO6,1015) JR,NITE(JR)                                                   00033400
   7        GO TO 900                                                                      00033500
   1  C                                                                                    00033600
   3    460 RQT=0.0                                                                        00033700
   7        ERRT=RQB                                                                       00033800
   7        DO 570 I=1,N                                                                   00033900
   3    570 RQT=RQT+V(I)*W(I)                                                              00034000
   7        IF (NWM-N) 559,559,564                                                         00034100
   3    564 CALL MLTPLY (W,B,V,MAXA,N)                                                     00034200
   7        GO TO 561                                                                      00034300
   3    559 DO 560 I=1,N                                                                   00034400
   3    560 W(I)=B(I)*V(I)                                                                 00034500
```

```
START                                                                                                    MOD
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8  FLAGS
   3    561 RQB=0.0                                                                      00034600
   7        DO 580 I=1,N                                                                 00034700
   3    580 RQB=RQB+V(I)*W(I)                                                            00034800
   1  C                                                                                  00034900
   1  C     OBTAIN ERROR BOUNDS -                                                        00035000
   1  C     THE BOUNDS DEPEND ON THE DISTANCE FROM THE ROOT AND CAN BE LARGE             00035100
   1  C                                                                                  00035200
   7        RQ=RQT/RQB                                                                   00035300
   7        ROOT(JR)=ROOT(JR)+RQ                                                         00035400
   7        ERR=DSQRT(ERRT/RQB)                                                          00035500
   7        ERRVL(JR)=ROOT(JR)-ERR                                                       00035600
   7        ERRVR(JR)=ROOT(JR)+ERR                                                       00035700
   1  C                                                                                  00035800
   7        BS=DSQRT(RQB)                                                                00035900
   7        DO 590 I=1,N                                                                 00036000
   7        W(I)=W(I)/BS                                                                 00036100
   3    590 V(I)=V(I)/BS                                                                 00036200
   7        WRITE (NT) (V(I),I=1,N)                                                      00036300
   7        JJ=JR                                                                        00036400
   7        IF (JJ.LE.NVM) GO TO 610                                                     00036500
   7        DO 600 K=1,N                                                                 00036600
   7        DO 600 L=2,NVM                                                               00036700
   3    600 WW(K,L-1)=WW(K,L)                                                            00036800
   7        JJ=NVM                                                                       00036900
   3    610 DO 620 K=1,N                                                                 00037000
   3    620 WW(K,JJ)=W(K)                                                                00037100
   1  C                                                                                  00037200
   7        CALL TTIME (TIM2)                                                            00037300
   7        TIM3=TIM2-TIM3                                                               00037400
   7        IF (IFPR.EQ.1)                                                               00037500
   6       *WRITE(KIO6,1120) TIM3                                                        00037600
   7        TIM(JR)=TIM2-TIM1                                                            00037700
   7        TIM1=TIM2                                                                    00037800
   1  C                                                                                  00037900
   1  C     READ TRIANGULARIZED MATRIX IN CASE RAYLEIGH QUOTIENT SHIFT                   00038000
   1  C     WAS PERFORMED                                                                00038100
   1  C                                                                                  00038200
   7        IF (NITE(JR).LE.(NITEM+2)) GO TO 680                                         00038300
   7        BACKSPACE NSTIF                                                              00038400
   7        READ (NSTIF) (A(I),I=1,NWK)                                                  00038500
   1  C                                                                                  00038600
   1  C     DECIDE STRATEGY FOR ITERATION TOWARDS NEXT ROOT                              00038700
   1  C                                                                                  00038800
   3    680 TOL=RTOL*ROOT(JR)                                                            00038900
   7        IF (NOV.GT.0) GO TO 700                                                      00039000
   7        IF (DABS(ROOT(JR)-RB).GT.TOL)GO TO 710                                       00039100
   7        IF (RA.GT.0.0) GO TO 720                                                     00039200
   7        RA=RB/2.                                                                     00039300
   7        CALL BANDET(A,B,V,MAXA,N,MA,NWK,NWM,RA,NSCH,DETA,ISC,1)                      00039400
```

```
START                                                                                                                  MOD
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8  FLAGS
   7          FA=DETA                                                                    00039500
   7          KA=0                                                                       00039600
   3    720   RB=RA                                                                      00039700
   7          FB=FA                                                                      00039800
   7          KB = KA                                                                    00039900
   7          KA=KR                                                                      00040000
   7          DETB=DETA                                                                  00040100
   7          RA=RR                                                                      00040200
   7          FA=FR                                                                      00040300
   7          DETA=DETR                                                                  00040400
   7          GO TO 710                                                                  00040500
   1  C                                                                                  00040600
   3    700   IF (ROOT(JR).GT.RC) NSK=1                                                  00040700
   7          IF (NSK.EQ.1) GO TO 730                                                    00040800
   7          IF (DABS(RC-ROOT(JR)).LT.TOL)GO TO 740                                     00040900
   7          IF (DABS(ROOT(JR)-RB).LT.TOL)GO TO 750                                     00041000
   7          RA=RB                                                                      00041100
   7          FA=FB                                                                      00041200
   7          DETA=DETB                                                                  00041300
   7          KA=KB                                                                      00041400
   3    750   RB=RC                                                                      00041500
   7          FB=FC                                                                      00041600
   7          KB=KC                                                                      00041700
   7          DETB=DETC                                                                  00041800
   7          GO TO 710                                                                  00041900
   3    740   IF(DABS(ROOT(JR)-RB).GT.TOL)GO TO 710                                      00042000
   7          IF (RA.GT.0.0) GO TO 760                                                   00042100
   7          RA=RB/2.                                                                   00042200
   7          CALL BANDET(A,B,V,MAXA,N,MA,NWK,NWM,RA,NSCH,DETA,ISC,1)                    00042300
   7          FA=DETA                                                                    00042400
   7          KA=0                                                                       00042500
   3    760   RB=RA                                                                      00042600
   7          FB=FA                                                                      00042700
   7          KB=KA                                                                      00042800
   7          KA=KR                                                                      00042900
   7          DETB=DETA                                                                  00043000
   7          RA=RR                                                                      00043100
   7          FA=FR                                                                      00043200
   7          DETA=DETR                                                                  00043300
   3    710   FA=FA/(RA-ROOT(JR))                                                        00043400
   7          FB=FB/(RB-ROOT(JR))                                                        00043500
   7          JR=JR+1                                                                    00043600
   7          IF (DABS(FA).GT.1.D-50) GO TO 711                                          00043700
   7          FA=FA*1.D50                                                                00043800
   7          KA=KA+1                                                                    00043900
   3    711   IF (DABS(FB).GT.1.D-50) GO TO 299                                          00044000
   7          FB=FB*1.D50                                                                00044100
   7          KB=KB+1                                                                    00044200
   1  C       ETA=2.0                                                                    00044300
```

```
START                                                                                                                  MOD
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8  FLAGS
   1  C     GO TO 300                                                                    00044400
   7        GO TO 299                                                                    00044500
   1  C                                                                                  00044600
   3    730 IF (RA.GT.0.0) GO TO 780                                                     00044700
   7        RA=RB/2.                                                                     00044800
   7        CALL BANDET(A,B,V,MAXA,N,MA,NWK,NWM,RA,NSCH,DETA,ISC,1)                      00044900
   7        FA=DETA                                                                      00045000
   7        KA=0                                                                         00045100
   3    780 IF (DABS(ROOT(JR)-RB).GT.TOL)GO TO 770                                       00045200
   7        RB=RA                                                                        00045300
   7        FB=FA                                                                        00045400
   7        KB=KA                                                                        00045500
   7        KA=KR                                                                        00045600
   7        DETB=DETA                                                                    00045700
   7        RA=RR                                                                        00045800
   7        FA=FR                                                                        00045900
   7        DETA=DETR                                                                    00046000
   3    770 FA=FA/(RA-ROOT(JR))                                                          00046100
   7        FB=FB/(RB-ROOT(JR))                                                          00046200
   7        FR=FR/(RR-ROOT(JR))                                                          00046300
   7        IF (DABS(FA).GT.1.D-50) GO TO 771                                            00046400
   7        FA =FA*1.D50                                                                 00046500
   7        KA=KA+1                                                                      00046600
   3    771 IF(DABS(FB).GT.1.D-50) GO TO 772                                             00046700
   7        FB = FB*1.D50                                                                00046800
   7        KB=KB+1                                                                      00046900
   3    772 IF(DABS(FR).GT.1.D-50) GO TO 773                                             00047000
   7        FR=FR*1.D50                                                                  00047100
   7        KR=KR+1                                                                      00047200
   1  C     IF (ROOT(JR).LE.RC) NOV=NOV-1                                                00047300
   3    773 IF (ROOT(JR).LE.RC) NOV=NOV-1                                                00047400
   7        JR=JR+1                                                                      00047500
   7        NITE(JR)=0                                                                   00047600
   7        ROOT(JR)=RC                                                                  00047700
   7        IF (NOV.GT.0) GO TO 400                                                      00047800
   7        NSK=0                                                                        00047900
   1  C     ETA=2.0                                                                      00048000
   1  C     GO TO 300                                                                    00048100
   7        GO TO 299                                                                    00048200
   1  C                                                                                  00048300
   3    900 NROOT=JR-1                                                                   00048400
   7        IF (NROOT.GT.0) GO TO 902                                                    00048500
   7        WRITE (6,1180)                                                               00048600
   7        STOP                                                                         00048700
   3    902 IF (IFPR.EQ.0) GO TO 905                                                     00048800
   7        WRITE(KIO6,1140)                                                             00048900
   7        WRITE(KIO6,1006)                                                             00049000
   6       *                (NITE(J),J=1,NROOT)                                        00049100
   7        WRITE(KIO6,1150)                                                             00049200
```

```
START                                                                    MOD
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8  FLAGS
   7        WRITE(KIO6,1008)                                          00049300
   6       *               (TIM(J),J=1,NROOT)                         00049400
   7        WRITE(KIO6,1160)                                          00049500
   7        WRITE(KIO6,1004)                                          00049600
   6       *               (ERRVL(J),J=1,NROOT)                       00049700
   7        WRITE(KIO6,1004)                                          00049800
   6       *               (ERRVR(J),J=1,NROOT)                       00049900
   1  C                                                               00050000
   1  C     CALCULATE PHYSICAL ERROR NORMS                            00050100
   1  C                                                               00050200
   3    905 REWIND NT                                                 00050300
   7        REWIND NSTIF                                              00050400
   7        READ (NSTIF) (A(I),I=1,NWK)                               00050500
   1  C                                                               00050600
   7        DO 904 L=1,NROOT                                          00050700
   7        RT=ROOT(L)                                                00050800
   7        READ (NT) (V(I),I=1,N)                                    00050900
   7        CALL MLTPLY (W,A,V,MAXA,N)                                00051000
   7        VNORM=0.0                                                 00051100
   7        DO 911 I=1,N                                              00051200
   3    911 VNORM=VNORM + W(I)*W(I)                                   00051300
   7        IF (NWM-N) 907,907,903                                    00051400
   3    903 CALL MLTPLY (WW,B,V,MAXA,N)                               00051500
   7        DO 906 I=1,N                                              00051600
   3    906 W(I)=W(I) - RT*WW(I,1)                                    00051700
   7        GO TO 908                                                 00051800
   3    907 DO 909 I=1,N                                              00051900
   3    909 W(I)=W(I) - RT*B(I)*V(I)                                  00052000
   3    908 WNORM=0.0                                                 00052100
   7        DO 910 I=1,N                                              00052200
   3    910 WNORM=WNORM + W(I)*W(I)                                   00052300
   7        VNORM=DSQRT(VNORM)                                        00052400
   7        WNORM=DSQRT(WNORM)                                        00052500
   7        ERRVL(L)=WNORM/VNORM                                      00052600
   3    904 CONTINUE                                                  00052700
   1  C                                                               00052800
   1  C   ARRANGE EIGENVALUES AND VECTORS IN ASCENDING ORDER          00052900
   1  C                                                               00053000
   7        REWIND NT                                                 00053100
   7        DO 912 L=1,NROOT                                          00053200
   3    912 READ (NT) (AVEC(I,L),I=1,N)                               00053300
   7        IF (JR.EQ.2) GO TO 950                                    00053400
   7        JR=JR-2                                                   00053500
   3    915 IS=0                                                      00053600
   7        DO 920 I=1,JR                                             00053700
   7        IF (ROOT(I+1).GE.ROOT(I)) GO TO 920                       00053800
   7        IS=IS+1                                                   00053900
   7        RT=ROOT(I+1)                                              00054000
   7        ROOT(I+1)=ROOT(I)                                         00054100
```

```
START COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8  MOD FLAGS
   7         ROOT(I)=RT                                                                    00054200
   7         RT=ERRVL(I+1)                                                                 00054300
   7         ERRVL(I+1)=ERRVL(I)                                                           00054400
   7         ERRVL(I)=RT                                                                   00054500
   7         DO 930 K=1,N                                                                  00054600
   7         RT=AVEC(K,I+1)                                                                00054700
   7         AVEC(K,I+1)=AVEC(K,I)                                                         00054800
   3     930 AVEC(K,I)=RT                                                                  00054900
   3     920 CONTINUE                                                                      00055000
   7         IF (IS.GT.0) GO TO 915                                                        00055100
   1   C                                                                                   00055200
   3     950 WRITE(KIO6,1170)                                                            00055300
   7         NROOT=NSCH                                                                    00055400
   7         WRITE(KIO6,1004)                                                              00055500
   6        *                (ROOT(J),J=1,NROOT)                                          00055600
   7         WRITE(KIO6,1190)                                                              00055700
   7         WRITE(KIO6,1004)                                                              00055800
   6        *                (ERRVL(J),J=1,NROOT)                                         00055900
   1   C                                                                                   00056000
   7         REWIND NT                                                                     00056100
   7         DO 970 I=1,NROOT                                                              00056200
   3     970 ROOT(I)=DSQRT(ROOT(I))                                                        00056300
   7         WRITE (NT) (ROOT(I),I=1,NROOT)                                                00056400
   7         NWA=N*NROOT                                                                   00056500
   7         WRITE (NT) (A(I),I=1,NWA)                                                     00056600
   1   C                                                                                   00056700
   7         RETURN                                                                        00056800
   1   C                                                                                   00056900
   2    1000 FORMAT (43H ***ERROR   NEG OR ZERO DIAGONAL ELEMENT A(,I4,4H) =  ,          00057000
   6       1         E11.4,21HBEFORE DECOMPOSITION    )                                   00057100
   2    1004 FORMAT (1H0,6E20.12)                                                         00057200
   2    1006 FORMAT (1H0,6I20)                                                            00057300
   2    1008 FORMAT (1H0,6F20.2)                                                          00057400
   2    1009 FORMAT (44H0***ERROR   SOLUTION TERMINATED IN *SECANT*  , /                  00057500
   6       1         12X,25HRIGID BODY MODE(S) FOUND., / 1X)                             00057600
   2    1010 FORMAT (51H1INVERSE ITERATION GIVES FOLLOWING APPROXIMATION TO,              00057700
   6       1         18H LOWEST EIGENVALUE, 1X)                                          00057800
   2    1014 FORMAT (/// 40H RAYLEIGH QUOTIENT SHIFT IS CARRIED OUT     )                00057900
   2    1015 FORMAT (42H0***ERROR   PRE-MATURE EXIT FROM *SECANT*  , / 12X,              00058000
   6       1         37HITERATION ABANDONED FOR ROOT NUMBER =, I4 / 12X,                 00058100
   6       2         37HNUMBER OF ITERATIONS PERFORMED      =, I4 / 1X)                  00058200
   2    1020 FORMAT (5H0RB = E20.12,7H NSCH = I4)                                         00058300
   2    1030 FORMAT (38H0***ERROR   SOLUTION STOP IN *SECANT* , / 12X, 1H(,               00058400
   6       1         I3,48H) FACTORIZATIONS PERFORMED IN AN ATTEMPT TO FIND,              00058500
   6       2         32H LOWER BOUND ON FIRST EIGENVALUE, / 12X,                         00058600
   6       3         16HCHECK THE MODEL., / 1X)                                          00058700
   2    1040 FORMAT (1H1,4X,4HROOT,4X,4HNITE,18X,2HRC,15X,12HDET (A-RC*B),15X,           00058800
   6       /2HFC,13X,3HETA,4X,3HISC)                                                    00058900
   2    1050 FORMAT (1H0,4X,I4,4X,I4,8X,3E22.14,F7.2,I6)                                  00059000
```

```
START                                                                    MOD
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8  FLAGS
   2   1060 FORMAT (42H0THE DEFLATED POLYNOMIAL HAS NO MORE ROOTS  )          00059100
   2   1070 FORMAT (29H0(RC-RB) IS SMALLER THAN TOL )                         00059200
   2   1080 FORMAT (16H0WE JUMPED OVER I4,16H UNKNOWN ROOT(S)  )              00059300
   2   1090 FORMAT (42H0***ERROR   PRE-MATURE EXIT FROM *SECANT*  ,           00059400
   6       1         34H CAUSED BY EITHER OF THE FOLLOWING, / 12X,            00059500
   6       2         22H(1) BAD MODEL DATA, OR, / 12X,                        00059600
   6       3         52H(2) ROOT CLUSTER (I.E., NEAR EQUAL OR REPEATED EIGEN, 00059700
   6       4         36HVALUES) ENCOUNTERED AT CURRENT SHIFT, / 16X,          00059800
   6       5         25HCAUSING STORAGE OVER-FLOW, 1X)                        00059900
   2   1100 FORMAT (1H0,34X,4HROOT,18X,2HRQ,18X,4HNOR=,I2)                    00060000
   2   1110 FORMAT (1H0,4X,I4,4X,I4,8X,2E22.14)                               00060100
   2   1120 FORMAT (20H0TIME FOR INV ITERN F5.2)                              00060200
   2   1140 FORMAT (42H0NO OF ITERATIONS FOR EACH EIGENVALUE ARE   /)         00060300
   2   1150 FORMAT (30H0TIME USED FOR EACH EIGENVALUE /)                      00060400
   2   1160 FORMAT (46H0FOLLOWING ARE ERROR BOUNDS ON THE EIGENVALUES /       00060500
   6       1         51H (THE BOUNDS DEPEND ON THE SHIFTING IN THE SOLUTION, 00060600
   6       2         28H AND CAN THEREFORE BE LARGE) )                        00060700
   2   1170 FORMAT (1H1,22H E I G E N V A L U E S    )                        00060800
   2   1180 FORMAT (38H0***ERROR   SOLUTION STOP IN *SECANT*   , / 12X,       00060900
   6       1         23HNO EIGENVALUES COMPUTED, / 1X)                        00061000
   2   1190 FORMAT (///  40H THE FOLLOWING ARE PHYSICAL ERROR BOUNDS,         00061100
   6       1             20H ON THE EIGENVALUES )                           00061200
   1   C                                                                      00061300
   7         END                                                            00061400
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7        SUBROUTINE STIME                                              00000100
   7        RETURN                                                        00000200
   7        END                                                           00000300
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8
```

```
   7        SUBROUTINE STRESS                                                      00000100
   7        IMPLICIT REAL*8(A-H,O-Z)                                               00000200
                                                                                   00000300
   1  C
   1  C . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 00000400
   1  C .                                                                       . 00000500
   1  C .   P R O G R A M                                                       . 00000600
   1  C .      . TO CALL THE ELEMENT SUBROUTINE FOR THE CALCULATION OF          . 00000700
   1  C .        STRESSES                                                       . 00000800
   1  C .                                                                       . 00000900
   1  C . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 00001000
                                                                                   00001100
   1  C
   7        COMMON/BBCCIO/KI01,KI02,KI03,KI04,KI05,KI06,KI07,KI08,KI09,           00001200
   6       1KI010,KI011,KI012,KI013,KI014,KI015,KI016,KI017,KI018,KI019,KI020,00001300
   6       2KI021,KI022,KI023,KI024,KI025,KI026,KI027,KI028,KI029,KI030          00001400
   7        COMMON /SOL/ NUMNP,NEQ,NWK,NWM,NWC,NUMEST,MIDEST,MAXEST,NSTE,MA      00001500
   7        COMMON /EL/ IND,ICOUNT,NPAR(20),NUMEG,NEGL,NEGNL,IMASS,IDAMP,ISTAT00001600
   6       1           ,NDOF,KLIN,IEIG,IMASSN,IDAMPN                           00001700
   7        COMMON /VAR/ NG,KPRI,MODEX,KSTEP,ITE,ITEMAX,IREF,IEQREF,INOCMD      00001800
   7        COMMON /TAPES/ NREAD,NWRITE                                            00001900
   7        COMMON A(1)                                                            00002000
                                                                                   00002100
   1  C
   7        NG=0                                                                   00002200
                                                                                   00002300
   1  C
   1  C                                                                            00002400
   1  C     L I N E A R   E L E M E N T   G R O U P S                              00002500
   1  C                                                                            00002600
   1  C                                                                            00002700
   7        IF (NEGL.EQ.0) GO TO 200                                               00002800
   7        REWIND KI011                                                           00002900
   1  C                                                                            00003000
   7        DO 100 N=1,NEGL                                                        00003100
   7        NG=NG + 1                                                              00003200
   1  C                                                                            00003300
   7        READ(KI011)                                                            00003400
   6       *          NUMEST,NPAR,(A(I),I=1,NUMEST)                               00003500
                                                                                   00003600
   1  C
   7        CALL ELEMNT                                                            00003700
   1  C                                                                            00003800
   3    100 CONTINUE                                                               00003900
                                                                                   00004000
   1  C
   1  C                                                                            00004100
   1  C     N O N L I N E A R   E L E M E N T   G R O U P S                        00004200
   1  C                                                                            00004300
   1  C                                                                            00004400
   7        IF (NEGNL.EQ.0) RETURN                                                 00004500
   1  C                                                                            00004600
   3    200 REWIND NREAD                                                           00004700
   7        DO 300 N=1,NEGNL                                                       00004800
   7        NG=NG + 1                                                              00004900
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   1  C                                                                    00005000
   7        READ (NREAD) NUMEST,NPAR,(A(I),I=1,NUMEST)                     00005100
   1  C                                                                    00005200
   7        CALL ELEMNT                                                    00005300
   1  C                                                                    00005400
   3    300 CONTINUE                                                       00005500
   1  C                                                                    00005600
   7        RETURN                                                         00005700
   7        END                                                            00005800
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7        SUBROUTINE STSTL (NEL,XX,PROP,C)                                      00000100
   7        IMPLICIT REAL*8(A-H,O-Z)                                              00000200
   1  C                                                                           00000300
   1  C                                                                           00000400
   1  C .      TO GENERATE THE GLOBAL STRESS-STRAIN LAW FOR                       00000500
   1  C .      ISOTROPIC AND ORTHOTROPIC MATERIALS                                00000600
   1  C                                                                           00000700
   1  C                                                                           00000800
   7        COMMON/BBCCIO/KIO1,KIO2,KIO3,KIO4,KIO5,KIO6,KIO7,KIO8,KIO9,          00000900
   6       1KIO10,KIO11,KIO12,KIO13,KIO14,KIO15,KIO16,KIO17,KIO18,KIO19,KIO20,  00001000
   6       2KIO21,KIO22,KIO23,KIO24,KIO25,KIO26,KIO27,KIO28,KIO29,KIO30         00001100
   7        COMMON /EL/ IND,ICOUNT,NPAR(20),NUMEG,NEGL,NEGNL,IMASS,IDAMP,ISTAT  00001200
   6       1           ,NDOF,KLIN,IEIG,IMASSN,IDAMPN                           00001300
   7        COMMON /TODIM/ BET,THIC,DE,IEL,NND5                                   00001400
   1  C                                                                           00001500
   7        DIMENSION XX(2,1),PROP(1),C(4,1),D(4,4),T(4,4)                        00001600
   1  C                                                                           00001700
   7        EQUIVALENCE (NPAR(5),ITYP2D),(NPAR(15),MODEL)                         00001800
   1  C                                                                           00001900
   1  C                                                                           00002000
   7        GO TO (1,2), MODEL                                                    00002100
   1  C                                                                           00002200
   1  C                                                                           00002300
   1  C                                                                           00002400
   1  C.... MODEL = 1    L I N E A R    I S O T R O P I C                        00002500
   1  C                                                                           00002600
   5      1 YM=PROP(1)                                                          00002700
   7        PV=PROP(2)                                                          00002800
   7        C1=YM/(1+PV)                                                        00002900
   7        B1=C1*PV/(1.-2.*PV)                                                 00003000
   7        A1=B1+C1                                                            00003100
   7        C(1,1)=A1                                                           00003200
   7        C(1,2)=B1                                                           00003300
   7        C(1,3)=0.                                                           00003400
   7        C(2,1)=B1                                                           00003500
   7        C(2,2)=A1                                                           00003600
   7        C(2,3)=0.                                                           00003700
   7        C(3,1)=0.                                                           00003800
   7        C(3,2)=0.                                                           00003900
   7        C(3,3)=C1/2.                                                        00004000
   1  C                                                                           00004100
   7        IF (ITYP2D.EQ.1) RETURN                                             00004200
   1  C                                                                           00004300
   7        C(1,4)=B1                                                           00004400
   7        C(2,4)=B1                                                           00004500
   7        C(3,4)=0.                                                           00004600
   7        C(4,1)=B1                                                           00004700
   7        C(4,2)=B1                                                           00004800
   7        C(4,3)=0.                                                           00004900
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8
   7          C(4,4)=A1                                                          00005000
                                                                                 00005100
   1  C
   7          IF (ITYP2D.LT.2) RETURN                                            00005200
                                                                                 00005300
   1  C
   1  C       FOR PLANE STRESS ANALYSIS CONDENSE STRESS-STRAIN MATRIX            00005400
   1  C                                                                          00005500
   7          GO TO 95                                                           00005600
                                                                                 00005700
   1  C
   1  C                                                                          00005800
   1  C                                                                          00005900
   1  C.... MODEL = 2    L I N E A R   O R T H O T R O P I C                     00006000
                                                                                 00006100
   1  C
   5      2 IF (PROP(3).EQ.0.) GO TO 1                                          00006200
   7        PI = 4.0* ATAN(1.0)                                                 00006300
                                                                                 00006400
   1  C
   1  C       COMPUTE THE ANGLE BETWEEN EDGE 1-2 AND THE GLOBAL X-AXIS           00006500
                                                                                 00006600
   1  C
   7          DX = XX(1,2) - XX(1,1)                                             00006700
   7          DY = XX(2,2) - XX(2,1)                                             00006800
   7          XL = DX**2 + DY**2                                                 00006900
   7          IF(XL.GT.1.0E-12) GO TO 102                                        00007000
   7          WRITE(KIO6,2000) NEL                                               00007100
   7          STOP                                                               00007200
   3    102 XL=DSQRT(XL)                                                         00007300
   7          SA=DABS(DY/XL)                                                     00007400
   7          AL=DATAN(SA/DSQRT(1-SA*SA))                                        00007500
   7          IF(DX.GE.0.0 .AND. DY.GE.0.0) P12 =          AL                    00007600
   7          IF(DX.LT.0.0 .AND. DY.GE.0.0) P12 = PI     - AL                   00007700
   7          IF(DX.LT.0.0 .AND. DY.LT.0.0) P12 = PI     + AL                   00007800
   7          IF(DX.GE.0.0 .AND. DY.LT.0.0) P12 = PI*2.0- AL                     00007900
                                                                                 00008000
   1  C
   1  C       COMPUTE THE ANGLE BETWEEN THE MATERIAL A-AXIS AND GLOBAL X-AXIS    00008100
                                                                                 00008200
   1  C
   7          PI2=PI*2.                                                          00008300
   7          IF(DABS(P12).GT.PI2) STOP                                          00008400
   7          GAM=P12 + BET                                                      00008500
   7          IF (GAM.GE.PI2) GAM=GAM-PI2                                        00008600
                                                                                 00008700
   1  C
   1  C       SET THE COORDINATE TRANSFORMATION FOR ROTATION OF PROPERTIES        00008800
                                                                                 00008900
   1  C
   7          IF(DABS(GAM).LT.1.0D-8) GO TO 202                                  00009000
   7          SG=DSIN(GAM)                                                       00009100
   7          CG=DCOS(GAM)                                                       00009200
   7          T(1,1) =  CG**2                                                    00009300
   7          T(1,2) =  SG**2                                                    00009400
   7          T(1,3) =  CG* SG                                                   00009500
   7          T(1,4) =  0.0                                                      00009600
   7          T(2,1) =  T(1,2)                                                   00009700
   7          T(2,2) =  T(1,1)                                                   00009800
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7        T(2,3) = -T(1,3)                                                    00009900
   7        T(2,4) =  0.0                                                       00010000
   7        T(3,1) =  T(2,3)* 2.0                                               00010100
   7        T(3,2) = -T(3,1)                                                    00010200
   7        T(3,3) =  T(1,1)- T(1,2)                                            00010300
   7        T(3,4) =  0.0                                                       00010400
   7        T(4,1) =  0.0                                                       00010500
   7        T(4,2) =  0.0                                                       00010600
   7        T(4,3) =  0.0                                                       00010700
   7        T(4,4) =  1.0                                                       00010800
   3   202 CONTINUE                                                            00010900
   1  C                                                                        00011000
   1  C     FORM THE STRAIN-STRESS LAW, SYSTEM (A,B,C)                         00011100
   1  C                                                                        00011200
   7        DUM = PROP(1)* PROP(2)* PROP(3)* PROP(7)                           00011300
   7        IF (DUM.GT.1.0E-08) GO TO 25                                       00011400
   7        WRITE(KIO6,2010)                                                   00011500
   7        STOP                                                               00011600
   4    25 C(1,1) = 1.0/PROP(1)                                                00011700
   7        C(2,2) = 1.0/PROP(2)                                               00011800
   7        C(3,3) = 1.0/PROP(7)                                               00011900
   7        C(4,4) = 1.0/PROP(3)                                               00012000
   7        C(1,2) =-PROP(4)* C(2,2)                                           00012100
   7        C(1,4) =-PROP(5)* C(4,4)                                           00012200
   7        C(2,4) =-PROP(6)* C(4,4)                                           00012300
   7        C(1,3) = 0.0                                                       00012400
   7        C(2,3) = 0.0                                                       00012500
   7        C(3,4) = 0.0                                                       00012600
   7        DO 30 I=1,4                                                        00012700
   7        DO 30 J=I,4                                                        00012800
   4    30 C(J,I) = C(I,J)                                                     00012900
   1  C                                                                        00013000
   1  C     FORM THE STRESS-STRAIN LAW, SYSTEM (A,B,C)                         00013100
   1  C                                                                        00013200
   7        CALL POSINV (C,4,4)                                                00013300
   1  C                                                                        00013400
   1  C     ROTATE THE STRESS-STRAIN MATRIX TO GLOBAL COORDINATES              00013500
   1  C                                                                        00013600
   7        IF(DABS(GAM).LT.1.0D-8) GO TO 90                                   00013700
   1  C                                                                        00013800
   1  C     T(TRANSPOSE) * C(MATERIAL)                                         00013900
   1  C                                                                        00014000
   7        DO 60 IR=1,4                                                       00014100
   7        DO 60 IC=1,4                                                       00014200
   7        D(IR,IC) = 0.0                                                     00014300
   7        DO 50 IN=1,4                                                       00014400
   4    50 D(IR,IC) = D(IR,IC) + T(IN,IR)* C(IN,IC)                            00014500
   4    60 CONTINUE                                                            00014600
   1  C                                                                        00014700
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   1  C      T(TRANSPOSE) * C(MATERIAL) * T                                    00014800
   1  C                                                                        00014900
   7         DO 80 IR=1,4                                                     00015000
   7         DO 80 IC=IR,4                                                    00015100
   7         C(IR,IC) = 0.0                                                   00015200
   7         DO 70 IN=1,4                                                     00015300
   4     70 C(IR,IC) = C(IR,IC) + D(IR,IN)* T(IN,IC)                          00015400
   4     80 C(IC,IR)=C(IR,IC)                                                 00015500
                                                                              00015600
   1  C                                                                       00015700
   4     90 IF (ITYP2D.LT.2) RETURN                                           00015800
   1  C                                                                       00015900
   1  C      FOR PLANE STRESS ANALYSIS CONDENSE STRESS-STRAIN MATRIX           00016000
   1  C                                                                       00016100
   4     95 DO 110 I=1,3                                                     00016200
   7         A=C(I,4)/C(4,4)                                                  00016300
   7         DO 110 J=I,3                                                     00016400
   7         C(I,J)=C(I,J) - C(4,J)*A                                         00016500
   3    110 C(J,I)=C(I,J)                                                     00016600
   7         RETURN                                                           00016700
   1  C                                                                       00016800
   1  C                                                                       00016900
   1  C                                                                       00017000
   2   2000 FORMAT (10H0*** ERROR,/                                           00017100
   6     1        43H ZERO LENGTH BETWEEN NODES 1-2 IN ELEMENT (,I4,1H))     00017200
   2   2010 FORMAT (45H0***ERROR  MATERIAL PROPERTIES NOT ADMISSABLE  )       00017300
   1  C                                                                       00017400
   7         END
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7        SUBROUTINE STSTN (XX,PROP,DISD,IDW,WA)                                 00000100
   7        IMPLICIT REAL*8(A-H,O-Z)                                               00000200
                                                                                   00000300
   1  C                                                                            00000400
   1  C                                                                            00000500
   1  C . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 00000600
   1  C .                                                                        . 00000700
   1  C .                                                                        . 00000800
   1  C .   S U B R O U T I N E                                                  . 00000900
   1  C .                                                                        . 00001000
   1  C .      TO FIND STRESS STRAIN LAW AND STRESSES FOR                        . 00001100
   1  C .      NONLINEAR MATERIAL MODELS                                         . 00001200
   1  C .                                                                        . 00001300
   1  C .                                                                        . 00001400
   1  C . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 00001500
   1  C                                                                            00001600
   1  C                                                                            00001700
   7        COMMON /EL/ IND,ICOUNT,NPAR(20),NUMEG,NEGL,NEGNL,IMASS,IDAMP,ISTAT     00001800
   6       1           ,NDOF,KLIN,IEIG,IMASSN,IDAMPN                               00001900
   7        COMMON /VAR/ NG,KPRI,MODEX,KSTEP,ITE,ITEMAX,IREF,IEQREF,INOCMD        00002000
   7        COMMON /MATMOD/ STRESS(4),STRAIN(4),C(4,4),IPT,NEL                    00002100
   1  C                                                                            00002200
   7        DIMENSION WA(IDW,1),XX(2,1),PROP(1),DISD(1),DN(4)                     00002300
   1  C                                                                            00002400
   7        EQUIVALENCE (NPAR(5),ITYP2D),(NPAR(15),MODEL),(NPAR(3),INDNL)         00002500
   6       1           ,(NPAR(17),NCON)                                           00002600
   1  C                                                                            00002700
   1  C                                                                            00002800
   7        IST=3                                                                 00002900
   7        IF (ITYP2D.EQ.0) IST=4                                                00003000
   7        IP=NCON/4                                                             00003100
   1  C                                                                            00003200
   1  C                                                                            00003300
   1  C     D E F I N I T I O N  O F  S T R A I N                                  00003400
   1  C                                                                            00003500
   1  C                                                                            00003600
   1  C     LINEAR STRAIN TERMS                                                    00003700
   1  C                                                                            00003800
   7        STRAIN(1)=DISD(1)                                                     00003900
   7        STRAIN(2)=DISD(2)                                                     00004000
   7        STRAIN(3)=DISD(3) + DISD(4)                                           00004100
   7        STRAIN(4)=0.                                                          00004200
   7        IF (ITYP2D.EQ.0) STRAIN(4)=DISD(5)                                    00004300
   7        IF (INDNL.EQ.1) GO TO 80                                              00004400
   1  C                                                                            00004500
   1  C     NONLINEAR STRAIN TERMS                                                 00004600
   1  C                                                                            00004700
   7        DN(1)=0.5*(DISD(1)*DISD(1) + DISD(4)*DISD(4))                         00004800
   7        DN(2)=0.5*(DISD(2)*DISD(2) + DISD(3)*DISD(3))                         00004900
   7        DN(3)=DISD(3)*DISD(1) + DISD(4)*DISD(2)
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8
   7        IF (IST.EQ.4) DN(4)=0.5*DISD(5)*DISD(5)                          00005000
                                                                             00005100
   1  C
   7        IF (INDNL.EQ.3) GO TO 60                                         00005200
                                                                             00005300
   1  C
   1  C     CALCULATE GREEN-LAGRANGE STRAINS (TOTAL LAGRANGE FORMULATION)    00005400
   1  C                                                                      00005500
   7        DO 20 I=1,IST                                                    00005600
   4     20 STRAIN(I)=STRAIN(I) + DN(I)                                      00005700
   7        GO TO 80                                                         00005800
                                                                             00005900
   1  C
   1  C     CALCULATE ALMANSI STRAINS (UPDATED LAGRANGIAN FORMULATION)       00006000
   1  C                                                                      00006100
   4     60 DO 40 I=1,IST                                                    00006200
   4     40 STRAIN(I)=STRAIN(I) - DN(I)                                      00006300
                                                                             00006400
   1  C                                                                      00006500
   1  C                                                                      00006600
   1  C                                                                      00006700
   1  C     C A L C U L A T I O N   O F   S T R E S S - S T R A I N          00006800
   1  C                     M A T R I X   A N D   S T R E S S E S            00006900
   1  C                                                                      00007000
   1  C                                                                      00007100
   4     80 GO TO ( 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12 ) , MODEL         00007200
                                                                             00007300
   1  C                                                                      00007400
   1  C                                                                      00007500
   1  C.... MODEL = 1    L I N E A R   I S O T R O P I C                     00007600
   1  C                                                                      00007700
   1  C                                                                      00007800
   5      1 A1=C(1,1)                                                       00007900
   7        B1=C(1,2)                                                       00008000
   1  C                                                                      00008100
   7        STRESS(1) = A1*STRAIN(1) + B1*STRAIN(2)                         00008200
   7        STRESS(2) = B1*STRAIN(1) + A1*STRAIN(2)                         00008300
   7        STRESS(3) = C(3,3)*STRAIN(3)                                    00008400
   7        STRESS(4) = 0.                                                  00008500
   1  C                                                                      00008600
   1  C     PLANE STRESS PROBLEM                                            00008700
   7        A2=-PROP(2)/(1. - PROP(2))                                      00008800
   7        IF (ITYP2D.EQ.2) STRAIN(4)=A2*(STRAIN(1) + STRAIN(2))           00008900
   7        IF (ITYP2D.EQ.2) GO TO 110                                      00009000
   1  C                                                                      00009100
   1  C     PLANE STRAIN PROBLEM                                            00009200
   7        STRESS(4) = B1*(STRAIN(1)+STRAIN(2))                            00009300
   7        IF (ITYP2D.EQ.1) GO TO 110                                      00009400
   1  C                                                                      00009500
   1  C     AXISYMMETRIC PROBLEM                                            00009600
   1  C                                                                      00009700
   7        STRESS(1) = STRESS(1) + B1*STRAIN(4)                            00009800
   7        STRESS(2) = STRESS(2) + B1*STRAIN(4)
   7        STRESS(4) = STRESS(4) + A1*STRAIN(4)
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8
   1  C                                                                          00009900
   3    110 RETURN                                                              00010000
   1  C                                                                          00010100
   1  C                                                                          00010200
   1  C.... MODEL = 2    L I N E A R   O R T H O T R O P I C                     00010300
   1  C                                                                          00010400
   1  C                                                                          00010500
   5      2 IF (PROP(3).EQ.0.) GO TO 1                                          00010600
   1  C                                                                          00010700
   7        DO 102 I=1,4                                                        00010800
   3    102 STRESS(I)=0.                                                        00010900
   1  C                                                                          00011000
   1  C     PLANE STRESS OR AXISYMMETRIC PROBLEM                                00011100
   1  C                                                                          00011200
   7        IST=4                                                               00011300
   7        IF (ITYP2D.NE.0) IST=3                                              00011400
   7        DO 103 J=1,IST                                                      00011500
   7        DO 103 I=1,IST                                                      00011600
   3    103 STRESS(I) = STRESS(I) + C(I,J)*STRAIN(J)                            00011700
   7        IF (ITYP2D.EQ.2) STRAIN(4) =                                        00011800
   6       1  -(C(4,1)*STRAIN(1)+C(4,2)*STRAIN(2)+C(4,3)*STRAIN(3))/C(4,4)     00011900
   7        IF (ITYP2D.NE.1) GO TO 120                                          00012000
   1  C                                                                          00012100
   1  C     PLANE STRAIN PROBLEM                                                00012200
   1  C                                                                          00012300
   7        STRESS(4)=C(4,1)*STRAIN(1) + C(4,2)*STRAIN(2) + C(4,3)*STRAIN(3)    00012400
   1  C                                                                          00012500
   3    120 RETURN                                                              00012600
   1  C                                                                          00012700
   1  C                                                                          00012800
   1  C.... MODEL = 3    V A R I A B L E   T A N G E N T   M O D U L I           00012900
   1  C                                                                          00013000
   1  C                                                                          00013100
   1  3     CALL ELT2D3                                                         00013200
   7        RETURN                                                              00013300
   1  C                                                                          00013400
   1  C                                                                          00013500
   1  C.... MODEL = 4    C U R V E   D E S C R I P T I O N   M O D E L           00013600
   1  C                                                                          00013700
   1  C                                                                          00013800
   1  4     CALL ELT2D4                                                         00013900
   7        RETURN                                                              00014000
   1  C                                                                          00014100
   1  C                                                                          00014200
   1  C.... MODEL = 5    C D M O D   W I T H   T E N S I O N   C U T - O F F     00014300
   1  C                                                                          00014400
   1  C                                                                          00014500
   1  5     CALL ELT2D4                                                         00014600
   7        RETURN                                                              00014700
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   1  C                                                                      00014800
   1  C                                                                      00014900
   1  C.... MODEL = 6    E L A S T O P L A S T I C (VON MISES YIELD CRITERION)00015000
   1  C                                                                      00015100
   1  C                                                                      00015200
   1  6      CALL ELT2D6                                                     00015300
   7         RETURN                                                          00015400
   1  C                                                                      00015500
   1  C                                                                      00015600
   1  C.... MODEL = 7    E L A S T O P L A S T I C (DRUCKER-PRAGER)           00015700
   1  C                                                                      00015800
   1  C                                                                      00015900
   1  7      CALL ELT2D7                                                     00016000
   7         RETURN                                                          00016100
   1  C                                                                      00016200
   1  C                                                                      00016300
   1  C.... MODEL = 8    I N C O M P R E S S I B L E   E L A S T I C         00016400
   1  C                       ( MOONEY-RIVLIN IN STATE OF PLANE STRESS )     00016500
   1  C                                                                      00016600
   1  C                                                                      00016700
   1  8      CALL ELT2D8                                                     00016800
   7         RETURN                                                          00016900
   1  C                                                                      00017000
   1  C                                                                      00017100
   1  C.... MODEL .GE. 9    U S E R   S U P P L I E D   M O D E L S          00017200
   1  C                                                                      00017300
   1  C                                                                      00017400
   1  9      CALL ELT2D9                                                     00017500
   7         RETURN                                                          00017600
   1  C                                                                      00017700
   1  10     CALL EL2D10                                                     00017800
   7         RETURN                                                          00017900
   1  C                                                                      00018000
   1  11     CALL EL2D11                                                     00018100
   7         RETURN                                                          00018200
   1  C                                                                      00018300
   1  12     CALL EL2D12                                                     00018400
   7         RETURN                                                          00018500
   1  C                                                                      00018600
   1  C                                                                      00018700
   2   2035 FORMAT (I8)                                                     00018800
   2   2040 FORMAT (13X,I5,6F15.4,F10.2)                                    00018900
   1  C                                                                      00019000
   1  C                                                                      00019100
   7         END                                                             00019200
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7        SUBROUTINE STST3L  (NEL,XX,PROP,C)                                  00000100
   7        IMPLICIT REAL*8(A-H,O-Z)                                            00000200
   1  C                                                                         00000300
   1  C                                                                         00000400
   1  C . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .  00000500
   1  C .                                                                   .  00000600
   1  C .                                                                   .  00000700
   1  C .   P R O G R A M                                                   .  00000800
   1  C .                                                                   .  00000900
   1  C .      TO GENERATE STRESS-STRAIN LAW FOR                            .  00001000
   1  C .        ISOTROPIC OR ORTHOTROPIC LINEAR ELASTIC                    .  00001100
   1  C .          THREE-DIMENSIONAL MATERIALS                              .  00001200
   1  C .                                                                   .  00001300
   1  C .                                                                   .  00001400
   1  C . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .  00001500
   1  C                                                                         00001600
   1  C                                                                         00001700
   1  C                                                                         00001800
   7        COMMON /EL/ IND,ICOUNT,NPAR(20),NUMEG,NEGL,NEGNL,IMASS,IDAMP,ISTAT00001900
   6       1           ,NDOFDM,KLIN,IEIG,IMASSN,IDAMPN                         00002000
   7        COMMON /THREED/ DE,IELD,IELX,IEL,NPT,IDW,NND9                     00002100
   7        DIMENSION XX(3,1),PROP(1),C(6,1),D(6,6),T(6,6)                    00002200
   1  C                                                                         00002300
   7        EQUIVALENCE (NPAR(15),MODEL)                                      00002400
   1  C                                                                         00002500
   1  C                                                                         00002600
   1  C                                                                         00002700
   7        GO TO (1,100), MODEL                                              00002800
   1  C                                                                         00002900
   1  C                                                                         00003000
   1  C     M O D E L  1     LINEAR ELASTIC ISOTROPIC                           00003100
   1  C                                                                         00003200
   1  C                                                                         00003300
   5      1 YM=PROP(1)                                                        00003400
   7        PV=PROP(2)                                                        00003500
   1  C                                                                         00003600
   7        C1=1. - 2.*PV                                                     00003700
   7        B1=YM/(1. + PV)                                                   00003800
   7        A1=B1/C1                                                          00003900
   7        D1=1. - PV                                                        00004000
   1  C                                                                         00004100
   7        DO 9 I=1,6                                                        00004200
   7        DO 9 J=1,6                                                        00004300
   5      9 C(I,J)=0.0                                                        00004400
   7        DO 10 I=1,3                                                       00004500
   4     10 C(I,I)=A1*D1                                                      00004600
   7        DO 11 I=2,3                                                       00004700
   4     11 C(1,I)=A1*PV                                                      00004800
   7        C(2,3)=A1*PV                                                      00004900
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7          DO 12 I=4,6                                                  00005000
   4       12 C(I,I)=B1/2.                                                 00005100
   7          DO 13 I=1,6                                                  00005200
   7          DO 13 J=1,6                                                  00005300
   4       13 C(J,I)=C(I,J)                                                00005400
   1  C                                                                    00005500
   7          RETURN                                                       00005600
   1  C                                                                    00005700
   1  C                                                                    00005800
   1  C                                                                    00005900
   3     100 CONTINUE                                                     00006000
   7          RETURN                                                       00006100
   1  C                                                                    00006200
   7          END                                                          00006300
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7        SUBROUTINE STST3N (DISD)                                          00000100
   7        IMPLICIT REAL*8(A-H,O-Z)                                          00000200
   1  C                                                                       00000300
   1  C                                                                       00000400
   1  C . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .  00000500
   1  C .                                                                   . 00000600
   1  C .     S U B R O U T I N E                                           . 00000700
   1  C .                                                                   . 00000800
   1  C .        TO FIND STRESSES FOR ALL MATERIAL MODELS AND               . 00000900
   1  C .           STRESS-STRAIN LAW FOR NONLINEAR MATERIAL MODELS         . 00001000
   1  C .                                                                   . 00001100
   1  C . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .  00001200
   1  C                                                                       00001300
   1  C                                                                       00001400
   7        COMMON /EL/ IND,ICOUNT,NPAR(20),NUMEG,NEGL,NEGNL,IMASS,IDAMP,ISTAT00001500
   6       1          ,NDOF,KLIN,IEIG,IMASSN,IDAMPN                          00001600
   7        COMMON /MTMD3D/  D(6,6),STRESS(6),STRAIN(6),IPT,NEL               00001700
   1  C                                                                       00001800
   7        DIMENSION DISD(1)                                                 00001900
   1  C                                                                       00002000
   7        EQUIVALENCE (NPAR(3),INDNL), (NPAR(15),MODEL)                     00002100
   1  C                                                                       00002200
   1  C                                                                       00002300
   1  C     D E F I N I T I O N   O F   S T R A I N                           00002400
   1  C                                                                       00002500
   1  C                                                                       00002600
   1  C     LINEAR STRAIN TERMS                                               00002700
   1  C                                                                       00002800
   7        STRAIN(1)=DISD(1)                                                 00002900
   7        STRAIN(2)=DISD(2)                                                 00003000
   7        STRAIN(3)=DISD(3)                                                 00003100
   7        STRAIN(4)=DISD(4) + DISD(6)                                       00003200
   7        STRAIN(5)=DISD(5) + DISD(8)                                       00003300
   7        STRAIN(6)=DISD(7) + DISD(9)                                       00003400
   7        IF (INDNL.EQ.1) GO TO 80                                          00003500
   1  C                                                                       00003600
   1  C     NONLINEAR STRAIN TERMS                                            00003700
   1  C          ( OMITTED )                                                  00003800
   1  C                                                                       00003900
   1  C                                                                       00004000
   1  C     C A L C U L A T I O N   O F   S T R E S S - S T R A I N           00004100
   1  C          M A T R I X   A N D   S T R E S S E S                        00004200
   1  C                                                                       00004300
   1  C                                                                       00004400
   4     80 GO TO (1, 2, 3, 4, 5, 6), MODEL                                   00004500
   1  C                                                                       00004600
   1  C                                                                       00004700
   1  C.... MODEL = 1     L I N E A R   I S O T R O P I C                     00004800
   1  C                                                                       00004900
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   5      1 DO 100 I=1,3                                                        00005000
   7        STRESS(I)=D(I,1)*STRAIN(1) + D(I,2)*STRAIN(2) + D(I,3)*STRAIN(3)    00005100
   3    100 STRESS(I+3)=D(4,4)*STRAIN(I+3)                                      00005200
   7        RETURN                                                              00005300
   1  C                                                                         00005400
   1  C                                                                         00005500
   1  C.... MODEL = 2     L I N E A R   O R T H O T R O P I C                   00005600
   1  C                                                                         00005700
   5      2 RETURN                                                            00005800
   1  C                                                                         00005900
   1  C                                                                         00006000
   1  C.... MODEL = 3     N O N L I N E A R   C U R V E                         00006100
   1  C                   D E S C R I P T I O N   M O D E L                     00006200
   1  C                                                                         00006300
   5      3 CALL ELT3D3                                                       00006400
   7        RETURN                                                            00006500
   1  C                                                                         00006600
   1  C                                                                         00006700
   1  C.... MODEL .GE. 4     U S E R   S U P P L I E D   M O D E L S          00006800
   1  C                                                                         00006900
   5      4 CALL ELT3D4                                                       00007000
   7        RETURN                                                            00007100
   1  C                                                                         00007200
   5      5 CALL ELT3D5                                                       00007300
   7        RETURN                                                            00007400
   1  C                                                                         00007500
   5      6 CALL ELT3D6                                                       00007600
   7        RETURN                                                            00007700
   1  C                                                                         00007800
   1  C                                                                         00007900
   7        END                                                               00008000
```

```
START                                                                                                  MOD
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8  FLAGS
   7        SUBROUTINE TDFE (ID,Y,Z,HT,LM,YZ,IELT,IPST,BETA,THICK,MATP,          00000100
   6       1                 DEN,PROP,WA,NOD5,ITABLE,NTABLE,NCON,IDWA,NDM,      00000200
   6       2                 ND5DIM,NDOF)                                       00000300
   7        IMPLICIT REAL*8(A-H,O-Z)                                            00000400
   1  C                                                                         00000500
   7        COMMON/BBCCIO/KIO1,KIO2,KIO3,KIO4,KIO5,KIO6,KIO7,KIO8,KIO9,        00000600
   6       1KIO10,KIO11,KIO12,KIO13,KIO14,KIO15,KIO16,KIO17,KIO18,KIO19,KIO20,00000700
   6       2KIO21,KIO22,KIO23,KIO24,KIO25,KIO26,KIO27,KIO28,KIO29,KIO30       00000800
   1  C                                                                         00000900
   7        COMMON /EL/ IND,ICOUNT,NPAR(20),NUMEG,NEGL,NEGNL,IMASS,IDAMP,ISTAT00001000
   6       1           ,NDOFDM,KLIN,IEIG,IMASSN,IDAMPN                         00001100
   7        COMMON /SOL/ NUMNP,NEQ,NWK,NWM,NWC,NUMEST,MIDEST,MAXEST,NSTE,MA    00001200
   7        COMMON /DIM/ N1,N2,N3,N4,N5,N6,N7,N8,N9,N10,N11,N12,N13,N14,N15    00001300
   7        COMMON /CONST/ DT,DTA,A0,A1,A2,A3,A4,A5,A6,A7,A8,A9,A10,A11        00001400
   6       1             ,A12,A13,A14,A15,A16,A17,A18,A19,A20,IOPE            00001500
   7        COMMON /DAMPNG/ ADAMP,BDAMP                                        00001600
   7        COMMON /VAR/ NG,KPRI,MODEX,KSTEP,ITE,ITEMAX,IREF,IEQREF,INOCMD     00001700
   7        COMMON /EM/ S(136),XM(16),B(4,16),NOD(8),NODM(8),RE(16),           00001800
   6       1            EDIS(16),AEM(204),XX(16)                              00001900
   7        COMMON /MATMOD/ STRESS(4),STRAIN(4),D(4,4),IPT,N                   00002000
   7        COMMON /TODIM/ BET,THIC,DE,IEL,NND5                                00002100
   7        COMMON /DISDER/ DISD(5)                                            00002200
   7        COMMON A(1)                                                        00002300
   1  C                                                                         00002400
   7        DIMENSION ID(NDOF,1), Y(1), Z(1), HT(1), LM(NDM,1), YZ(NDM,1),     00002500
   6       1          IELT(1), IPST(1), BETA(1), THICK(1), MATP(1),           00002600
   6       2          DEN(1),PROP(NCON,1),WA(IDWA,1),NOD5(ND5DIM,1),          00002700
   6       3          ITABLE(NTABLE,9)                                        00002800
   7        DIMENSION EVAL(9,2),XG(4,4),NOD5M(4)                               00002900
   1  C                                                                         00003000
   7        EQUIVALENCE (NPAR(2),NUME),(NPAR(16),NUMMAT),(NPAR(15),MODEL)      00003100
   6       1           ,(NPAR(10),NINT),(NPAR(1),NPAR1),(NPAR(7),MXNODS)      00003200
   6       2           ,(NPAR(5),ITYP2D),(NPAR(3),INDNL)                      00003300
   1  C                                                                         00003400
   7        DATA XG /     0.,             0.,             0.,            0.,  00003500
   6       1 -.5773502691896, .5773502691896,             0.,            0.,  00003600
   6       2 -.7745966692415, .0000000000000, .7745966692415,            0.,  00003700
   6       3 -.8611363115941,-.3399810435849, .3399810435849, .8611363115941/ 00003800
   1  C                                                                         00003900
   7        DATA EVAL / 1.0,-1.0,-1.0, 1.0, 0.0,-1.0, 0.0, 1.0, 0.0,           00004000
   6       1            1.0, 1.0,-1.0,-1.0, 1.0, 0.0,-1.0, 0.0, 0.0/          00004100
   7        DATA ANODE/4HNODE/                                                 00004200
   1  C                                                                         00004300
   1  C     ** NOTE ** DURING TIME INTEGRATION Y=DISP, Z=VEL, HT=R             00004400
   1  C                                                                         00004500
   1  C                                                                         00004600
   7        NPT = NINT*NINT                                                    00004700
   7        IDW=IDWA/NPT                                                       00004800
   1  C                                                                         00004900
```

```
START                                                                                MOD
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8  FLAGS

   7        IF (KPRI.EQ.0) GO TO 800                                                00005000
   7        IF (IND.GT.0) GO TO 420                                                 00005100
   1  C                                                                             00005200
   1  C                                                                             00005300
   1  C     R E A D   A N D   G E N E R A T E   E L E M E N T                       00005400
   1  C     I N F O R M A T I O N                                                   00005500
   1  C     CENTER NODE CAN BE ADDED FOR STRESS OUTPUT.DEGREE OF FREEDOM OF         00005600
   1  C     THE CENTER NODE IS ELIMINATED DURING STATIC CONDENSATION.               00005700
   1  C     S.N.MUKHERJEE.BBC,BADEN.AUG.1976.                                       00005800
   1  C                                                                             00006000
   7        WRITE(KIO6,2050)                                                        00006100
   6       *              NPAR1,NUME,INDNL,ITYP2D                                  00006200
   7        WRITE(KIO6,2015)                                                        00006300
   6       *              MXNODS,NINT,NTABLE                                       00006400
   7        IF (NG.LE.NEGL) INDNL=1                                                 00006500
   7        DO 10 I=1,NUMMAT                                                        00006600
   7        READ(KIO5,1000) N,DEN(N)                                                00006700
   7        READ(KIO5,1001)                                                         00006800
   6       *              (PROP(J,N), J=1,NCON)                                    00006900
   4    10 CALL MATRT2 (N,DEN(N),PROP(1,N))                                        00007000
   1  C                                                                             00007100
   1  C     FORM TABLES FOR STRESS OUTPUT                                           00007200
   1  C                                                                             00007300
   7        IF (NTABLE.EQ.0) GO TO 90                                               00007400
   7        WRITE(KIO6,2070)                                                        00007500
   7        DO 20 L=1,NTABLE                                                        00007600
   7        READ(KIO5,1002)                                                         00007700
   6       *              (ITABLE(L,I),I=1,9)                                      00007800
   4    20 WRITE(KIO6,2060)                                                        00007900
   6       *              L,(ITABLE(L,I),I=1,9)                                    00008000
   1  C                                                                             00008100
   1  C     READ ELEMENT INFORMATION                                                00008200
   1  C                                                                             00008300
   4    90 WRITE(KIO6,2005)                                                        00008400
   6       *              (ANODE,I,I=1,8)                                          00008500
   7        N=1                                                                     00008600
   3   100 READ(KIO5,1004)                                                        00008700
   6       *              M,IEL,IPS,BET,THIC,MTYP,KG,(NOD(I),I=1,8)                00008800
   7        IF (IEL.EQ.0) IEL=MXNODS                                                00008900
   7        IF (NDM.GE.IEL*2) GO TO 105                                             00009000
   7        WRITE(KIO6,2010) M                                                      00009100
   7        STOP                                                                    00009200
   3   105 IF (KG.EQ.0) KG=1                                                       00009300
   7        IF (ITYP2D.EQ.1) THIC=1.0                                               00009400
   3   120 IF (M.NE.N) GO TO 200                                                   00009500
   7        DO 110 I=1,8                                                            00009600
   3   110 NODM(I)=NOD(I)                                                          00009700
   7        IF (IEL.EQ.4) GO TO 115                                                 00009800
   7        II=0                                                                    00009900
```

```
START                                                                                                  MOD
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8  FLAGS
   7        DO 114 I=5,8                                                                 00010000
   7        NN=NOD(I)                                                                    00010100
   7        IF (NN.EQ.0) GO TO 114                                                       00010200
   7        II=II + 1                                                                    00010300
   7        NOD5M(II)=I                                                                  00010400
   3    114 CONTINUE                                                                     00010500
   3    115 IELM=IEL                                                                     00010600
   7        THICM=THIC                                                                   00010700
   7        IPSM=IPS                                                                     00010800
   7        BETM=BET                                                                     00010900
   7        MTYPE=MTYP                                                                   00011000
   7        KKK=KG                                                                       00011100
   1  C                                                                                  00011200
   1  C     SAVE ELEMENT INFORMATION                                                     00011300
   1  C                                                                                  00011400
   3    200 I2=0                                                                         00011500
   7        DO 130 I=1,IELM                                                              00011600
   7        II=NODM(I)                                                                   00011700
   7        IF (I.LE.4) GO TO 131                                                        00011800
   7        JJ=NOD5M(I-4)                                                                00011900
   7        II=NODM(JJ)                                                                  00012000
   3    131 I2=I2 + 2                                                                    00012100
   7        YZ(I2-1,N)=Y(II)                                                             00012200
   3    130 YZ(I2  ,N)=Z(II)                                                             00012300
   7        MATP(N)=MTYPE                                                                00012400
   7        BETA(N)=BETM                                                                 00012500
   7        THICK(N)=THICM                                                               00012600
   7        IELT(N)=IELM                                                                 00012700
   7        IPST(N)=IPSM                                                                 00012800
   7        IF (IELM.EQ.4) GO TO 135                                                     00012900
   7        NN=IELM - 4                                                                  00013000
   7        DO 132 I=1,NN                                                                00013100
   3    132 NOD5(I,N)=NOD5M(I)                                                           00013200
   1  C                                                                                  00013300
   3    135 KK=-2                                                                        00013400
   7        DO 140 I=1,IELM                                                              00013500
   7        II=NODM(I)                                                                   00013600
   7        IF (I.LE.4) GO TO 137                                                        00013700
   7        JJ=NOD5M(I-4)                                                                00013800
   7        II=NODM(JJ)                                                                  00013900
   3    137 KK=KK + 2                                                                    00014000
   7        DO 140 L=1,2                                                                 00014100
   3    140 LM(KK+L,N)=ID(L,II)                                                          00014200
   1  C                                                                                  00014300
   1  C     UPDATE COLUMN HEIGHTS AND BANDWIDTH(J.EISENBERG.AGBABIAN JACOBSON)           00014400
   1  C                                                                                  00014500
   7        ND=IELM*2                                                                    00014600
   7        CALL COLHT(HT,ND,LM(1,N))                                                    00014700
   1  C                                                                                  00014800
```

```
START                                                                                MOD
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8  FLAGS
   1  C     INITIALIZE WORKING STORAGE ARRAY FOR MATERIAL LAW                        00014900
   1  C                                                                             00015000
   7        IEL=IELM                                                                00015100
   7        NND5=IELM - 4                                                           00015200
   7        CALL INITWA (MODEL)                                                     00015300
   1  C                                                                             00015400
   7        WRITE(KIO6,2004)                                                        00015500
   6       *              N,IELM,IPSM,BETM,THICM,MTYP,KKK,(NODM(I),I=1,8)          00015600
   7        IF (N.EQ.NUME) RETURN                                                   00015700
   1  C                                                                             00015800
   7        N=N+1                                                                   00015900
   7        DO 160 I=1,IELM                                                         00016000
   3    160 NODM(I)=NODM(I)+KKK                                                     00016100
   7        IF (N.GT.M) GO TO 100                                                   00016200
   7        GO TO 120                                                               00016300
   1  C                                                                             00016400
   3    420 GO TO (440,560,560,700), IND                                            00016500
   1  C                                                                             00016600
   1  C                                                                             00016700
   1  C     A S S E M B L E   E F F E C T I V E   L I N E A R                       00016800
   1  C                       S T R U C T U R E   S T I F F N E S S                 00016900
   1  C                                                                             00017000
   1  C                                                                             00017100
   3    440 GO TO (520,540),ICOUNT                                                  00017200
   1  C                                                                             00017300
   3    520 DO 445 I=1,16                                                           00017400
   7        RE(I)=0.                                                                00017500
   3    445 EDIS(I)=0.                                                              00017600
   7        DO 500 N=1,NUME                                                         00017700
   7        MTYPE=MATP(N)                                                           00017800
   7        IEL=IELT(N)                                                             00017900
   7        THIC=THICK(N)                                                           00018000
   7        BET=BETA(N)                                                             00018100
   7        ND=2*IEL                                                                00018200
   7        NND5=IEL - 4                                                            00018300
   7        DO 240 I=1,136                                                          00018400
   3    240 S(I)=0.                                                                 00018500
   1  C                                                                             00018600
   1  C     CALCULATE STIFFNESS MATRIX                                              00018700
   1  C                                                                             00018800
   7        CALL QUADS (ND,B,S,YZ(1,N),PROP(1,MTYPE),RE,EDIS,                       00018900
   6       1            IDW,WA(1,N),NOD5(1,N))                                     00019000
   1  C                                                                             00019100
   7        CALL ADDBAN (A(N2),A(N1),S,LM(1,N),ND)                                  00019200
   3    500 CONTINUE                                                                00019300
   7        RETURN                                                                  00019400
   1  C                                                                             00019500
   1  C     ADD MASS EFFECT AND ADAMP*MASS PORTION OF RAYLEIGH DAMPING              00019600
   1  C                                                                             00019700
```

```
START                                                                                                  MOD
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8  FLAGS
   3    540 DO 640 N=1,NUME                                                                  00019800
   7        MTYPE=MATP(N)                                                                    00019900
   7        THIC=THICK(N)                                                                    00020000
   7        IEL=IELT(N)                                                                      00020100
   7        ND=2*IEL                                                                         00020200
   7        NND5=IEL - 4                                                                     00020300
   7        DE=DEN(MTYPE)                                                                    00020400
   1  C                                                                                      00020500
   7        CALL QUADM (N,ND,XM,S,YZ(1,N),NOD5(1,N))                                         00020600
   1  C                                                                                      00020700
   7        IF (IMASS.EQ.2) GO TO 280                                                        00020800
   7        IST=0                                                                            00020900
   7        FAC=A0 + A1*ADAMP                                                                00021000
   7        CALL ADDMA (A(N5),XM,LM(1,N),ND,FAC,IST)                                         00021100
   7        DO 260 I=1,ND                                                                    00021200
   3    260 XM(I)=XM(I)*FAC                                                                  00021300
   7        CALL ADDDI(A(N2),A(N1),XM,LM(1,N),ND)                                            00021400
   7        GO TO 640                                                                        00021500
   3    280 FAC=A0 + A1*ADAMP                                                                00021600
   7        DO 270 I=1,136                                                                   00021700
   3    270 S(I)=S(I)*FAC                                                                    00021800
   7        CALL ADDBAN (A(N2),A(N1),S,LM(1,N),ND)                                           00021900
   3    640 CONTINUE                                                                         00022000
   1  C                                                                                      00022100
   7        RETURN                                                                           00022200
   1  C                                                                                      00022300
   1  C                                                                                      00022400
   1  C     A S S E M B L E   M A S S   M A T R I X                                          00022500
   1  C                                                                                      00022600
   1  C                                                                                      00022700
   3    560 DO 660 N=1,NUME                                                                  00022800
   7        MTYPE=MATP(N)                                                                    00022900
   7        THIC=THICK(N)                                                                    00023000
   7        IEL=IELT(N)                                                                      00023100
   7        ND=2*IEL                                                                         00023200
   7        NND5=IEL - 4                                                                     00023300
   7        DE=DEN(MTYPE)                                                                    00023400
   1  C                                                                                      00023500
   7        CALL QUADM (N,ND,XM,S,YZ(1,N),NOD5(1,N))                                         00023600
   1  C                                                                                      00023700
   7        CALL ADDBAN (A(N2),A(N1),S,LM(1,N),ND)                                           00023800
   3    660 CONTINUE                                                                         00023900
   7        RETURN                                                                           00024000
   1  C                                                                                      00024100
   1  C                                                                                      00024200
   1  C     A S S E M B L E   N O N L I N E A R   F I N A L                                  00024300
   1  C     E F F E C T I V E   S T R U C T U R E   S T I F F N E S S                        00024400
   1  C                         A N D   L O A D S                                            00024500
   1  C                                                                                      00024600
```

```
START                                                                                MOD
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8  FLAGS

   1  C                                                                              00024700
   3    700 DO 710 N=1,NUME                                                          00024800
   7        MTYPE=MATP(N)                                                            00024900
   7        IEL=IELT(N)                                                              00025000
   7        THIC=THICK(N)                                                            00025100
   7        ND=2*IEL                                                                 00025200
   7        NND5=IEL - 4                                                             00025300
   7        DO 740 I=1,ND                                                            00025400
   7        RE(I)=0.0                                                                00025500
   7        EDIS(I)=0.                                                               00025600
   7        II=LM(I,N)                                                               00025700
   7        IF (II.LE.0) GO TO 740                                                   00025800
   7        EDIS(I)=Y(II)                                                            00025900
   3    740 CONTINUE                                                                 00026000
   7        DO 750 I=1,136                                                           00026100
   3    750 S(I)=0.                                                                  00026200
   1  C                                                                              00026300
   7        CALL QUADS (ND,B,S,YZ(1,N),PROP(1,MTYPE),RE,EDIS,                        00026400
   6       1            IDW,WA(1,N),NOD5(1,N))                                     00026500
   1  C                                                                              00026600
   7        DO 760 I=1,ND                                                            00026700
   7        II=LM(I,N)                                                               00026800
   7        IF (II.EQ.0) GO TO 760                                                   00026900
   7        HT(II)=HT(II)-RE(I)                                                      00027000
   3    760 CONTINUE                                                                 00027100
   1  C                                                                              00027200
   7        IF (ICOUNT-2) 745,745,710                                                00027300
   3    745 IF (IREF) 710,730,710                                                    00027400
   3    730 CALL ADDBAN (A(N4),A(N1),S,LM(1,N),ND)                                   00027500
   1  C                                                                              00027600
   3    710 CONTINUE                                                                 00027700
   7        RETURN                                                                   00027800
   1  C                                                                              00027900
   1  C                                                                              00028000
   1  C     S T R E S S   C A L C U L A T I O N S                                    00028100
   1  C                                                                              00028200
   1  C                                                                              00028300
   3    800 IST=4                                                                    00028400
   7        IF (ITYP2D.GT.0) IST=3                                                   00028500
   7        STRESS(4) = 0.0                                                          00028600
   7        STRAIN(4) = 0.0                                                          00028700
   1  C                                                                              00028800
   7        IF (NG.LE.NEGL) GO TO 801                                                00028900
   7        IF (MODEL.GT.2) GO TO 850                                                00029000
   1  C                                                                              00029100
   1  C     STRESS CALCULATIONS FOR ELEMENTS WITH CONSTANT MATERIAL VARIABLES         00029200
   1  C                                                                              00029300
   3    801 IPRNT=0                                                                  00029400
   7        DO 840 N=1,NUME                                                          00029500
```

```
START                                                                                                                  MOD
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8  FLAGS
   7        IPS=IPST(N)                                                                                    00029600
   7        IF (IPS.EQ.0) GO TO 840                                                                        00029700
   7        IPRNT=IPRNT + 1                                                                                00029800
   7        IF (IPRNT.NE.1) GO TO 802                                                                      00029900
   7        WRITE(KIO6,2020) NG                                                                            00030000
   7        IF (ITYP2D.EQ.0)                                                                               00030100
   6       *WRITE(KIO6,2022)                                                                               00030200
   7        IF (ITYP2D.EQ.1)WRITE(KIO6,2024)                                                               00030300
   7        IF (ITYP2D.EQ.2) WRITE(KIO6,2026)                                                              00030400
   7        WRITE(KIO6,2030)                                                                               00030500
   3    802 MTYPE=MATP(N)                                                                                  00030600
   7        IEL    = IELT(N)                                                                               00030700
   7        THIC   = THICK(N)                                                                              00030800
   7        ND     = 2*IEL                                                                                 00030900
   7        NND5=IEL - 4                                                                                   00031000
   1  C                                                                                                    00031100
   7        DO 805 I=1,ND                                                                                  00031200
   7        EDIS(I) = 0.0                                                                                  00031300
   7        II      = LM(I,N)                                                                              00031400
   7        IF(II.LE.0) GO TO 805                                                                          00031500
   7        EDIS(I) = Y(II)                                                                                00031600
   3    805 CONTINUE                                                                                       00031700
   1  C                                                                                                    00031800
   7        IF (INDNL.GT.2) GO TO 806                                                                      00031900
   7        DO 808 I=1,ND                                                                                  00032000
   3    808 XX(I)=YZ(I,N)                                                                                  00032100
   7        GO TO 807                                                                                      00032200
   3    806 DO 809 I=1,ND                                                                                  00032300
   3    809 XX(I)=YZ(I,N) + EDIS(I)                                                                        00032400
   1  C                                                                                                    00032500
   1  C      FORM THE LINEAR STRESS/STRAIN LAW                                                             00032600
   1  C                                                                                                    00032700
   3    807 CALL STSTL (N,XX,PROP(1,MTYPE),D)                                                              00032800
   1  C                                                                                                    00032900
   7        WRITE(KIO6,2035) N                                                                             00033000
   1  C                                                                                                    00033100
   1  C      CALCULATE AND PRINT ELEMENT STRESSES AT * IPS * LOCATIONS                                     00033200
   1  C                                                                                                    00033300
   7        IF (NTABLE.EQ.0) GO TO 831                                                                     00033400
   7        DO 830 II=1,9                                                                                  00033500
   7        I=ITABLE(IPS,II)                                                                               00033600
   7        IF (I.EQ.0) GO TO 830                                                                          00033700
   1  C                                                                                                    00033800
   7        CALL DERIQ (N,XX,B,DET,EVAL(I,1),EVAL(I,2),X1BAR,NOD5(1,N))                                    00033900
   1  C                                                                                                    00034000
   7        DO 810 J=1,5                                                                                   00034100
   3    810 DISD(J)=0.0                                                                                    00034200
   7        DO 815 J=2,ND,2                                                                                00034300
   7        JJ=J - 1                                                                                       00034400
```

```
START                                                                                                  MOD
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8  FLAGS
   7        DISD(1)=DISD(1) + B(1,JJ)*EDIS(JJ)                                          00034500
   7        DISD(2)=DISD(2) + B(2,J)*EDIS(J)                                            00034600
   7        DISD(3)=DISD(3) + B(3,JJ)*EDIS(JJ)                                          00034700
   3     815 DISD(4)=DISD(4) + B(3,J)*EDIS(J)                                           00034800
   7        IF (IST.EQ.3) GO TO 819                                                     00034900
   7        DO 818 J=1,ND,2                                                             00035000
   3     818 DISD(5)=DISD(5) + B(4,J)*EDIS(J)                                           00035100
   1  C                                                                                 00035200
   1  C                                                                                 00035300
   3     819 IPT=1                                                                      00035400
   7        CALL STSTN (XX,PROP(1,MTYPE),DISD,                                          00035500
   6       1            IDW,WA(1,N))                                                   00035600
   1  C                                                                                 00035700
   1  C     TRANSFORM PIOLA-KIRKHHOFF STRESSES TO CAUCHY STRESSES                       00035800
   1  C                                                                                 00035900
   1  C        CS = (1./DET(F)) * ( F * PK * F(TRANSPOSED) )                            00036000
   1  C                                                                                 00036100
   7        IF (KLIN.EQ.0) GO TO 822                                                    00036200
   7        IF (INDNL.NE.2) GO TO 822                                                   00036300
   1  C                                                                                 00036400
   7        CALL CAUCHY                                                                 00036500
   1  C                                                                                 00036600
   1  C     COMPUTE PRINCIPAL STRESSES AND DIRECTIONS                                   00036700
   1  C                                                                                 00036800
   3     822 CALL MAXMIN (STRESS,P1,P2,AG)                                              00036900
   1  C                                                                                 00037000
   7        WRITE(KIO6,2040)                                                            00037100
   6       *              I,STRESS,P1,P2,AG                                            00037200
   1  C                                                                                 00037300
   3     830 CONTINUE                                                                   00037400
   7        GO TO 840                                                                   00037500
   1  C                                                                                 00037600
   1  C     CALCULATE AND PRINT ELEMENT STRESSES AT INTEGRATION POINTS                  00037700
   1  C                                                                                 00037800
   3     831 DO 839 LX=1,NINT                                                           00037900
   7        E1=XG(LX,NINT)                                                              00038000
   7        DO 839 LY=1,NINT                                                            00038100
   7        E2=XG(LY,NINT)                                                              00038200
   7        IPT=(LX-1)*NINT + LY                                                        00038300
   1  C                                                                                 00038400
   7        CALL DERIQ (N,XX,B,DET,E1,E2,X1BAR,NOD5(1,N))                               00038500
   1  C                                                                                 00038600
   7        DO 832 J=1,5                                                                00038700
   3     832 DISD(J)=0.0                                                                00038800
   7        DO 833 J=2,ND,2                                                             00038900
   7        JJ=J - 1                                                                    00039000
   7        DISD(1)=DISD(1) + B(1,JJ)*EDIS(JJ)                                          00039100
   7        DISD(2)=DISD(2) + B(2,J)*EDIS(J)                                            00039200
   7        DISD(3)=DISD(3) + B(3,JJ)*EDIS(JJ)                                          00039300
```

```
START                                                                                                    MOD
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8                   FLAGS
   3    833 DISD(4)=DISD(4) + B(3,J)*EDIS(J)                                                          00039400
   7        IF (IST.EQ.3) GO TO 835                                                                   00039500
   7        DO 834 J=1,ND,2                                                                           00039600
   3    834 DISD(5)=DISD(5) + B(4,J)*EDIS(J)                                                          00039700
                                                                                                      00039800
   1  C
   3    835 CALL STSTN (XX,PROP(1,MTYPE),DISD,IDW,WA(1,N))                                            00039900
                                                                                                      00040000
   1  C
   1  C     TRANSFORM PIOLA-KIRKHHOFF STRESSES TO CAUCHY STRESSES                                     00040100
                                                                                                      00040200
   1  C
   1  C        CS = (1./DET(F)) * ( F * PK * F(TRANSPOSED) )                                          00040300
   1  C                                                                                              00040400
   7        IF (KLIN.EQ.0) GO TO 827                                                                  00040500
   7        IF (INDNL.NE.2) GO TO 827                                                                 00040600
   1  C                                                                                              00040700
   7        CALL CAUCHY                                                                               00040800
                                                                                                      00040900
   1  C
   1  C     COMPUTE PRINCIPAL STRESSES AND DIRECTIONS                                                 00041000
   1  C                                                                                              00041100
   3    827 CALL MAXMIN (STRESS,P1,P2,AG)                                                             00041200
                                                                                                      00041300
   1  C
   7        WRITE(KIO6,2040) IPT,STRESS,P1,P2,AG                                                      00041400
                                                                                                      00041500
   1  C
   3    839 CONTINUE                                                                                  00041600
                                                                                                      00041700
   1  C
   3    840 CONTINUE                                                                                  00041800
   7        RETURN                                                                                    00041900
   1  C                                                                                              00042000
   1  C     NONLINEAR MATERIAL MODEL STRESS CALCULATIONS                                              00042100
   1  C                                                                                              00042200
   3    850 IPRNT=0                                                                                   00042300
   7        DO 890 N=1,NUME                                                                           00042400
   7        IF (IPST(N).EQ.0) GO TO 890                                                               00042500
   7        IPRNT=IPRNT + 1                                                                           00042600
   7        IF (IPRNT.NE.1) GO TO 852                                                                 00042700
   7        WRITE(KIO6,2020) NG                                                                       00042800
   7        IF (ITYP2D.EQ.0) WRITE(KIO6,2022)                                                         00042900
   7        IF (ITYP2D.EQ.1)WRITE(KIO6,2024)                                                          00043000
   7        IF (ITYP2D.EQ.2)WRITE(KIO6,2026)                                                          00043100
   3    852 MTYPE=MATP(N)                                                                             00043200
   7        IEL=IELT(N)                                                                               00043300
   7        THIC=THICK(N)                                                                             00043400
   7        ND=2*IEL                                                                                  00043500
   7        NND5=IEL - 4                                                                              00043600
                                                                                                      00043700
   1  C
   7        DO 860 I=1,ND                                                                             00043800
   7        EDIS(I)=0.                                                                                00043900
   7        II=LM(I,N)                                                                                00044000
   7        IF (II.LE.0) GO TO 860                                                                    00044100
   7        EDIS(I)=Y(II)                                                                             00044200
```

```
START                                                                                MOD
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8  FLAGS
   3    860 CONTINUE                                                                  00044300
   1  C                                                                               00044400
   7        IF (INDNL.GT.2) GO TO 866                                                 00044500
   7        DO 868 I=1,ND                                                             00044600
   3    868 XX(I)=YZ(I,N)                                                             00044700
   7        GO TO 867                                                                 00044800
   3    866 DO 869 I=1,ND                                                             00044900
   3    869 XX(I)=YZ(I,N) + EDIS(I)                                                   00045000
   1  C                                                                               00045100
   1  C     CALCULATE AND PRINT STRESSES AT INTEGRATION POINTS                        00045200
   1  C                                                                               00045300
   3    867 DO 885 LX=1,NINT                                                          00045400
   7        E1 = XG(LX,NINT)                                                          00045500
   7        DO 885 LY=1,NINT                                                          00045600
   7        E2 = XG(LY,NINT)                                                          00045700
   7        IPT = (LX-1)*NINT + LY                                                    00045800
   1  C                                                                               00045900
   7        CALL DERIQ (N,XX,B,DET,E1,E2,X1BAR,NOD5(1,N))                             00046000
   1  C                                                                               00046100
   7        DO 870 I=1,5                                                              00046200
   3    870 DISD(I)=0.0                                                               00046300
   7        DO 875 J=2,ND,2                                                           00046400
   7        I=J - 1                                                                   00046500
   7        DISD(1)=DISD(1) + B(1,I)*EDIS(I)                                          00046600
   7        DISD(2)=DISD(2) + B(2,J)*EDIS(J)                                          00046700
   7        DISD(3)=DISD(3) + B(3,I)*EDIS(I)                                          00046800
   3    875 DISD(4)=DISD(4) + B(3,J)*EDIS(J)                                          00046900
   7        IF (IST.EQ.3) GO TO 879                                                   00047000
   7        DO 878 I=1,ND,2                                                           00047100
   3    878 DISD(5)=DISD(5) + B(4,I)*EDIS(I)                                          00047200
   1  C                                                                               00047300
   1  C                                                                               00047400
   3    879 CALL STSTN (XX,PROP(1,MTYPE),DISD,                                        00047500
   6       1            IDW,WA(1,N))                                                 00047600
   1  C                                                                               00047700
   3    885 CONTINUE                                                                  00047800
   1  C                                                                               00047900
   3    890 CONTINUE                                                                  00048000
   1  C                                                                               00048100
   7        RETURN                                                                    00048200
   1  C                                                                               00048300
   1  C                                                                               00048400
   2   1000 FORMAT (I5,F10.0)                                                         00048500
   2   1001 FORMAT (8F10.0)                                                           00048600
   2   1002 FORMAT (9I5)                                                              00048700
   2   1004 FORMAT (I5,I3,I2,2F10.0,10I5)                                             00048800
   2   2004 FORMAT (1H ,2I5,5X,I1,1X,2(F10.0,1X),2I6,9X,I4,7(4X,I4))                  00048900
   2   2005 FORMAT (////4X,20H ELEMENT INFORMATION ,                                  00049000
   6       1        //52H     M   IEL   IPS       BET       THIC    MTYP   KG,00049100
```

```
START                                                                    MOD
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8  FLAGS
   6        2          10X,8(A4,I1,3X))                                              00049200
   2    2010 FORMAT (///12H *** ELEMENT,I5,46H+EXCEEDS MAXIMUM NUMBER OF NODES       00049300
   6       1(NPAR(4)) ***)                                                          00049400
   2    2015 FORMAT (32H MAX NUMBER OF NODES DESCRIBING /,                          00049500
   6        1         20H     ANY ONE ELEMENT,10(2H .),16H( NPAR(7) ). . =,I5//,00049600
   6        2         33H NUMBER OF INTEGRATION POINTS FOR/,                      00049700
   6        3         40H     ELEMENT STIFFNESS GENERATION. . . .,                00049800
   6        4         16H( NPAR(10)). . =,I5//,                                   00049900
   6        5         40H NUMBER OF STRESS OUTPUT TABLES  . . . .,                00050000
   6        6         16H( NPAR(13)). . =,I5/                                     00050100
   6        7         40H     EQ.0, PRINT AT INTEGRATION POINTS      //)          00050200
   2    2020 FORMAT (1H1,46H S T R E S S   C A L C U L A T I O N S   F O R,3X,  00050300
   6       1         25HE L E M E N T   G R O U P,3X,I2,3X,15H(2/D CONTINUUM) )00050400
   2    2022 FORMAT (82X,14H(AXISYMMETRIC), // 1X)                                  00050500
   2    2024 FORMAT (82X,14H(PLANE STRAIN), // 1X)                                  00050600
   2    2026 FORMAT (82X,14H(PLANE STRESS), // 1X)                                  00050700
   2    2030 FORMAT (8H ELEMENT,4X,6HOUTPUT,/ 2X,6HNUMBER,2X,8HLOCATION,7X,     00050800
   6       1         8HSIGMA-X1,7X,8HSIGMA-X2,8X,7HTAU-X12,7X,8HSIGMA-X3,        00050900
   6       2         7X,8HSIGMA-P+,7X,8HSIGMA-P-,5X,5HANGLE, / 1X)               00051000
   2    2035 FORMAT (I8)                                                            00051100
   2    2040 FORMAT (13X,I5,6F15.4,F10.2)                                           00051200
   2    2050 FORMAT (36H E L E M E N T   D E F I N I T I O N ///,                   00051300
   6       1         14H ELEMENT TYPE ,13(2H .),16H( NPAR(1) ). . =,I5/,         00051400
   6       2         25H     EQ.1, TRUSS ELEMENTS/,                              00051500
   6       3         25H     EQ.2, 2-DIM ELEMENTS/,                              00051600
   6       3         25H     EQ.3, 3-DIM ELEMENTS//,                             00051700
   6       4         20H NUMBER OF ELEMENTS.,10(2H .),16H( NPAR(2) ). . =,I5//,00051800
   6       5         40H TYPE OF NONLINEAR ANALYSIS  . . . . . .     ,           00051900
   6       6         16H( NPAR(3) ). . =,I5/,                                    00052000
   6       7         40H     EQ.1, MATERIAL NONLINEARITY ONLY    /,              00052100
   6       7         40H     EQ.2, TOTAL LAGRANGIAN FORMULATION   /,             00052200
   6       8         44H     EQ.3, UPDATED LAGRANGIAN FORMULATION          //,00052300
   6       4         16H ELEMENT SUBTYPE,12(2H .),16H( NPAR(5) ). . =,I5/,       00052400
   6       5         32H     EQ.0, AXISYMMETRIC ELEMENTS/,                       00052500
   6       6         32H     EQ.1, PLANE STRAIN ELEMENTS/,                       00052600
   6       7         32H     EQ.2, PLANE STRESS ELEMENTS/)                       00052700
                                                                                  00052800
   2    2060 FORMAT (10I10)                                                         00052900
   2    2070 FORMAT (//40H S T R E S S   O U T P U T   T A B L E S   //          00053000
   6       1         10H     TABLE,9X,1H1,9X,1H2,9X,1H3,9X,1H4,9X,1H5,9X,1H6,    00053100
   6       2         9X,1H7,9X,1H8,9X,1H9/)                                      00053200
   1    C                                                                         00053300
   7         END                                                                  00054000
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8  MOD FLAGS
   7        SUBROUTINE THDFE (ID,X,Y,Z,HT,LM,XYZ,IELTD,IELTX,IPST,MATP,          00000100
   6       1                  NOD9,IREUSE,DEN,PROP,WA,S,XM,B,RE,EDIS,ITABLE,    00000200
   6       2                  NTABLE,NCON,IDWA,NDM,NDM2,NDOF,ND9DIM)            00000300
   7        IMPLICIT REAL*8(A-H,O-Z)                                            00000400
   1  C                                                                         00000500
   1  C                                                                         00000600
   7        COMMON/BBCCIO/KIO1,KIO2,KIO3,KIO4,KIO5,KIO6,KIO7,KIO8,KIO9,       00000700
   6       1KIO10,KIO11,KIO12,KIO13,KIO14,KIO15,KIO16,KIO17,KIO18,KIO19,KIO20,00000800
   6       2KIO21,KIO22,KIO23,KIO24,KIO25,KIO26,KIO27,KIO28,KIO29,KIO30       00000900
                                                                               00001000
   1  C                                                                         00001100
   7        COMMON /EL/ IND,ICOUNT,NPAR(20),NUMEG,NEGL,NEGNL,IMASS,IDAMP,ISTAT00001100
   6       1           ,NDOFDM,KLIN,IEIG,IMASSN,IDAMPN                         00001200
   7        COMMON /SOL/ NUMNP,NEQ,NWK,NWM,NWC,NUMEST,MIDEST,MAXEST,NSTE,MA   00001300
   7        COMMON /DIM/ N1,N2,N3,N4,N5,N6,N7,N8,N9,N10,N11,N12,N13,N14,N15   00001400
   7        COMMON /CONST/ DT,DTA,A0,A1,A2,A3,A4,A5,A6,A7,A8,A9,A10,A11       00001500
   6       1              ,A12,A13,A14,A15,A16,A17,A18,A19,A20,IOPE           00001600
                                                                               00001700
   7        COMMON /DAMPNG/ ADAMP,BDAMP                                       00001800
   7        COMMON /VAR/ NG,KPRI,MODEX,KSTEP,ITE,ITEMAX,IREF,IEQREF,INOCMD    00001900
   7        COMMON /THREED/ DE,IELD,IELX,IEL,NPT,IDW,NND9                     00002000
   7        COMMON /MTMD3D/  D(6,6),STRESS(6),STRAIN(6),IPT,N                 00002100
   7        COMMON A(1)                                                       00002200
   1  C                                                                         00002300
   7        DIMENSION ID(NDOF,1),X(1),Y(1),Z(1),HT(1),LM(NDM,1),XYZ(NDM,I),   00002400
   6       1          IELTD(1),IELTX(1),IPST(1),MATP(1),DEN(1),              00002500
   6       2          PROP(NCON,1),WA(IDWA,1),S(1),XM(1),B(1),RE(1),         00002600
   6       3          EDIS(1),NOD9(ND9DIM,1),IREUSE(1),ITABLE(NTABLE,1)      00002700
   7        DIMENSION EVAL(27,3),NOD(21),NODM(21),NOD9M(13),DISD(9)           00002800
   7        DIMENSION XG(4,4)                                                 00002900
   7        EQUIVALENCE (NPAR(2),NUME),(NPAR(16),NUMMAT),(NPAR(15),MODEL)     00003000
   6       1           ,(NPAR(10),NINT),(NPAR(11),NINTZ),(NPAR(7),MXNODS)    00003100
   6       2           ,(NPAR(3),INDNL)                                      00003201  **
   1  C    NODAL VALUES OF SHAPE FUNCTIONS ARE CORRECTED IN DATA BLOCK 'EVAL'  00003301  **
   1  C    WEIGHTS FOR GAUSS INTEGRATION ARE CORRECTED IN DATA BLOCK 'XG'      00003401  **
   1  C    S.N.MUKHERJEE.BBC,BADEN OCT 1977.                                  00003500
   7        DATA XG /     0.,              0.,              0.,             0., 00003600
   6       1 -.5773502691896, .5773502691896,              0.,             0., 00003700
   6       2 -.7745966692415, .0000000000000, .7745966692415,             0., 00003800
   6       3 -.8611363115941,-.3399810435849, .3399810435849, .8611363115941/ 00003900
   7        DATA EVAL / 1.,-1.,-1., 1., 1.,-1.,-1., 1., 0.,-1., 0., 1.,       00004000
   6       1            0.,-1., 0., 1., 1.,-1.,-1., 1., 0., 0.,-1., 0.,     00004100
   6       2            1., 0., 0.,                                          00004200
   6       3            1., 1.,-1.,-1., 1., 1.,-1.,-1., 1., 0.,-1., 0.,     00004300
   6       4            1., 0.,-1., 0., 1., 1.,-1.,-1., 0., 1., 0.,-1.,     00004400
   6       5            0., 0., 0.,                                          00004500
   6       6            1., 1., 1., 1.,-1.,-1.,-1.,-1., 1., 1., 1., 1.,     00004600
   6       7           -1.,-1.,-1.,-1., 0., 0., 0., 0., 0., 0., 0., 0.,     00004700
   6       8            0., 1.,-1. /                                         00004800
   7        DATA ANODE/4HNODE/                                                00004900
   1  C
```

```
START                                                                                MOD
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8  FLAGS
   1  C     ** NOTE ** DURING TIME INTEGRATION X=DISP, Y=VEL, Z=ACC, HT=R      00005000
   1  C                                                                        00005100
   1  C                                                                        00005200
   1  C                                                                        00005300
   7        IF (KPRI.EQ.0) GO TO 800                                          00005400
   7        IF (IND.GT.0) GO TO 420                                           00005500
   1  C                                                                        00005600
   1  C                                                                        00005700
   1  C     R E A D   A N D   G E N E R A T E   E L E M E N T                  00005800
   1  C     I N F O R M A T I O N                                              00005900
   1  C     NEW TABLES ARE ADDED FOR ELEMENT STRESS OUTPUT LOCATION            00006000
   1  C     S.N.MUKHERJEE.BBC,BADEN. SEPT.1977                                 00006100
   7        NPT=NINT*NINT*NINTZ                                               00006200
   7        IDW=IDWA/NPT                                                      00006300
   7        IF (INDNL.EQ.1) GO TO 5                                           00006400
   7        WRITE(KIO6,3000)                                                  00006500
   7        STOP                                                              00006600
   5      5 WRITE(KIO6,2050)                                                  00006700
   6       *               NPAR(1),NUME,INDNL                                 00006800
   7        WRITE(KIO6,2051)                                                  00006900
   6       *                MXNODS,NINT,NINTZ,NTABLE                          00007000
   7        DO 10 I=1,NUMMAT                                                  00007100
   7        READ(KIO5,1000) N,DEN(N)                                          00007200
   7        READ(KIO5,1001)                                                   00007300
   6       *               (PROP(J,N), J=1,NCON)                              00007400
   4     10 CALL MATWRT (N,DEN,PROP(1,N))                                     00007500
   1  C                                                                        00007600
   1  C     READ TABLES FOR ELEMENT STRESS OUTPUT LOCATIONS                    00007700
   1  C                                                                        00007800
   7        IF (NTABLE.EQ.0) GO TO 90                                         00007900
   7        WRITE(KIO6,2070)                                                  00008000
   7        DO 25 L=1,NTABLE                                                  00008100
   7        READ(KIO5,1007)                                                   00008200
   6       *               (ITABLE(L,I),I=1,16)                               00008300
   4     25 WRITE(KIO6,2060)                                                  00008400
   6       *               L,(ITABLE(L,I),I=1,16)                             00008500
                                                                              00008600
   1  C                                                                        00008700
   1  C     READ ELEMENT INFORMATION                                           00008800
   1  C                                                                        00008900
   4     90 IELN=8                                                            00009000
   7        IF (MXNODS.GT.8) IELN=21                                          00009100
   7        WRITE(KIO6,2005)                                                  00009200
   6       *               (ANODE,I,I=1,IELN)                                 00009300
   7        N=1                                                               00009400
   3    100 READ(KIO5,1004)                                                  00009500
   6       *               M,IELD,IELX,IPS,MTYP,IST,KG                        00009600
   7        IF (IELD.EQ.0) IELD=MXNODS                                        00009700
   7        IF (IELX.EQ.0) IELX=IELD                                          00009800
   7        IEL=IELD
```

```
START                                                                    MOD
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8  FLAGS
   7        IF (IELX.GT.IELD) IELX=IELD                                  00009900
   7        READ(KIO5,1005) (NOD(I),I=1,8)                               00010000
   7        READ(KIO5,1005) (NOD(I),I=9,21)                              00010100
   7        IF (NDM.GE.IEL*3) GO TO 105                                  00010200
   7        WRITE(KIO6,2010) M                                           00010300
   7        STOP                                                         00010400
   3    105 IF (KG.EQ.0) KG=1                                            00010500
   1    C                                                                00010600
   3    120 IF (M.NE.N) GO TO 200                                        00010700
   7        DO 110 I=1,IELN                                              00010800
   3    110 NODM(I)=NOD(I)                                               00010900
   7        IF (IEL.EQ.8) GO TO 115                                      00011000
   7        II=0                                                         00011100
   7        DO 114 I=9,21                                                00011200
   7        NN=NOD(I)                                                    00011300
   7        IF (NN.EQ.0) GO TO 114                                       00011400
   7        II=II + 1                                                    00011500
   7        NOD9M(II)=I                                                  00011600
   3    114 CONTINUE                                                     00011700
   7        NN=II + 8                                                    00011800
   7        IF (NN.EQ.IEL) GO TO 115                                     00011900
   7        WRITE(KIO6,2090) N                                           00012000
   7        STOP                                                         00012100
   3    115 IELM=IEL                                                     00012200
   7        IELDM=IELD                                                   00012300
   7        IELXM=IELX                                                   00012400
   7        IPSM=IPS                                                     00012500
   7        MTYPE=MTYP                                                   00012600
   7        ISTM=IST                                                     00012700
   7        KKK=KG                                                       00012800
   1    C                                                                00012900
   1    C       SAVE ELEMENT INFORMATION                                 00013000
   1    C                                                                00013100
   3    200 I2=0                                                         00013200
   7        DO 130 I=1,IELM                                              00013300
   7        II=NODM(I)                                                   00013400
   7        IF (I.LE.8) GO TO 131                                        00013500
   7        JJ=NOD9M(I-8)                                                00013600
   7        II=NODM(JJ)                                                  00013700
   3    131 I2=I2 + 3                                                    00013800
   7        XYZ(I2-2,N)=X(II)                                            00013900
   7        XYZ(I2-1,N)=Y(II)                                            00014000
   3    130 XYZ(I2,N)=Z(II)                                              00014100
   7        MATP(N)=MTYPE                                                00014200
   7        IELTD(N)=IELDM                                               00014300
   7        IELTX(N)=IELXM                                               00014400
   7        IPST(N)=IPSM                                                 00014500
   7        IREUSE(N)=ISTM                                               00014600
   7        IF (IELM.EQ.8) GO TO 135                                     00014700
```

```
START                                                                    MOD
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8  FLAGS
   7        NN=IELM - 8                                                          00014800
   7        DO 132 I=1,NN                                                        00014900
   3    132 NOD9(I,N)=NOD9M(I)                                                   00015000
   3    135 KK=-3                                                                00015100
   7        DO 140 I=1,IELM                                                      00015200
   7        II=NODM(I)                                                           00015300
   7        IF (I.LE.8) GO TO 137                                                00015400
   7        JJ=NOD9M(I-8)                                                        00015500
   7        II=NODM(JJ)                                                          00015600
   3    137 KK=KK + 3                                                            00015700
   7        DO 140 L=1,3                                                         00015800
   3    140 LM(KK+L,N)=ID(L,II)                                                  00015900
   1  C                                                                          00016000
   1  C     UPDATE COLUMN HEIGHTS AND BANDWIDTH                                  00016100
   1  C                                                                          00016200
   7        ND=IELM*3                                                            00016300
   7        CALL COLHT(HT,ND,LM(1,N))                                            00016400
   1  C                                                                          00016500
   1  C     INITIALIZE STORAGE AND PRINT ELEMENT INFORMATION                     00016600
   1  C                                                                          00016700
   7        CALL INTWA3 (MODEL)                                                  00016800
   7        WRITE(KIO6,2004)                                                     00016900
   6       *              N,IELDM,IELXM,IPSM,MTYPE,ISTM,KKK,                    00017000
   6       1              (NODM(I),I=1,IELN)                                    00017100
   1  C                                                                          00017200
   7        IF (N.EQ.NUME) RETURN                                                00017300
   7        N=N+1                                                                00017400
   7        DO 160 I=1,IELN                                                      00017500
   7        IF (NODM(I).EQ.0) GO TO 160                                          00017600
   7        NODM(I)=NODM(I) + KKK                                                00017700
   3    160 CONTINUE                                                             00017800
   7        IF (N.GT.M) GO TO 100                                                00017900
   7        GO TO 120                                                            00018000
   1  C                                                                          00018100
   3    420 GO TO (440,560,560,700), IND                                         00018200
   1  C                                                                          00018300
   1  C                                                                          00018400
   1  C     A S S E M B L E  E F F E C T I V E  L I N E A R                      00018500
   1  C     S T R U C T U R A L  S T I F F N E S S                                00018600
   1  C                                                                          00018700
   1  C                                                                          00018800
   3    440 GO TO (520,540) ,ICOUNT                                              00018900
   1  C                                                                          00019000
   3    520 DO 445 I=1,NDM                                                       00019100
   7        RE(I)=0.0                                                            00019200
   3    445 EDIS(I)=0.0                                                          00019300
   7        NPT=NINT*NINT*NINTZ                                                  00019400
   7        DO 500 N=1,NUME                                                      00019500
   7        MTYPE=MATP(N)                                                        00019600
```

```
START COL ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8  MOD FLAGS
   7          IELD=IELTD(N)                                                                00019700
   7          IELX=IELTX(N)                                                                00019800
   7          IEL=IELD                                                                     00019900
   7          IST=IREUSE(N)                                                                00020000
   7          ND=3*IELD                                                                    00020100
   7          IF (IST.NE.0) GO TO 525                                                      00020200
   7          NND9=IELD - 8                                                                00020300
   7          DO 240 I=1,NDM2                                                              00020400
   3     240  S(I)=0.0                                                                     00020500
   1  C                                                                                    00020600
   1  C     CALCULATE STIFFNESS MATRIX                                                     00020700
   1  C                                                                                    00020800
   7          CALL QUADS3  (ND,B,S,XYZ(1,N),PROP(1,MTYPE),RE,EDIS,                         00020900
   6         1              WA(1,N),NOD9(1,N))                                            00021000
   1  C                                                                                    00021100
   3     525  CONTINUE                                                                     00021200
   7          CALL ADDBAN (A(N2),A(N1),S,LM(1,N),ND)                                       00021300
   3     500  CONTINUE                                                                     00021400
   7          RETURN                                                                       00021500
   1  C                                                                                    00021600
   1  C     ADD MASS EFFECT AND ADAMP*MASS PORTION OF RAYLEIGH DAMPING                     00021700
   1  C                                                                                    00021800
   3     540  DO 640 N=1,NUME                                                              00021900
   7          MTYPE=MATP(N)                                                                00022000
   7          IELD=IELTD(N)                                                                00022100
   7          IELX=IELTX(N)                                                                00022200
   7          IEL=IELD                                                                     00022300
   7          IST=IREUSE(N)                                                                00022400
   7          ND=3*IELD                                                                    00022500
   7          NND9=IELD - 8                                                                00022600
   7          DE=DEN(MTYPE)                                                                00022700
   7          IF (IST.NE.0) GO TO 550                                                      00022800
   1  C                                                                                    00022900
   7          CALL QUADM3  (N,ND,NDM2,XM,S,XYZ(1,N),NOD9(1,N))                             00023000
   1  C                                                                                    00023100
   3     550  IF (IMASS.EQ.2) GO TO 280                                                   00023200
   7          FAC=A0 + A1*ADAMP                                                            00023300
   7          CALL ADDMA (A(N5),XM,LM(1,N),ND,FAC,IST)                                     00023400
   7          IF (IST.NE.0) GO TO 265                                                      00023500
   7          DO 260 I=1,ND                                                                00023600
   3     260  XM(I)=XM(I)*FAC                                                              00023700
   3     265  CONTINUE                                                                     00023800
   7          CALL ADDDI (A(N2),A(N1),XM,LM(1,N),ND)                                       00023900
   7          GO TO 640                                                                    00024000
   3     280  IF (IST.NE.0) GO TO 275                                                     00024100
   7          FAC=A0 + A1*ADAMP                                                            00024200
   7          DO 270 I=1,NDM2                                                              00024300
   3     270  S(I)=S(I)*FAC                                                               00024400
   3     275  CONTINUE                                                                     00024500
```

```
START                                                                                  MOD
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8  FLAGS
   7          CALL ADDBAN (A(N2),A(N1),S,LM(1,N),ND)                                  00024600
   3     640  CONTINUE                                                                00024700
   1  C                                                                               00024800
   7          RETURN                                                                  00024900
   1  C                                                                               00025000
   1  C                                                                               00025100
   1  C     A S S E M B L E  M A S S  M A T R I X                                     00025200
   1  C                                                                               00025300
   1  C                                                                               00025400
   1  C                                                                               00025500
   3     560  DO 660 N=1,NUME                                                        00025600
   7          MTYPE=MATP(N)                                                           00025700
   7          IELD=IELTD(N)                                                           00025800
   7          IELX=IELTX(N)                                                           00025900
   7          IEL=IELD                                                                00026000
   7          IST=IREUSE(N)                                                           00026100
   7          ND=3*IELD                                                               00026200
   7          NND9=IELD - 8                                                           00026300
   7          DE=DEN(MTYPE)                                                           00026400
   7          IF (IST.NE.0) GO TO 320                                                 00026500
   1  C                                                                               00026600
   7          CALL QUADM3  (N,ND,NDM2,XM,S,XYZ(1,N),NOD9(1,N))                        00026700
   1  C                                                                               00026800
   3     320  CALL ADDBAN (A(N2),A(N1),S,LM(1,N),ND)                                 00026900
   1  C                                                                               00027000
   3     660  CONTINUE                                                                00027100
   7          RETURN                                                                  00027200
   1  C                                                                               00027300
   1  C                                                                               00027400
   1  C     A S S E M B L E  N O N L I N E A R  F I N A L                             00027500
   1  C     E F F E C T I V E  S T R U C T U R A L  S T I F F N E S S                 00027600
   1  C                                  A N D  L O A D S                             00027700
   1  C                                                                               00027800
   1  C                                                                               00027900
   3     700  DO 710 N=1,NUME                                                        00028000
   7          MTYPE=MATP(N)                                                           00028100
   7          IELD=IELTD(N)                                                           00028200
   7          IELX=IELTX(N)                                                           00028300
   7          IEL=IELD                                                                00028400
   7          ND=3*IELD                                                               00028500
   7          NND9=IELD - 8                                                           00028600
   7          DO 740 I=1,ND                                                           00028700
   7          RE(I)=0.0                                                               00028800
   7          EDIS(I)=0.0                                                             00028900
   7          II=LM(I,N)                                                              00029000
   7          IF (II.LE.0) GO TO 740                                                  00029100
   7          EDIS(I)=X(II)                                                           00029200
   3     740  CONTINUE                                                                00029300
   7          DO 750 I=1,NDM2                                                         00029400
   3     750  S(I)=0.0
```

```
START                                                                                MOD
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8  FLAGS
   1  C                                                                          00029500
   7        CALL QUADS3  (ND,B,S,XYZ(1,N),PROP(1,MTYPE),RE,EDIS,                00029600
   6       1              WA(1,N),NOD9(1,N))                                    00029700
   1  C                                                                          00029800
   7        DO 760 I=1,ND                                                      00029900
   7        II=LM(I,N)                                                         00030000
   7        IF (II.EQ.0) GO TO 760                                             00030100
   7        HT(II)=HT(II) - RE(I)                                              00030200
   3     760 CONTINUE                                                          00030300
   1  C                                                                          00030400
   7        IF (ICOUNT - 2) 745,745,710                                        00030500
   3     745 IF (IREF) 710,730,710                                             00030600
   3     730 CALL ADDBAN (A(N4),A(N1),S,LM(1,N),ND)                            00030700
   1  C                                                                          00030800
   3     710 CONTINUE                                                          00030900
   7        RETURN                                                             00031000
   1  C                                                                          00031100
   1  C                                                                          00031200
   1  C     S T R E S S  C A L C U L A T I O N S                                 00031300
   1  C                                                                          00031400
   1  C                                                                          00031500
   3     800 IF (NG.LE.NEGL) GO TO 801                                         00031600
   7        IF (MODEL.GT.2) GO TO 850                                          00031700
   1  C                                                                          00031800
   1  C     STRESS CALCULATIONS FOR LINEAR MATERIAL MODELS                       00031900
   1  C                                                                          00032000
   3     801 IPRNT=0                                                           00032100
   7        DO 840 N=1,NUME                                                    00032200
   7        IPS=IPST(N)                                                        00032300
   7        IF (IPS.EQ.0) GO TO 840                                            00032400
   7        IPRNT=IPRNT + 1                                                    00032500
   7        IF (IPRNT.NE.1) GO TO 802                                          00032600
   7        WRITE(KIO6,2020) NG                                                00032700
   7        WRITE(KIO6,2030)                                                   00032800
   3     802 MTYPE=MATP(N)                                                     00032900
   7        IELD=IELTD(N)                                                      00033000
   7        IELX=IELTX(N)                                                      00033100
   7        IEL=IELD                                                           00033200
   7        ND=3*IEL                                                           00033300
   7        NND9=IELD - 8                                                      00033400
   1  C                                                                          00033500
   7        DO 805 I=1,ND                                                      00033600
   7        EDIS(I)=0.                                                         00033700
   7        II=LM(I,N)                                                         00033800
   7        IF (II.LE.0) GO TO 805                                             00033900
   7        EDIS(I)=X(II)                                                      00034000
   3     805 CONTINUE                                                          00034100
   1  C                                                                          00034200
   1  C     FORM LINEAR STRESS-STRAIN LAW                                        00034300
```

```
START                                                                                                  MOD
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8  FLAGS
   1  C                                                                                  00034400
   7        CALL STST3L  (N,XYZ(1,N),PROP(1,MTYPE),D)                                   00034500
   1  C                                                                                  00034600
   7        WRITE(KIO6,2035) N                                                          00034700
   1  C                                                                                  00034800
   1  C     CALCULATE AND PRINT ELEMENT STRESSES AT IPS LOCATIONS                        00034900
   1  C                                                                                  00035000
   7        IF (NTABLE.EQ.0) GO TO 831                                                  00035100
   7        DO 830 II=1,16                                                              00035200
   7        I=ITABLE(IPS,II)                                                            00035300
   7        IF (I.EQ.0) GO TO 840                                                       00035400
   7        CALL DERIQ3  (N,XYZ(1,N),B,DET,EVAL(I,1),EVAL(I,2),EVAL(I,3),               00035500
   6       1              NOD9(1,N))                                                   00035600
   1  C                                                                                  00035700
   7        DO 810 J=1,6                                                                00035800
   3    810 STRAIN(J)=0.                                                                00035900
   7        DO 815 K=3,ND,3                                                             00036000
   7        L=K - 2                                                                     00036100
   7        J=K - 1                                                                     00036200
   7        STRAIN(1)=STRAIN(1) + B(L)*EDIS(L)                                          00036300
   7        STRAIN(2)=STRAIN(2) + B(J)*EDIS(J)                                          00036400
   7        STRAIN(3)=STRAIN(3) + B(K)*EDIS(K)                                          00036500
   7        STRAIN(4)=STRAIN(4) + B(J)*EDIS(L) + B(L)*EDIS(J)                           00036600
   7        STRAIN(5)=STRAIN(5) + B(K)*EDIS(L) + B(L)*EDIS(K)                           00036700
   3    815 STRAIN(6)=STRAIN(6) + B(K)*EDIS(J) + B(J)*EDIS(K)                           00036800
   1  C                                                                                  00036900
   1  C     ISOTROPIC MATERIAL                                                           00037000
   1  C                                                                                  00037100
   7        IF (MODEL.GT.1) GO TO 821                                                   00037200
   7        DO 818 K=1,3                                                                00037300
   7        STRESS(K)=0.                                                                00037400
   7        DO 818 J=1,3                                                                00037500
   3    818 STRESS(K)=STRESS(K) + D(K,J)*STRAIN(J)                                      00037600
   7        STRESS(4)=D(4,4)*STRAIN(4)                                                  00037700
   7        STRESS(5)=D(5,5)*STRAIN(5)                                                  00037800
   7        STRESS(6)=D(6,6)*STRAIN(6)                                                  00037900
   7        GO TO 830                                                                   00038000
   1  C                                                                                  00038100
   3    821 CONTINUE                                                                    00038200
   1  C                                                                                  00038300
   3    830 WRITE(KIO6,2040) I,STRESS                                                   00038400
   1  C                                                                                  00038500
   1  C     CALCULATE AND PRINT STRESSES AT INTEGRATION POINTS                           00038600
   1  C                                                                                  00038700
   3    831 IPT=0                                                                       00038800
   7        DO 839 LX=1,NINT                                                            00038900
   7        E1=XG(LX,NINT)                                                              00039000
   7        DO 839 LY=1,NINT                                                            00039100
   7        E2=XG(LY,NINT)                                                              00039200
```

```
START COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8  MOD FLAGS
   7            DO 839 LZ=1,NINTZ                                                          00039300
   7            E3=XG(LZ,NINTZ)                                                            00039400
   7            IPT=IPT + 1                                                                00039500
   1  C                                                                                    00039600
   7            CALL DERIQ3  (N,XYZ(1,N),B,DET,E1,E2,E3,NOD9(1,N))                         00039700
   1  C                                                                                    00039800
   7            DO 832 J=1,6                                                               00039900
   3      832   STRAIN(J)=0.0                                                              00040000
   7            DO 833 K=3,ND,3                                                            00040100
   7            L=K - 2                                                                    00040200
   7            J=K - 1                                                                    00040300
   7            STRAIN(1)=STRAIN(1) + B(L)*EDIS(L)                                         00040400
   7            STRAIN(2)=STRAIN(2) + B(J)*EDIS(J)                                         00040500
   7            STRAIN(3)=STRAIN(3) + B(K)*EDIS(K)                                         00040600
   7            STRAIN(4)=STRAIN(4) + B(J)*EDIS(L) + B(L)*EDIS(J)                          00040700
   7            STRAIN(5)=STRAIN(5) + B(K)*EDIS(L) + B(L)*EDIS(K)                          00040800
   3      833   STRAIN(6)=STRAIN(6) + B(K)*EDIS(J) + B(J)*EDIS(K)                          00040900
   1  C                                                                                    00041000
   1  C     ISOTROPIC MATERIAL                                                             00041100
   1  C                                                                                    00041200
   7            IF (MODEL.GT.1) GO TO 835                                                  00041300
   7            DO 834 K=1,3                                                               00041400
   7            STRESS(K)=0.0                                                              00041500
   7            DO 834 J=1,3                                                               00041600
   3      834   STRESS(K)=STRESS(K) + D(K,J)*STRAIN(J)                                     00041700
   7            STRESS(4)=D(4,4)*STRAIN(4)                                                 00041800
   7            STRESS(5)=D(5,5)*STRAIN(5)                                                 00041900
   7            STRESS(6)=D(6,6)*STRAIN(6)                                                 00042000
   7            GO TO 836                                                                  00042100
   1  C                                                                                    00042200
   3      835   CONTINUE                                                                   00042300
   1  C                                                                                    00042400
   3      836   WRITE(KIO6,2040) IPT,STRESS                                                00042500
   1  C                                                                                    00042600
   3      839   CONTINUE                                                                   00042700
   1  C                                                                                    00042800
   3      840   CONTINUE                                                                   00042900
   7            RETURN                                                                     00043000
   1  C                                                                                    00043100
   1  C                                                                                    00043200
   1  C     STRESS CALCULATIONS FOR NONLINEAR MATERIAL MODELS                              00043300
   1  C                                                                                    00043400
   3      850   IPRNT=0                                                                    00043500
   7            DO 890 N=1,NUME                                                            00043600
   7            IF (IPST(N).EQ.0) GO TO 890                                                00043700
   7            IPRNT=IPRNT + 1                                                            00043800
   7            IF (IPRNT.NE.1) GO TO 852                                                  00043900
   7            WRITE(KIO6,2020) NG                                                        00044000
   3      852   MTYPE=MATP(N)                                                              00044100
```

```
START                                                                                                  MOD
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8  FLAGS
   7          IELD=IELTD(N)                                                            00044200
   7          IELX=IELTX(N)                                                            00044300
   7          IEL=IELD                                                                 00044400
   7          ND=3*IEL                                                                 00044500
   7          NND9=IELD - 8                                                            00044600
   1  C                                                                                00044700
   7          DO 860 I=1,ND                                                            00044800
   7          EDIS(I)=0.0                                                              00044900
   7          II=LM(I,N)                                                               00045000
   7          IF (II.LE.0) GO TO 860                                                   00045100
   7          EDIS(I)=X(II)                                                            00045200
   3    860 CONTINUE                                                                   00045300
   1  C                                                                                00045400
   1  C       CALCULATE AND PRINT STRESSES AT INTEGRATION POINTS                       00045500
   1  C                                                                                00045600
   7          IPT=0                                                                    00045700
   7          DO 885 LX=1,NINT                                                         00045800
   7          E1=XG(LX,NINT)                                                           00045900
   7          DO 885 LY=1,NINT                                                         00046000
   7          E2=XG(LY,NINT)                                                           00046100
   7          DO 885 LZ=1,NINTZ                                                        00046200
   7          E3=XG(LZ,NINTZ)                                                          00046300
   7          IPT=IPT + 1                                                              00046400
   1  C                                                                                00046500
   7          CALL DERIQ3  (N,XYZ(1,N),B,DET,E1,E2,E3,NOD9(1,N))                       00046600
   1  C                                                                                00046700
   7          DO 870 I=1,9                                                             00046800
   3    870 DISD(I)=0.0                                                                00046900
   7          DO 875 K=3,ND,3                                                          00047000
   7          I=K - 2                                                                  00047100
   7          J=K - 1                                                                  00047200
   7          DISD(1)=DISD(1) + B(I)*EDIS(I)                                           00047300
   7          DISD(2)=DISD(2) + B(J)*EDIS(J)                                           00047400
   7          DISD(3)=DISD(3) + B(K)*EDIS(K)                                           00047500
   7          DISD(4)=DISD(4) + B(J)*EDIS(I)                                           00047600
   7          DISD(5)=DISD(5) + B(K)*EDIS(I)                                           00047700
   7          DISD(6)=DISD(6) + B(I)*EDIS(J)                                           00047800
   7          DISD(7)=DISD(7) + B(K)*EDIS(J)                                           00047900
   7          DISD(8)=DISD(8) + B(I)*EDIS(K)                                           00048000
   3    875 DISD(9)=DISD(9) + B(J)*EDIS(K)                                             00048100
   1  C                                                                                00048200
   7          CALL STST3N  (DISD)                                                      00048300
   1  C                                                                                00048400
   3    885 CONTINUE                                                                   00048500
   3    890 CONTINUE                                                                   00048600
   7          RETURN                                                                   00048700
   1  C                                                                                00048800
   2   1000 FORMAT (I5,F10.0)                                                          00048900
   2   1001 FORMAT (8F10.0)                                                            00049000
```

```
START                                                                                                  MOD
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8  FLAGS

   2   1004 FORMAT (7I5)                                                                              00049100
   2   1005 FORMAT (13I5)                                                                             00049200
   2   1007 FORMAT (16I5)                                                                             00049300
   2   2004 FORMAT (/1H ,3I5,4X,I2,1X,3I6,5X,I4,7(4X,I4)/                                             00049400
   6        1        46X,I4,7(4X,I4) / 46X,I4,7(4X,I4))                                             00049500
   2   2005 FORMAT (///40H E L E M E N T   I N F O R M A T I O N      ,                               00049600
   6        1        ///30H     M  IELD IELX  IPS   MTYP                                            00049700
   6        2        11H   IST   KG,4X,8(A4,I1,3X)/45X,A4,I1,3X,7(A4,I2,2X)/                       00049800
   6        3        45X,8(A4,I2,2X))                                                               00049900
   2   2010 FORMAT (///12H *** ELEMENT,I5,46H+EXCEEDS MAXIMUM NUMBER OF NODES                       00050000
   6        1(NPAR(4)) ***)                                                                        00050100
   2   2020 FORMAT (1H1,46H S T R E S S   C A L C U L A T I O N S   F O R,3X,                       00050200
   6        1        25HE L E M E N T   G R O U P,3X,I2,3X,15H(3/D CONTINUUM)                        00050300
   6        2        ///)                                                                           00050400
   2   2030 FORMAT (8H ELEMENT,4X,6HOUTPUT,/ 2X,6HNUMBER,2X,8HLOCATION,7X,                          00050500
   6        1        8HSIGMA-X1,7X,8HSIGMA-X2,7X,8HSIGMA-X3,8X,7HTAU-X12,                           00050600
   6        2        8X,7HTAU-X13,8X,7HTAU-X23 / 1X)                                                00050700
   2   2035 FORMAT (I8)                                                                               00050800
   2   2040 FORMAT (13X,I5,6F15.4)                                                                    00050900
   2   2050 FORMAT (36H E L E M E N T   D E F I N I T I O N ///,                                      00051000
   6        1        14H ELEMENT TYPE ,13(2H .),16H( NPAR(1) ). . =,I5/,                            00051100
   6        2        25H     EQ.1, TRUSS ELEMENTS/,                                                 00051200
   6        3        25H     EQ.2, 2-DIM ELEMENTS/,                                                 00051300
   6        3        25H     EQ.3, 3-DIM ELEMENTS//,                                                00051400
   6        4        20H NUMBER OF ELEMENTS.,10(2H .),16H( NPAR(2) ). . =,I5//                      00051500
   6        5        40H TYPE OF NONLINEAR ANALYSIS  . . . . . .      ,                             00051600
   6        6        16H( NPAR(3) ). . =,I5/,                                                       00051700
   6        7        40H     EQ.1, MATERIAL NONLINEARITY ONLY      /,                               00051800
   6        8        40H     EQ.2, TOTAL LAGRANGIAN FORMULATION    /,                               00051900
   6        9        44H     EQ.3, UPDATED LAGRANGIAN FORMULATION            /)                     00052000
   2   2051 FORMAT (32H MAX NUMBER OF NODES  DESCRIBING/,                                             00052100
   6        9        20H     ANY ONE ELEMENT,10(2H .),16H( NPAR(7) ). . =,I5//,                     00052200
   6        1        40H INTEGRATION ORDER (R-S DIRECTION) FOR  /,                                  00052300
   6        2        40H     ELEMENT STIFFNESS GENERATION. . . .,                                   00052400
   6        3        16H( NPAR(10)). . =,I5//,                                                      00052500
   6        4        40H INTEGRATION ORDER (T DIRECTION) FOR    /,                                  00052600
   6        5        40H     ELEMENT STIFFNESS GENERATION. . . .,                                   00052700
   6        6        16H( NPAR(11)). . =,I5//,                                                      00052800
   6        7        40H NUMBER OF STRESS OUTPUT TABLES  . . . .,                                   00052900
   6        8        16H( NPAR(13)). . =,I5/                                                        00053000
   6        9        38H     EQ.0, PRINT AT INTEGRATION POINTS              ///)                   00053100
   2   2070 FORMAT (//40H S T R E S S   O U T P U T   T A B L E S    //                               00053200
   6        1        10H     TABLE,6X,1H1,6X,1H2,6X,1H3,6X,1H4,6X,1H5,6X,1H6,                       00053300
   6        2        6X,1H7,6X,1H8,6X,1H9,5X,2H10,5X,2H11,5X,2H12,5X,2H13,                          00053400
   6        3        5X,2H14,5X,2H15,5X,2H16//)                                                     00053500
   2   2060 FORMAT (I10,16I7)                                                                         00053600
   2   2090 FORMAT (44H *** STOP - INCORRECT NODAL DATA FOR EL. NO. ,I5)                              00053700
   2   3000 FORMAT (54H **STOP - NPAR(3) MUST EQUAL 1 IN THIS NONSAP VERSION*)                        00053800
   1   C                                                                                              00053900
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8  MOD FLAGS

   7        END                                                          00054000
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7        SUBROUTINE THREDM                                                      00000100
   7        IMPLICIT REAL*8(A-H,O-Z)                                               00000200
   7        COMMON/BBCCIO/KI01,KI02,KI03,KI04,KI05,KI06,KI07,KI08,KI09,            00000300
   6       1KI010,KI011,KI012,KI013,KI014,KI015,KI016,KI017,KI018,KI019,KI020,     00000400
   6       2KI021,KI022,KI023,KI024,KI025,KI026,KI027,KI028,KI029,KI030           00000500
   7        COMMON /SOL/ NUMNP,NEQ,NWK,NWM,NWC,NUMEST,MIDEST,MAXEST,NSTE,MA        00000600
   7        COMMON /DIM/ N1,N2,N3,N4,N5,N6,N7,N8,N9,N10,N11,N12,N13,N14,N15        00000700
   7        COMMON /DIMEL/ N101,N102,N103,N104,N105,N106,N107,N108,N109,N110,      00000800
   6       1               N111,N112,N113,N114,N120,N121,N122,N123,N124,N125      00000900
   7        COMMON /EL/ IND,ICOUNT,NPAR(20),NUMEG,NEGL,NEGNL,IMASS,IDAMP,ISTAT    00001000
   6       1           ,NDOF,KLIN,IEIG,IMASSN,IDAMPN                              00001100
   7        COMMON /THREED/ DE,IELD,IELX,IEL,NPT,IDW,NND9                         00001200
   7        COMMON A(1)                                                           00001300
   1  C                                                                           00001400
   7        DIMENSION NMCON(10),IDWAS(10)                                         00001500
   1  C                                                                           00001600
   7        EQUIVALENCE (NPAR(15),MODEL),(NPAR(10),NINT),(NPAR(11),NINTZ)         00001700
   6       1           ,(NPAR(7),MXNODS),(NPAR(2),NUME),(NPAR(16),NUMMAT)        00001800
   6       2           ,(NPAR(13),NTABLE)                                        00001900
   1  C                                                                           00002000
   7        DATA NMCON / 2,7,24,7*0 /, IDWAS / 0,0,13,7*0 /                       00002100
   1  C                                                                           00002200
   7        NDM=3*MXNODS                                                          00002300
   7        NDM2=(NDM*NDM)/2 + NDM/2 + 1                                          00002400
   7        ND9DIM=MXNODS - 8                                                     00002500
   7        NCON=NMCON(MODEL)                                                     00002600
   7        IDW =IDWAS(MODEL)                                                     00002700
   7        IF (NCON.NE.0) GO TO 20                                               00002800
   7        NCON=NPAR(17)                                                         00002900
   7        IDW=NPAR(18)                                                          00003000
   4     20 IDWA=IDW*(NINT*NINT*NINTZ)                                            00003100
   7        NPAR(17)=NCON                                                         00003200
   7        NPAR(18)=IDW                                                          00003300
   7        IF (MODEL.NE.2) GO TO 21                                              00003400
   7        WRITE(KI06,3000)                                                      00003500
   7        STOP                                                                  00003600
   1  C                                                                           00003700
   1  C                                                                           00003800
   1  C     STORAGE ALLOCATION                                                    00003900
   1  C                                                                           00004000
   4     21 N101=1                                                               00004100
   7        N102=N101 + NDM*NUME                                                  00004200
   7        N103=N102 + NDM*NUME                                                  00004300
   1  C                                                                           00004400
   7        N104=N103 + NUME                                                      00004500
   7        N105=N104 + NUME                                                      00004600
   7        N107=N105 + NUME                                                      00004700
   7        N108=N107 + NUME                                                      00004800
   7        N109=N108 + ND9DIM*NUME                                               00004900
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7        N110=N109 + NUME                                                      00005000
   1  C                                                                           00005100
   7        N111=N110 + NUMMAT                                                    00005200
   7        N112=N111 + NCON*NUMMAT                                               00005300
   7        N113=N112 + IDWA*NUME                                                 00005400
   7        NLAST=N113 + 16*NTABLE - 1                                            00005500
   1  C                                                                           00005600
   7        N120=NLAST + 1                                                        00005700
   7        N121=N120 + NDM2                                                      00005800
   7        N122=N121 + NDM                                                       00005900
   7        N123=N122 + NDM                                                       00006000
   7        N124=N123 + NDM                                                       00006100
   7        N125=N124 + NDM - 1                                                   00006200
   1  C                                                                           00006300
   7        IF (NLAST.LE.NUMEST) GO TO 100                                        00006400
   7        WRITE(KIO6,2000) NLAST                                                00006500
   7        STOP                                                                  00006600
   1  C                                                                           00006700
   3    100 MIDEST=NLAST                                                        00006800
   1  C                                                                           00006900
   7        IF (IND.GT.3) GO TO 110                                               00007000
   7        M2=N2                                                                 00007100
   7        M3=N3                                                                 00007200
   7        M4=N4                                                                 00007300
   7        M5=N5                                                                 00007400
   7        GO TO 120                                                             00007500
   3    110 M2=N2                                                                 00007600
   7        M3=N7                                                                 00007700
   7        M4=N8                                                                 00007800
   7        M5=N3                                                                 00007900
   7        IF (ICOUNT.LT.3) GO TO 120                                            00008000
   7        M2=N6                                                                 00008100
   7        M5=N5                                                                 00008200
   1  C                                                                           00008300
   3    120 CALL THDFE (A(N1),A(M2),A(M3),A(M4),A(M5),A(N101),A(N102),           00008400
   6       1            A(N103),A(N104),A(N105),A(N107),A(N108),                 00008500
   6       2            A(N109),A(N110),A(N111),A(N112),A(N120),A(N121),         00008600
   6       3            A(N122),A(N123),A(N124),A(N113),NTABLE,NCON,IDWA,        00008700
   6       4            NDM,NDM2,NDOF,ND9DIM)                                    00008800
   1  C                                                                           00008900
   7        RETURN                                                                00009000
   1  C                                                                           00009100
   2     2000 FORMAT (///40H *** ERROR NUMEST MUST BE INCREASED TO       ,I6)     00009200
   2     3000 FORMAT (///40H ***STOP*** MODEL 2 NOT PERMITTED         )           00009300
   1  C                                                                           00009400
   7        END                                                                   00009500
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7        SUBROUTINE TODMFE                                                      00000100
   7        IMPLICIT REAL*8(A-H,O-Z)                                               00000200
   7        COMMON/BBCCIO/KIO1,KIO2,KIO3,KIO4,KIO5,KIO6,KIO7,KIO8,KIO9,            00000300
   6       1KIO10,KIO11,KIO12,KIO13,KIO14,KIO15,KIO16,KIO17,KIO18,KIO19,KIO20,     00000400
   6       2KIO21,KIO22,KIO23,KIO24,KIO25,KIO26,KIO27,KIO28,KIO29,KIO30           00000500
   7        COMMON /SOL/ NUMNP,NEQ,NWK,NWM,NWC,NUMEST,MIDEST,MAXEST,NSTE,MA        00000600
   7        COMMON /DIM/ N1,N2,N3,N4,N5,N6,N7,N8,N9,N10,N11,N12,N13,N14,N15        00000700
   7        COMMON /DIMEL/ N101,N102,N103,N104,N105,N106,N107,N108,N109,N110,      00000800
   6       1               N111,N112,N113,N114,N120,N121,N122,N123,N124,N125       00000900
   7        COMMON /EL/ IND,ICOUNT,NPAR(20),NUMEG,NEGL,NEGNL,IMASS,IDAMP,ISTAT    00001000
   6       1           ,NDOF,KLIN,IEIG,IMASSN,IDAMPN                              00001100
   7        COMMON /TODIM/ BET,THIC,DE,IEL,NND5                                    00001200
   7        COMMON A(1)                                                            00001300
   7        DIMENSION NMCON(20),IDWAS(20)                                          00001400
   1  C                                                                            00001500
   7        EQUIVALENCE (NPAR(2),NUME),(NPAR(16),NUMMAT),(NPAR(17),NCON)           00001600
   6       1           ,(NPAR(18),IDW),(NPAR(7),MXNODS)                           00001700
   6       2           ,(NPAR(15),MODEL),(NPAR(10),NINT)                          00001800
   6       3           ,(NPAR(13),NTABLE)                                         00001900
   1  C                                                                            00002000
   1  C                                                                            00002100
   7        DATA NMCON / 2, 7, 7, 24, 24, 4, 4, 2, 12*0 /,                         00002200
   6       1     IDWAS / 0, 0,10,12,12,10,11, 0, 12*0 /                           00002300
   1  C                                                                            00002400
   1  C                                                                            00002500
   7        NDM=2*MXNODS                                                           00002600
   7        ND5DIM=MXNODS - 4                                                      00002700
   7        NCONT=NMCON(MODEL)                                                     00002800
   7        IDWT=IDWAS(MODEL)                                                      00002900
   7        IF (NCONT.NE.0) GO TO 20                                               00003000
   7        NCONT=NCON                                                             00003100
   7        IDWT=IDW                                                               00003200
   4    20  IDWA=IDWT*NINT*NINT                                                    00003300
   7        NCON=NCONT                                                             00003400
   7        IDW=IDWT                                                               00003500
   1  C                                                                            00003600
   1  C     STORAGE ALLOCATION                                                     00003700
   1  C                                                                            00003800
   7        N101=1                                                                 00003900
   7        N102=N101 + NDM*NUME                                                   00004000
   7        N103=N102 + NDM*NUME                                                   00004100
   1  C                                                                            00004200
   7        N104=N103 + NUME                                                       00004300
   7        N105=N104 + NUME                                                       00004400
   7        N106=N105 + NUME                                                       00004500
   7        N107=N106 + NUME                                                       00004600
   7        N108=N107 + NUME                                                       00004700
   1  C                                                                            00004800
   7        N109=N108 + NUMMAT                                                     00004900
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7        N110=N109 + NCON*NUMMAT                                                  00005000
   7        N111=N110 + IDWA*NUME                                                    00005100
   7        N112=N111 + ND5DIM*NUME                                                  00005200
   7        NLAST=N112 + 9*NTABLE - 1                                                00005300
   1  C                                                                              00005400
   7        IF (NLAST.LE.NUMEST) GO TO 100                                           00005500
   7        WRITE(KIO6,2000) NLAST                                                   00005600
   7        STOP                                                                     00005700
   1  C                                                                              00005800
   3    100 MIDEST=NLAST                                                           00005900
   1  C                                                                              00006000
   7        IF (IND.GT.3) GO TO 110                                                  00006100
   7        M3=N3                                                                    00006200
   7        M4=N4                                                                    00006300
   7        M5=N5                                                                    00006400
   7        GO TO 120                                                                00006500
   3    110 M3=N2                                                                  00006600
   7        M4=N7                                                                    00006700
   7        M5=N3                                                                    00006800
   7        IF (ICOUNT.LT.3) GO TO 120                                               00006900
   7        M3=N6                                                                    00007000
   7        M5=N5                                                                    00007100
   1  C                                                                              00007200
   3    120 CALL TDFE (A(N1),A(M3),A(M4),A(M5),A(N101),A(N102),A(N103),             00007300
   6       1           A(N104),A(N105),A(N106),A(N107),A(N108),A(N109),             00007400
   6       2           A(N110),A(N111),A(N112),NTABLE,NCON,IDWA,NDM,ND5DIM,         00007500
   6       3           NDOF)                                                        00007600
   1  C                                                                              00007700
   7        RETURN                                                                   00007800
   1  C                                                                              00007900
   2   2000 FORMAT (///40H *** ERROR NUMEST MUST BE INCREASED TO , I6)              00008000
   1  C                                                                              00008100
   7        END                                                                      00008200
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7        SUBROUTINE TRUSS                                                      00000100
   7        IMPLICIT REAL*8(A-H,O-Z)                                              00000200
   1  C                                                                           00000300
   7        COMMON/BBCCIO/KIO1,KIO2,KIO3,KIO4,KIO5,KIO6,KIO7,KIO8,KIO9,           00000400
   6       1KIO10,KIO11,KIO12,KIO13,KIO14,KIO15,KIO16,KIO17,KIO18,KIO19,KIO20,   00000500
   6       2KIO21,KIO22,KIO23,KIO24,KIO25,KIO26,KIO27,KIO28,KIO29,KIO30          00000600
   7        COMMON /DIM/ N1,N2,N3,N4,N5,N6,N7,N8,N9,N10,N11,N12,N13,N14,N15       00000700
   7        COMMON /SOL/ NUMNP,NEQ,NWK,NWM,NWC,NUMEST,MIDEST,MAXEST,NSTE,MA       00000800
   7        COMMON /EL/ IND,ICOUNT,NPAR(20),NUMEG,NEGL,NEGNL,IMASS,IDAMP,ISTAT   00000900
   6       1           ,NDOF,KLIN,IEIG,IMASSN,IDAMPN                             00001000
   7        COMMON /VAR/ NG,KPRI,MODEX,KSTEP,ITE,ITEMAX,IREF,IEQREF,INOCMD       00001100
   7        COMMON A(1)                                                           00001200
   1  C                                                                           00001300
   7        EQUIVALENCE (NPAR(2),NUME),(NPAR(16),NUMMAT),(NPAR(17),NCON)         00001400
   1  C                                                                           00001500
   7        N101=1                                                                00001600
   7        N102=N101 + NUMMAT                                                    00001700
   7        N103=N102 + NUMMAT                                                    00001800
   7        N104=N103 + NUMMAT                                                    00001900
   7        N105=N104 + NUMMAT                                                    00002000
   7        N106=N105 + 6*NUME                                                    00002100
   7        N107=N106 + 6*NUME                                                    00002200
   7        N108=N107 + NUME                                                      00002300
   7        N109=N108 + NUME                                                      00002400
   7        N110=N109 + NUME                                                      00002500
   7        NLAST=N110 + NCON*NUMMAT - 1                                          00002600
   1  C                                                                           00002700
   7        IF (NLAST.LE.NUMEST) GO TO 100                                        00002800
   7        WRITE(KIO6,2000) NLAST                                                00002900
   7        STOP                                                                  00003000
   1  C                                                                           00003100
   3    100 MIDEST=NLAST                                                        00003200
   1  C                                                                           00003300
   7        IF (IND.GT.3) GO TO 110                                               00003400
   7        M2=N2                                                                 00003500
   7        M3=N3                                                                 00003600
   7        M4=N4                                                                 00003700
   7        M5=N5                                                                 00003800
   7        GO TO 120                                                             00003900
   3    110 M2=N2                                                               00004000
   7        M3=N7                                                                 00004100
   7        M4=N8                                                                 00004200
   7        M5=N3                                                                 00004300
   7        IF (ICOUNT.LT.3) GO TO 120                                            00004400
   7        M2=N6                                                                 00004500
   7        M5=N5                                                                 00004600
   1  C                                                                           00004700
   3    120 CALL RUSS(A(N1),A(M2),A(M3),A(M4),A(M5),A(N101),A(N102),A(N103),     00004800
   6       1          A(N104),A(N105),A(N106),A(N107),A(N108),A(N109),           00004900
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   6        2          A(N110),NCON,NDOF)                                    00005000
   1  C                                                                      00005100
   7        RETURN                                                          00005200
   1  C                                                                      00005300
   2   2000 FORMAT (///40H *** ERROR  NUMEST MUST BE INCREASED TO , I6)      00005400
   1  C                                                                      00005500
   7        END                                                             00005600
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7        SUBROUTINE TTIME(X)                                          00000100
   7        IMPLICIT REAL*8(A-H,O-Z)                                     00000200
   7        X = 0.                                                       00000300
   7        RETURN                                                       00000400
   7        END                                                          00000500
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7        SUBROUTINE VTMOD (PROP,SIG,EPS,PMAX,LD)                                00000100
   7        IMPLICIT REAL*8(A-H,O-Z)                                               00000200
   1  C                                                                            00000300
   7        COMMON/BBCCIO/KIO1,KIO2,KIO3,KIO4,KIO5,KIO6,KIO7,KIO8,KIO9,            00000400
   6       1KIO10,KIO11,KIO12,KIO13,KIO14,KIO15,KIO16,KIO17,KIO18,KIO19,KIO20,     00000500
   6       2KIO21,KIO22,KIO23,KIO24,KIO25,KIO26,KIO27,KIO28,KIO29,KIO30           00000600
   7        COMMON /VAR/ NG,KPRI,MODEX,KSTEP,ITE,ITEMAX,IREF,IEQREF,INOCMD         00000700
   7        COMMON /EL/ IND,ICOUNT,NPAR(20),NUMEG,NEGL,NEGNL,IMASS,IDAMP,ISTAT     00000800
   6       1           ,NDOF,KLIN,IEIG,IMASSN,IDAMPN                               00000900
   7        COMMON /MATMOD/ STRESS(4),STRAIN(4),C(4,4),IPT,NEL                     00001000
   7        COMMON /DISDER/ DISD(5)                                                00001100
   1  C                                                                            00001200
   7        REAL*8 JEY,JEY2,KA,KUNLO,KLOAD,KEY(2)                                  00001300
   7        DIMENSION PROP(1),SIG(1),EPS(1),GEE(2),ILIN(2,2),ILOD(2,4)             00001400
   7        EQUIVALENCE (NPAR(5),ITYP2D),(KEY(1),KUNLO),(KEY(2),KLOAD),            00001500
   6       1            (GEE(1),GUNLO),(GEE(2),GLOAD),(NPAR(3),INDNL)             00001600
   1  C                                                                            00001700
   7        DATA NGLAST /1000/, ILIN /1,3,2,4/, ILOD /1,1,1,2,2,1,2,2/             00001800
   1  C                                                                            00001900
   7        IKAS=ILOD(1,LD)                                                        00002000
   7        IGAS=ILOD(2,LD)                                                        00002100
   1  C                                                                            00002200
   1  C                                                                            00002300
   1  C        1. CALCULATE STRESS AND STRAIN DEVIATORS OF THE PREVIOUS STEP       00002400
   1  C                                                                            00002500
   7        TMM=(SIG(1)+SIG(2)+SIG(4))/3.                                          00002600
   7        T11=SIG(1) - TMM                                                       00002700
   7        T22=SIG(2) - TMM                                                       00002800
   7        T33=SIG(4) - TMM                                                       00002900
   7        T12=SIG(3)                                                             00003000
   7        JEY=.5 * (T11*T11 + T22*T22 + T33*T33) + T12*T12                       00003100
   1  C                                                                            00003200
   7        EMM=(EPS(1)+EPS(2)+EPS(4))/3.                                          00003300
   7        E11=EPS(1) - EMM                                                       00003400
   7        E22=EPS(2) - EMM                                                       00003500
   7        E33=EPS(4) - EMM                                                       00003600
   7        E12=EPS(3)                                                             00003700
   1  C                                                                            00003800
   1  C                                                                            00003900
   1  C        2. CALCULATE PARAMETERS BASED ON STRESSES AND STRAINS OF            00004000
   1  C           THE PREVIOUS TIME STEP                                           00004100
   1  C                                                                            00004200
   7        KUNLO=PROP(1)                                                          00004300
   7        KLOAD=PROP(2)-PROP(3)*EMM+PROP(4)*EMM*EMM                              00004400
   7        GUNLO=PROP(5)-PROP(6)*TMM                                              00004500
   7        GLOAD=GUNLO+PROP(7)*DSQRT(JEY)                                         00004600
   1  C                                                                            00004700
   1  C                                                                            00004800
   1  C        3. USING PARAMETERS CALCULATED IN (2.), FIND CURRENT STRESSES       00004900
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   1  C            BASED ON GIVEN STRAINS AND ASSUMPTIONS OF LOADING           00005000
   1  C                                                                        00005100
   7         KA=KEY(IKAS)                                                     00005200
   7         G =GEE(IGAS)                                                     00005300
   7         IF (ITYP2D.NE.2) GO TO 120                                       00005400
   1  C                                                                        00005500
   1  C      FOR PLANE STRESS, CALCULATE STRAIN(4)                             00005600
   1  C                                                                        00005700
   7         PV=(3.*KA-2.*G)/(6.*KA+2.*G)                                     00005800
   7         STRAIN(4)=EPS(4)+(PV/(PV-1.))*(STRAIN(1)+STRAIN(2)-EPS(1)-EPS(2)) 00005900
   1  C                                                                        00006000
   3    120  EPSMM=(STRAIN(1)+STRAIN(2)+STRAIN(4))/3.                         00006100
   7         EPS11=STRAIN(1)-EPSMM                                            00006200
   7         EPS22=STRAIN(2)-EPSMM                                            00006300
   7         EPS33=STRAIN(4)-EPSMM                                            00006400
   7         EPS12=STRAIN(3)                                                  00006500
   1  C                                                                        00006600
   7         PEE=3.*KA*(EPSMM-EMM)+TMM                                        00006700
   1  C                                                                        00006800
   7         S11=2.*G*(EPS11-E11)+T11                                         00006900
   7         S22=2.*G*(EPS22-E22)+T22                                         00007000
   7         S33=2.*G*(EPS33-E33)+T33                                         00007100
   7         S12=   G*(EPS12-E12)+T12                                         00007200
   1  C                                                                        00007300
   1  C                                                                        00007400
   1  C         4. CALCULATE STRESSES                                          00007500
   1  C                                                                        00007600
   7         STRESS(1)=S11+PEE                                                00007700
   7         STRESS(2)=S22+PEE                                                00007800
   7         STRESS(4)=S33+PEE                                                00007900
   7         STRESS(3)=S12                                                    00008000
   1  C                                                                        00008100
   7         IF (KPRI.EQ.0) GO TO 700                                         00008200
   1  C                                                                        00008300
   1  C                                                                        00008400
   1  C         5. UPDATE STRESSES AND STRAINS                                 00008500
   1  C                                                                        00008600
   7         SIG(1)=STRESS(1)                                                 00008700
   7         SIG(2)=STRESS(2)                                                 00008800
   7         SIG(3)=STRESS(3)                                                 00008900
   7         SIG(4)=STRESS(4)                                                 00009000
   1  C                                                                        00009100
   7         EPS(1)=STRAIN(1)                                                 00009200
   7         EPS(2)=STRAIN(2)                                                 00009300
   7         EPS(3)=STRAIN(3)                                                 00009400
   7         EPS(4)=STRAIN(4)                                                 00009500
   1  C                                                                        00009600
   1  C                                                                        00009700
   1  C         6. REDEFINE AND UPDATE LOADING CONDITION                       00009800
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   1  C                                                                    00009900
   7        JEY2=.5*(S11*S11+S22*S22+S33*S33) + S12*S12                    00010000
   1  C                                                                    00010100
   7        IKAC=1                                                         00010200
   7        IGAC=1                                                         00010300
   7        IF (PEE.LT.PMAX) IKAC=2                                        00010400
   7        IF (PEE.EQ.TMM) IKAC=IKAS                                      00010500
   7        IF (JEY2.GT.JEY) IGAC=2                                        00010600
   1  C                                                                    00010700
   7        IF (IKAC.EQ.2) PMAX=PEE                                        00010800
   7        LD=ILIN(IKAC,IGAC)                                             00010900
   1  C                                                                    00011000
   7        IF (ICOUNT.EQ.3) RETURN                                        00011100
   1  C                                                                    00011200
   1  C                                                                    00011300
   1  C        7. FORM THE STRESS-STRAIN RELATIONSHIP                      00011400
   1  C                                                                    00011500
   7        KLOAD=PROP(2)-PROP(3)*EPSMM+PROP(4)*EPSMM*EPSMM                00011600
   7        IF (KUNLO.GE.KLOAD) GO TO 150                                  00011700
   7        KLOAD=KUNLO                                                    00011800
   1  C                                                                    00011900
   3    150 GUNLO=PROP(5)-PROP(6)*PEE                                      00012000
   7        GLOAD=GUNLO+PROP(7)*DSQRT(JEY2)                                00012100
   7        IF (GLOAD.GT.0.) GO TO 160                                     00012200
   7        WRITE(KIO6,2200) NEL,KSTEP,GLOAD                               00012300
   7        STOP                                                           00012400
   1  C                                                                    00012500
   3    160 KA=KEY(IKAC)                                                   00012600
   7        G =GEE(IGAC)                                                   00012700
   1  C                                                                    00012800
   7        YM=(9.*KA*G)/(3.*KA+G)                                         00012900
   7        PV=(3.*KA-2.*G)/(6.*KA+2.*G)                                   00013000
   1  C                                                                    00013100
   7        IF (PV.GE.0. .AND. PV.LT.0.5) GO TO 200                        00013200
   1  C                                                                    00013300
   7        WRITE(KIO6,2000) NEL,PV                                        00013400
   7        STOP                                                           00013500
   1  C                                                                    00013600
   3    200 C1=YM/(1.+PV)                                                  00013700
   7        B1=C1*PV/(1.-2.*PV)                                            00013800
   7        A1=B1+C1                                                       00013900
   7        C(1,1)=A1                                                      00014000
   7        C(1,2)=B1                                                      00014100
   7        C(1,3)=0.                                                      00014200
   7        C(2,1)=B1                                                      00014300
   7        C(2,2)=A1                                                      00014400
   7        C(2,3)=0.                                                      00014500
   7        C(3,1)=0.                                                      00014600
   7        C(3,2)=0.                                                      00014700
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7          C(3,3)=C1/2.                                                    00014800
   1  C                                                                       00014900
   7          IF (ITYP2D.EQ.1) RETURN                                         00015000
   1  C                                                                       00015100
   7          C(1,4)=B1                                                       00015200
   7          C(2,4)=B1                                                       00015300
   7          C(3,4)=0.                                                       00015400
   7          C(4,1)=B1                                                       00015500
   7          C(4,2)=B1                                                       00015600
   7          C(4,3)=0.                                                       00015700
   7          C(4,4)=A1                                                       00015800
   1  C                                                                       00015900
   7          IF (ITYP2D.LT.2) RETURN                                         00016000
   1  C                                                                       00016100
   1  C       FOR PLANE STRESS ANALYSIS CONDENSE STRESS-STRAIN MATRIX         00016200
   1  C                                                                       00016300
   7          DO 95 I=1,3                                                     00016400
   7          A=C(I,4)/C(4,4)                                                 00016500
   7          DO 95 J=I,3                                                     00016600
   7          C(I,J)=C(I,J) - C(4,J)*A                                        00016700
   4       95 C(J,I)=C(I,J)                                                   00016800
   7          RETURN                                                          00016900
   1  C                                                                       00017000
   1  C                                                                       00017100
   1  C                                                                       00017200
   1  C       P R I N T I N G   O F   S T R E S S E S                         00017300
   1  C                                                                       00017400
   3     700 IF (INDNL.NE.2) GO TO 800                                        00017500
   1  C                                                                       00017600
   1  C       IN TOTAL LAGRANGIAN FORMULATION,                                00017700
   1  C       CAUCHY STRESSES ARE CALCULATED AND PRINTED                      00017800
   1  C                                                                       00017900
   7          CALL CAUCHY                                                     00018000
   1  C                                                                       00018100
   3     800 IF (NG.NE.NGLAST) GO TO 805                                      00018200
   7          IF (NEL.GT.NELAST) GO TO 825                                    00018300
   7          IF (IPT-1) 830,820,830                                          00018400
   1  C                                                                       00018500
   3     805 NGLAST=NG                                                        00018600
   3     820 IF (ITYP2D) 810,815,810                                          00018700
   3     810 WRITE(KIO6,2010)                                                 00018800
   7          GO TO 825                                                       00018900
   3     815 WRITE(KIO6,2012)                                                 00019000
   1  C                                                                       00019100
   3     825 NELAST=NEL                                                       00019200
   7          WRITE(KIO6,2020) NEL                                            00019300
   3     830 CALL MAXMIN (STRESS,P1,P2,AG)                                    00019400
   7          IF (ITYP2D) 840,845,840                                         00019500
   1  C                                                                       00019600
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   3    840 WRITE(KIO6,2030)                                                    00019700
   6       *                IPT,(STRESS(I),I=1,3),P1,P2,AG                      00019800
   7         RETURN                                                            00019900
   1  C                                                                        00020000
   3    845 WRITE(KIO6,2040)                                                    00020100
   6       *                IPT,STRESS(4),(STRESS(I),I=1,3),P1,P2,AG            00020200
   7         RETURN                                                            00020300
   1  C                                                                        00020400
   2   2000 FORMAT (1H1,52H***ERROR   INADMISSABLE POISSONS RATIO FOR ELEMENT  00020500
   6       1         I6//77H STOP IN CALCULATION OF MATERIAL LAW USING THE VARI00020600
   6       2ABLE TANGENT MODULI MODEL    ,E11.4)                             00020700
   2   2010 FORMAT ( 19H ELEMENT     OUTPUT/                                   00020800
   6       1          109H  NUMBER   LOCATION        STRESS-YY      STRESS-ZZ  00020900
   6       1    STRESS-YZ          MAX STRESS     MIN STRESS    ANGLE/)       00021000
   2   2012 FORMAT ( 19H ELEMENT     OUTPUT/                                   00021100
   6       1          124H  NUMBER   LOCATION        STRESS-XX      STRESS-YY  00021200
   6       1    STRESS-ZZ      STRESS-YZ          MAX STRESS     MIN STRESS   00021300
   6       2ANGLE/)                                                          00021400
   2   2020 FORMAT (I5)                                                        00021500
   2   2030 FORMAT (15X,I2,4X,3(1X,E14.6),4X,2(1X,E14.6),3X,F6.2)              00021600
   2   2040 FORMAT (15X,I2,4X,4(1X,E14.6),4X,2(1X,E14.6),3X,F6.2)              00021700
   2   2200 FORMAT (1H1, 38H INADMISSIBLE SHEAR MODULUS IN LOADING /           00021800
   6       1         12H ELEMENT   =,I5/ 12H TIME STEP =,I5/ 8H GLOAD =,      00021900
   6       2         E14.6//38H STOP IN VARIABLE TANGENT MODULI MODEL  )      00022000
   1  C                                                                        00022100
   7         END                                                              00022200
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7        SUBROUTINE WRITE (DISP,VEL,ACC,ID,NEQ,NDOF,NUMNP)                    00000100
   7        IMPLICIT REAL*8(A-H,O-Z)                                             00000200
   1  C .                                                                      . 00000300
   1  C .   P R O G R A M                                                      . 00000400
   1  C .      . TO PRINT DISPLACEMENTS AND (IF ISTAT.NE.0)                    . 00000500
   1  C .        VELOCITIES AND ACCELERATIONS                                  . 00000600
   1  C .                                                                      . 00000700
   1  C .          IND.EQ.0,   PRINT DISPL/VEL/ACC AT ALL NODES                . 00000800
   1  C .          IND.NE.0,   PRINT DISPL/VEL/ACC ONLY AT NODES               . 00000900
   1  C .                      CONTAINED IN PRINTOUT BLOCKS                    . 00001000
   1  C .                                                                      . 00001100
   1  C . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 00001200
   7        COMMON/BBCCIO/KIO1,KIO2,KIO3,KIO4,KIO5,KIO6,KIO7,KIO8,KIO9,        00001300
   6       1KIO10,KIO11,KIO12,KIO13,KIO14,KIO15,KIO16,KIO17,KIO18,KIO19,KIO20,00001400
   6       2KIO21,KIO22,KIO23,KIO24,KIO25,KIO26,KIO27,KIO28,KIO29,KIO30        00001500
   1  C                                                                        00001600
   7        COMMON /EL/ IND,ICOUNT,NPAR(20),NUMEG,NEGL,NEGNL,IMASS,IDAMP,ISTAT00001700
   6       1            ,NDOFDM,KLIN,IEIG,IMASSN,IDAMPN                      00001800
   7        COMMON /PRCON/ NPB,IDC,IVC,IAC,IPC,IPNODE(2,8)                      00001900
   7        DIMENSION DISP(NEQ),VEL(NEQ),ACC(NEQ),ID(NDOF,1)                    00002000
   7        DIMENSION D(6)                                                      00002100
   1  C                                                                        00002200
   1  C     READ ID ARRAY INTO CORE                                            00002300
   1  C                                                                        00002400
   7        REWIND KIO18                                                        00002500
   7        NDBLK=NUMNP                                                         00002600
   7        READ(KIO18)                                                         00002700
   6       *         ((ID(I,J),I=1,NDOF),J=1,NUMNP)                           00002800
   1  C                                                                        00002900
   1  C     PRINT DISPLACEMENTS                                                00003000
   1  C                                                                        00003100
   7        IC=4                                                                00003200
   7        IF (IND.EQ.0) GO TO 10                                              00003300
   7        IF (IDC.EQ.0) GO TO 180                                             00003400
   4     10 WRITE(KIO6,2000)                                                    00003500
   7        IC=IC + 5                                                           00003600
   7        DO 150 IB=1,NPB                                                     00003700
   7        NODE1=IPNODE(1,IB)                                                  00003800
   7        IF (NODE1.EQ.0) GO TO 150                                           00003900
   7        NODE2=IPNODE(2,IB)                                                  00004000
   7        IF (IND.EQ.0) NODE1=1                                               00004100
   7        IF (IND.EQ.0) NODE2=NUMNP                                           00004200
   1  C                                                                        00004300
   7        DO 100 II=NODE1,NODE2                                               00004400
   7        IC=IC + 1                                                           00004500
   7        IF (IC.LT.56) GO TO 105                                             00004600
   7        WRITE(KIO6,2045)                                                    00004700
   7        IC=4                                                                00004800
   3   105  DO 110 I=1,6                                                       00004900
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   3    110 D(I)=0.                                                        00005000
   7        DO 120 I=1,NDOF                                                00005100
   7        KK=ID(I,II)                                                    00005200
   7        IL=I                                                           00005300
   7        IF (NDOF.EQ.2) IL=I + 1                                        00005400
   3    120 IF (KK.NE.0) D(IL)=DISP(KK)                                    00005500
   3    100 WRITE(KIO6,2010) II,D                                          00005600
   1  C                                                                    00005700
   7        IF (IND.EQ.0) GO TO 180                                        00005800
   7        IF (IC.GE.55) GO TO 150                                        00005900
   7        IC=IC+1                                                        00006000
   7        WRITE(KIO6,2050)                                               00006100
   3    150 CONTINUE                                                       00006200
   3    180 IF (ISTAT.EQ.0) RETURN                                         00006300
   1  C                                                                    00006400
   1  C     PRINT VELOCITIES                                               00006500
   1  C                                                                    00006600
   7        IF (IND.EQ.0) GO TO 201                                        00006700
   7        IF (IVC.EQ.0) GO TO 280                                        00006800
   3    201 IC=IC + 5 + IDC                                                00006900
   7        IF (IDC.NE.0) WRITE (6,2050)                                   00007000
   7        IF (IC.GE.54) GO TO 205                                        00007100
   7        WRITE(KIO6,2020)                                               00007200
   7        GO TO 206                                                      00007300
   3    205 WRITE(KIO6,2022)                                               00007400
   7        IC=4                                                           00007500
   3    206 DO 250 IB=1,NPB                                                00007600
   7        NODE1=IPNODE(1,IB)                                             00007700
   7        IF (NODE1.EQ.0) GO TO 250                                      00007800
   7        NODE2=IPNODE(2,IB)                                             00007900
   7        IF (IND.EQ.0) NODE1=1                                          00008000
   7        IF (IND.EQ.0) NODE2=NUMNP                                      00008100
   1  C                                                                    00008200
   7        DO 200 II=NODE1,NODE2                                          00008300
   7        IC=IC + 1                                                      00008400
   7        IF (IC.LT.56) GO TO 207                                        00008500
   7        WRITE(KIO6,2022)                                               00008600
   7        IC=4                                                           00008700
   3    207 DO 210 I=1,6                                                   00008800
   3    210 D(I)=0.                                                        00008900
   7        DO 220 I=1,NDOF                                                00009000
   7        KK=ID(I,II)                                                    00009100
   7        IL=I                                                           00009200
   7        IF (NDOF.EQ.2) IL=I + 1                                        00009300
   3    220 IF (KK.NE.0) D(IL)=VEL(KK)                                     00009400
   3    200 WRITE(KIO6,2010) II,D                                          00009500
   1  C                                                                    00009600
   7        IF (IND.EQ.0) GO TO 280                                        00009700
   7        IF (IC.GE.55) GO TO 250                                        00009800
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7         IC=IC+1                                                        00009900
   7        WRITE (6,2050)                                                  00010000
   3    250 CONTINUE                                                        00010100
   1  C                                                                     00010200
   1  C     PRINT ACCELERATIONS                                             00010300
   1  C                                                                     00010400
   3    280 IF (IND.EQ.0) GO TO 290                                         00010500
   7        IF (IAC.EQ.0) RETURN                                            00010600
   7        IF (IDC.EQ.0 .AND. IVC.EQ.0) GO TO 305                          00010700
   3    290 IC=IC + 6                                                       00010800
   7        IF (IC.GE.54) GO TO 303                                         00010900
   7        WRITE(KIO6,2050)                                                00011000
   7        WRITE(KIO6,2030)                                                00011100
   7        GO TO 308                                                       00011200
   3    303 WRITE(KIO6,2032)                                                00011300
   7        IC=4                                                            00011400
   7        GO TO 308                                                       00011500
   3    305 IC=IC + 5                                                       00011600
   7        WRITE(KIO6,2030)                                                00011700
   3    308 DO 350 IB=1,NPB                                                 00011800
   7        NODE1=IPNODE(1,IB)                                              00011900
   7        IF (NODE1.EQ.0) GO TO 350                                       00012000
   7        NODE2=IPNODE(2,IB)                                              00012100
   7        IF (IND.EQ.0) NODE1=1                                           00012200
   7        IF (IND.EQ.0) NODE2=NUMNP                                       00012300
   1  C                                                                     00012400
   7        DO 300 II=NODE1,NODE2                                           00012500
   7        IC=IC + 1                                                       00012600
   7        IF (IC.LT.56) GO TO 307                                         00012700
   7        WRITE(KIO6,2032)                                                00012800
   7        IC=4                                                            00012900
   3    307 DO 310 I=1,6                                                    00013000
   3    310 D(I)=0.                                                         00013100
   7        DO 320 I=1,NDOF                                                 00013200
   7        KK=ID(I,II)                                                     00013300
   7        IL=I                                                            00013400
   7        IF (NDOF.EQ.2) IL=I + 1                                         00013500
   3    320 IF (KK.NE.0) D(IL)=ACC(KK)                                      00013600
   3    300 WRITE(KIO6,2010) II,D                                           00013700
   1  C                                                                     00013800
   7        IF (IND.EQ.0) RETURN                                            00013900
   7        IF (IC.GE.55) GO TO 350                                         00014000
   7        IC=IC+1                                                         00014100
   7        WRITE(KIO6,2050)                                                00014200
   3    350 CONTINUE                                                        00014300
   1  C                                                                     00014400
   7        RETURN                                                          00014500
   1  C                                                                     00014600
   2   2000 FORMAT (/27H  D I S P L A C E M E N T S  // 7H  NODE 12X        00014700
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   6        114HX-DISPLACEMENT 4X 14HY-DISPLACEMENT 4X 14HZ-DISPLACEMENT        00014800
   6        28X 10HX-ROTATION 8X 10HY-ROTATION 8X 10HZ-ROTATION  /)             00014900
   2    2010 FORMAT (2X,I5,8X,6E18.6)                                          00015000
   2    2020 FORMAT(/22H  V E L O C I T I E S  // 7H  NODE 16X 10HX-VELOCITY    00015100
   6        18X 10HY-VELOCITY 8X 10HZ-VELOCITY                                 00015200
   6        28X 10HX-ROTATION 8X 10HY-ROTATION 8X 10HZ-ROTATION  /)            00015300
   2    2022 FORMAT (1H1,21H V E L O C I T I E S   //7H  NODE 16X 10HX-VELOCITY00015400
   6        18X 10HY-VELOCITY 8X 10HZ-VELOCITY 8X 10HX-ROTATION 8X             00015500
   6        310HY-ROTATION 8X 10HZ-ROTATION  /)                                00015600
   2    2030 FORMAT (/27H  A C C E L E R A T I O N S  // 7H  NODE 12X          00015700
   6        114HX-ACCELERATION 4X 14HY-ACCELERATION 4X 14HZ-ACCELERATION       00015800
   6        28X 10HX-ROTATION 8X 10HY-ROTATION 8X 10HZ-ROTATION  /)            00015900
   2    2032 FORMAT (1H1,26H A C C E L E R A T I O N S  // 7H  NODE 12X        00016000
   6        114HX-ACCELERATION 4X 14HY-ACCELERATION 4X 14HZ-ACCELERATION       00016100
   6        28X 10HX-ROTATION 8X 10HY-ROTATION 8X 10HZ-ROTATION  /)            00016200
   2    2045 FORMAT (1H1, 26H D I S P L A C E M E N T S  // 7H  NODE 12X       00016300
   6        114HX-DISPLACEMENT 4X 14HY-DISPLACEMENT 4X 14HZ-DISPLACEMENT       00016400
   6        28X 10HX-ROTATION 8X 10HY-ROTATION 8X 10HZ-ROTATION  /)            00016500
   2    2050 FORMAT (1H )                                                      00016600
   1    C                                                                      00016700
   7         END                                                               00016800
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7        SUBROUTINE WRTFRQ  (FRQ,PHI,ID,NUMNP,NDOF,NEQ,IM)                    00000100
   7        IMPLICIT REAL*8(A-H,O-Z)                                             00000200
   7        COMMON/BBCCIO/KI01,KI02,KI03,KI04,KI05,KI06,KI07,KI08,KI09,          00000300
   6       1KI010,KI011,KI012,KI013,KI014,KI015,KI016,KI017,KI018,KI019,KI020,00000400
   6       2KI021,KI022,KI023,KI024,KI025,KI026,KI027,KI028,KI029,KI030         00000500
   1  C                                                                         00000600
   7        DIMENSION FRQ(1),PHI(1),ID(NDOF,1)                                  00000700
   7        DIMENSION D(6)                                                      00000800
   1  C                                                                         00000900
   1  C     PRINT EIGENVECTOR FOR MODE IM                                       00001000
   1  C                                                                         00001100
   7        WRITE(KI06,1000) IM,FRQ(IM)                                         00001200
   7        NB=NEQ*(IM-1)                                                       00001300
   7        DO 100 II=1,NUMNP                                                   00001400
   7        DO 110 I=1,6                                                        00001500
   3    110 D(I)=0.                                                             00001600
   7        DO 120 I=1,NDOF                                                     00001700
   7        KK=ID(I,II)                                                         00001800
   7        IL=I                                                                00001900
   7        IF (NDOF.EQ.2) IL=I + 1                                             00002000
   3    120 IF (KK.NE.0) D(IL)=PHI(NB+KK)                                       00002100
   3    100 WRITE(KI06,1010) II,D                                               00002200
   7        RETURN                                                              00002300
   1  C                                                                         00002400
   2   1000 FORMAT (///29H   M O D E  S H A P E  N O .  I3,                     00002500
   6          1       55X,14H( FREQUENCY =  ,E11.4, 2H ) //                     00002600
   6          2       9H    NODE 12X 14HX-DISPLACEMENT 4X 14HY-DISPLACEMENT 4X  00002700
   6          3       14HZ-DISPLACEMENT 8X 10HX-ROTATION 8X 10HY-ROTATION       00002800
   6          4       8X 10HZ-ROTATION  /)                                      00002900
   2   1010 FORMAT (2X,I5,8X,6E18.6)                                           00003000
   1  C                                                                         00003100
   7        END                                                                 00003200
```

```
START                                                                                MOD
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8  FLAGS

   7        COMMON /EL/ IND,ICOUNT,NPAR(20),NUMEG,NEGL,NEGNL,IMASS,IDAMP,ISTAT00001000
   6       1           ,NDOFDM,KLIN,IEIG,IMASSN,IDAMPN                        00001100
   7        COMMON /SOL/ NUMNP,NEQ,NWK,NWM,NWC,NUMEST,MIDEST,MAXEST,NSTE,MA   00001200
   7        COMMON /DIM/ N1,N2,N3,N4,N5,N6,N7,N8,N9,N10,N11,N12,N13,N14,N15   00001300
   7        COMMON /CONST/ DT,DTA,A0,A1,A2,A3,A4,A5,A6,A7,A8,A9,A10,A11       00001400
   6       1            ,A12,A13,A14,A15,A16,A17,A18,A19,A20,IOPE            00001500
   7        COMMON /DAMPNG/ ADAMP,BDAMP                                       00001600
   7        COMMON /VAR/ NG,KPRI,MODEX,KSTEP,ITE,ITEMAX,IREF,IEQREF,INOCMD    00001700
   7        COMMON /EM/ S(136),XM(16),B(4,16),NOD(8),NODM(8),RE(16),          00001800
   6       1            EDIS(16),AEM(204),XX(16)                             00001900
   7        COMMON /MATMOD/ STRESS(4),STRAIN(4),D(4,4),IPT,N                  00002000
   7        COMMON /TODIM/ BET,THIC,DE,IEL,NND5                               00002100
   7        COMMON /DISDER/ DISD(5)                                           00002200
   7        COMMON A(1)                                                       00002300
   1  C                                                                       00002400
   7        DIMENSION ID(NDOF,1), Y(1), Z(1), HT(1), LM(NDM,1), YZ(NDM,1),    00002500
   6       1          IELT(1), IPST(1), BETA(1), THICK(1), MATP(1),          00002600
   6       2          DEN(1),PROP(NCON,1),WA(IDWA,1),NOD5(ND5DIM,1),         00002700
   6       3          ITABLE(NTABLE,9)                                       00002800
   7        DIMENSION EVAL(9,2),XG(4,4),NOD5M(4)                              00002900
   1  C                                                                       00003000
   7        EQUIVALENCE (NPAR(2),NUME),(NPAR(16),NUMMAT),(NPAR(15),MODEL)     00003100
   6       1           ,(NPAR(10),NINT),(NPAR(1),NPAR1),(NPAR(7),MXNODS)     00003200
   6       2           ,(NPAR(5),ITYP2D),(NPAR(3),INDNL)                     00003300
   1  C                                                                       00003400
   7        DATA XG /     0.,             0.,             0.,            0.,  00003500
   6       1 -.5773502691896, .5773502691896,             0.,            0.,  00003600
   6       2 -.7745966692415, .0000000000000, .7745966692415,            0.,  00003700
   6       3 -.8611363115941,-.3399810435849, .3399810435849, .8611363115941/ 00003800
   1  C                                                                       00003900
   7        DATA EVAL / 1.0,-1.0,-1.0, 1.0, 0.0,-1.0, 0.0, 1.0, 0.0,          00004000
   6       1            1.0, 1.0,-1.0,-1.0, 1.0, 0.0,-1.0, 0.0, 0.0/         00004100
   7        DATA ANODE/4HNODE/                                                00004200
   1  C                                                                       00004300
   1  C     ** NOTE ** DURING TIME INTEGRATION Y=DISP, Z=VEL, HT=R            00004400
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8  MOD FLAGS

   7        IMPLICIT REAL*8(A-H,O-Z)                                                 00002100
   7        COMMON/BBCCIO/KI01,KI02,KI03,KI04,KI05,KI06,KI07,KI08,KI09,              00002200
   6       1KI010,KI011,KI012,KI013,KI014,KI015,KI016,KI017,KI018,KI019,KI020,00002300
   6       2KI021,KI022,KI023,KI024,KI025,KI026,KI027,KI028,KI029,KI030             00002400
   7        DIMENSION X(1),Y(1),Z(1),ID(NDOF,1)                                      00002500
   7        DIMENSION IDT(6),IDTOLD(6)                                               00002600
   7        DIMENSION IPRC(4)                                                        00002700
```

PROJECT: TSO
LIBRARY: TDA10
TYPE: FH2079

DATE: 82/08/25
TIME: 14:24
PAGE: 001

GENERAL DATA:
VOLUME SERIAL: 501004
DEVICE TYPE: 3350
ORGANIZATION: PO
CREATION DATE: 82/08/18

GENERAL DATA:
RECORD FORMAT: FB
RECORD LENGTH: 80
BLOCK SIZE: 6,160
1ST EXTENT SIZE: 65
SECONDARY QUAN: 0

CURRENT ALLOCATION:
65 TRACKS
1 EXTENT
30 DIRECTORY BLOCKS

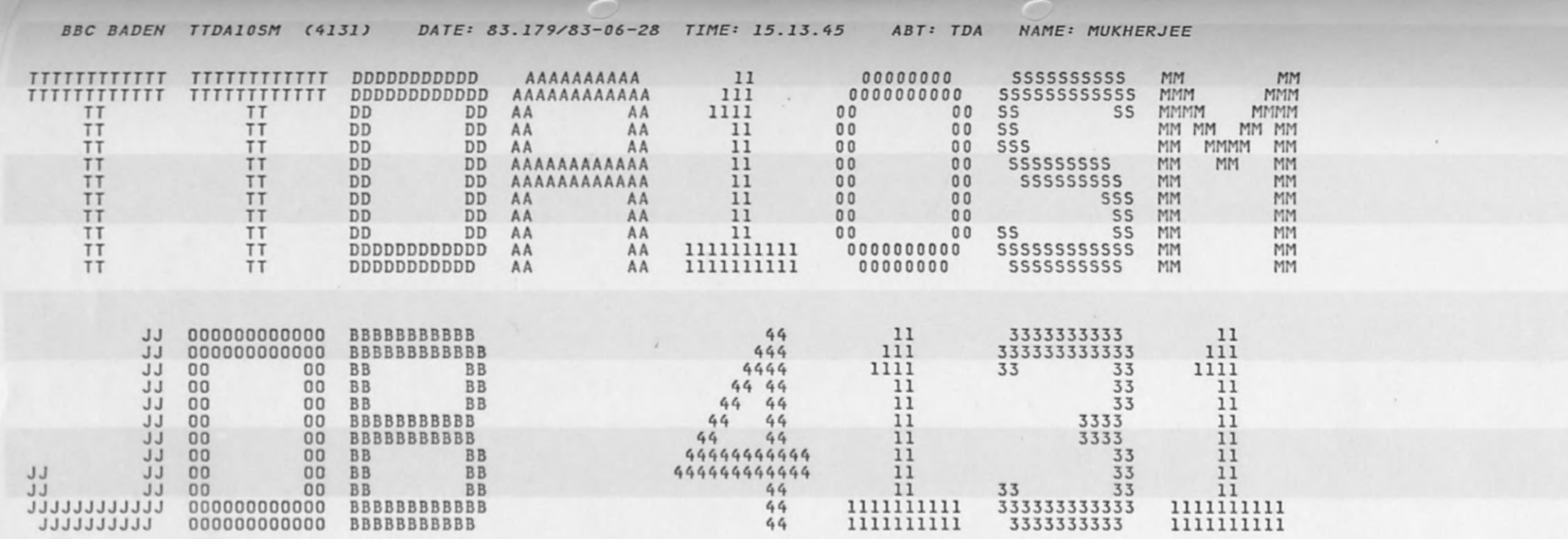
CURRENT UTILIZATION:
63 TRACKS
1 EXTENT
8 DIRECTORY BLOCKS
90 MEMBERS

MEMBER NAME	VERS.MOD LEVEL	CREATION DATE	DATE AND TIME LAST MODIFIED	CURRENT NO. LINES	INITIAL NO. LINES	MODIFIED NO. LINES	USER ID
AAMAIN	01.01	82/08/24	82/08/25 09:34	575	575	1	TDA10
ADDBAN							
ADDCM	01.02	82/08/18	82/08/19 13:46	59	60	2	TDA10
ADDDI							
ADDMA							
ADDRES							
ASSEM							
BANDET							
CAUCHY							
CDMOD	01.03	82/08/18	82/08/20 13:57	334	340	4	TDA10
CMOD3D							
COLHT							
COLSOL							
DCRACK							
DERIQ							
DERIQ3							
DRUCK							
ELCAL							
ELEMNT							
ELPAL							
ELTTT6							
ELT2D3							
ELT2D4							
ELT2D6							
ELT2D7							
ELT2D8							
ELT2D9							
ELT3D3							
ELT3D4							
ELT3D5							
ELT3D6							
EL2D10							
EL2D11							

MEMBER NAME	VERS.MOD LEVEL	CREATION DATE	DATE AND TIME LAST MODIFIED	CURRENT NO. LINES	INITIAL NO. LINES	MODIFIED NO. LINES	USER ID
EL2D12							
EQUIT							
ERROR							
FREQS	01.02	82/08/18	82/08/18 17:51	130	134	1	TDA10
FUNCT							
FUNCT2							
ICDMOD	01.00	82/08/19	82/08/19 14:01	113	113	0	TDA10
ICMOD3							
IDRUCK							
IELPAL							
INITAL							
INITWA							
INPUT							
INTWA3							
IVTMOD							
LOADEF							
LOADS							
MATBAR							
MATRT2							
MATWRT							
MAXAEX	01.01	82/08/18	82/08/18 17:46	10	10	1	TDA10
MAXMIN							
MIDEP							
MLTPLY							
MOONEY							
MULT							
NEWDAV							
NODMAS							
NONSPI							
OPCOEF							
PNORM							
POSINV							
PRAGER							
QUADM							
QUADM3							
QUADS							
QUADS3							
READID	01.01	82/08/18	82/08/18 17:48	15	15	1	TDA10
RUSS							
SECANT	01.00	82/08/19	82/08/19 14:25	612	612	0	TDA10
STIME							

MEMBER NAME	VERS.MOD LEVEL	CREATION DATE	DATE AND TIME LAST MODIFIED	CURRENT NO. LINES	INITIAL NO. LINES	MODIFIED NO. LINES	USER ID
STRESS							
STSTL							
STSTN							
STST3L							
STST3N							
TDFE	01.00	82/08/19	82/08/19 14:43	532	532	0	TDA10
THDFE	01.01	82/08/19	82/08/19 15:13	540	539	3	TDA10
THREDM							
TODMFE							
TRUSS							
TTIME							
VTMOD							
WRITE							
WRTFRQ							
XYZ	01.00	82/08/19	82/08/19 16:39	35	35	0	TDA10
ZZ	01.00	82/08/19	82/08/19 16:37	7	7	0	TDA10
MAXIMUMS:	01.03	82/08/24	82/08/25 09:34	612	612	4	
TOTALS:				2,962	2,972	13	

END OF MEMBER LIST


TTTTTTTTTTTT TTTTTTTTTTTT DDDDDDDDDDD AAAAAAAAAA 11 00000000 SSSSSSSSSS MM MM
TTTTTTTTTTTT TTTTTTTTTTTT DDDDDDDDDDDD AAAAAAAAAAAA 111 0000000000 SSSSSSSSSSSS MMMM MMMM
TT TT DD DD AA AA 1111 00 00 SS SS MMMMM MMMMM
TT TT DD DD AA AA 11 00 00 SS MM MM MM MM
TT TT DD DD AA AA 11 00 00 SSS MM MMMM MM
TT TT DD DD AAAAAAAAAAAA 11 00 00 SSSSSSSSS MM MM MM
TT TT DD DD AAAAAAAAAAAA 11 00 00 SSSSSSSSS MM MM
TT TT DD DD AA AA 11 00 00 SSS MM MM
TT TT DD DD AA AA 11 00 00 SS MM MM
TT TT DD DD AA AA 11 00 00 SS SS MM MM
TT TT DDDDDDDDDDDD AA AA 1111111111 0000000000 SSSSSSSSSSS MM MM
TT TT DDDDDDDDDDD AA AA 1111111111 00000000 SSSSSSSSS MM MM
JJ 000000000000 BBBBBBBBBB 44 11 3333333333 11
JJ 000000000000 BBBBBBBBBBB 444 111 333333333333 111
JJ OO OO BB BB 4444 1111 33 33 1111
JJ OO OO BB BB 44 44 11 33 11
JJ OO OO BB BB 44 44 11 33 11
JJ OO OO BBBBBBBBBB 44 44 11 3333 11
JJ OO OO BBBBBBBBBB 44 44 11 3333 11
JJ OO OO BB BB 44444444444 11 33 11
JJ JJ OO OO BB BB 44444444444 11 33 11
JJ JJ OO OO BB BB 44 11 33 33 11
JJJJJJJJJJJJ 000000000000 BBBBBBBBBBB 44 1111111111 333333333333 1111111111
JJJJJJJJJ 000000000000 BBBBBBBBBB 44 1111111111 3333333333 1111111111

```
C  SUBROUTINE FOR SUPPRESSING UNWANTED REFLECTED VELOCITY FROM THE
C  BOUNDARY.(VELOCITY CORRECTION=VELCOR)
C  PROGRAMMED BY S.N.MUKHERJEE.BBC,BADEN AUG,1977
C UPDATED: MARCH,1978. SEPT,1980.
C NODAL POINT DATA AND ELEMENT PROPERTIES ARE READ IN NONSAP.
C ELEMENT STIFFNESS MATRIX AND ASSEMBLED GLOBAL STIFFNESS MATRIX,
C MASS MATRIX ARE FORMED IN NONSAP AND USED IN VELCOR.
C**************************************************************************
C TO USE VELCOR WITH ANY OTHER FINITE ELEMENT PROGRAM WHICH HAS
C THE CAPBILITIES OF DYNAMIC ANALYSIS THE FOLLOWING IN PUT DATA ARE
C REQUIRED UNLESS OR  OTHERWISE AVAILABLE.
C  I = MATERIAL IDENTIFICATION NUMBER
C  BLK(I) =BULK MODULUS.
C  SHR(I) = SHEAR MODULUS
C  ALPHA(I) = ANGLE OF FRICTION
C  CELL(I)  = COHESION
C  KBP = NUMBER OF NODAL POINTS WHERE  KINEMATIC INPUT PRESCRIBED
C  FORMAT I3
C  (KBP(I),I=1,KBP) =NODAL POINT NUMBER WITH KINEMATIC INPUT
C  NQ =NUMBER OF NODAL POINTS TO BE QUIETED FORMAT(I3)
C  A BLANK CARD FOR NQ=0
C  IF AN INPUT IS GIVEN IN NQ THEN THE FOLLOWING CARDS MUST BE SUPPLIED.
C
C  ((IQ(I,J),J=1,5,I=NQ)      FORMAT (5I5)
C  IQ(I,1) = TYPE OF BOUNDARY I=1 VERTICAL BOUNDARY, I=2 HORIZONTAL
C  BOUNDARY
C  IQ(1,2) = NODAL POINT NUMBER OF CORRECTOR
C  IQ(1,3) = NODAL POINT NUMBER OF SENSOR
C  IQ(1,4) = MATERIAL PROPERTY INDEX
C  IQ(1,5) =SEQUENCE NUMBER OF THE POINT IN THE KBNP ARRAY.
C **************************************************************************
C
0001          SUBROUTINE VELCOR(X0,X1,X2,R,BLK,SHR,RO,TT,DELT,NNN,NUMNP,NUMAT,
             1BETA,MBAND,B,NEQ,KODE,A)
0002          COMMON /EL/ IND,ICOUNT,NPAR(20),NUMEG,NEGL,NEGNL,IMASS,IDAMP,ISTAT00001000
             1            ,NDOFDM,KLIN,IEIG,IMASSN,IDAMPN                    00001100
0003          COMMON /SOL/ NWK,NWM,NWC,NUMEST,MIDEST,MAXEST,NSTE,MA           00001200
0004          COMMON /DIM/ N1,N2,N3,N4,N5,N6,N7,N8,N9,N10,N11,N12,N13,N14,N15 00001300
0005          COMMON /CONST/ DT,DTA,A0,A1,A2,A3,A4,A5,A6,A7,A8,A9,A10,A11     00001400
             1             ,A12,A13,A14,A15,A16,A17,A18,A19,A20,IOPE         00001500
0006          COMMON /DAMPNG/ ADAMP,BDAMP                                     00001600
0007          COMMON /VAR/ NG,KPRI,MODEX,KSTEP,ITE,ITEMAX,IREF,IEQREF,INOCMD  00001700
C             COMMON /EM/ S(136),XM(16),B(4,16),NOD(8),NODM(8),RE(16),        00001800
C            1            EDIS(16),AEM(204),XX(16)                           00001900
0008          COMMON /MATMOD/ STRESS(4),STRAIN(4),D(4,4),IPT,N                00002000
0009          COMMON /TODIM/ BET,THIC,DE,IEL,NND5                             00002100
0010          COMMON /DISDER/ DISD(5)                                         00002200
0011          COMMON/KINBC/KBP,KBNP(60),NTIME,NQ,IQ(60,5),DV(120),V(150,3),
             &IV(60)
C             COMMON A(1)                                                     00002300
C                                                                             00002400
0012          DIMENSION Y(1), Z(1), HT(1),                                    00002500
             1          IELT(1), IPST(1), BETA(1), THICK(1), MATP(1),        00002600
```

```
                     2          DEN(1)                                                    00002700
0013                  DIMENSION EVAL(9,2),XG(4,4),NOD5M(4),X0(1),X1(1),X2(1),R(1),      00002900
                     & BLK(1),SHR(1),RO(1),B(1),KODE(1),CPS(15,2),VJ(2),A(550,1)       00003000
0014                  EQUIVALENCE (NPAR(2),NUME),(NPAR(16),NUMMAT),(NPAR(15),MODEL)     00003100
                     1           ,(NPAR(10),NINT),(NPAR(1),NPAR1),(NPAR(7),MXNODS)     00003200
                     2           ,(NPAR(5),ITYP2D),(NPAR(3),INDNL)                     00003300
               C                                                                       00003400
0015                  DATA XG /     0.,              0.,              0.,          0., 00003500
                     1 -.5773502691896, .5773502691896,              0.,          0., 00003600
                     2 -.7745966692415, .0000000000000, .7745966692415,          0., 00003700
                     3 -.8611363115941,-.3399810435849, .3399810435849, .8611363115941/ 00003800
               C                                                                       00003900
0016                  DATA EVAL / 1.0,-1.0,-1.0, 1.0, 0.0,-1.0, 0.0, 1.0, 0.0,          00004000
                     1            1.0, 1.0,-1.0,-1.0, 1.0, 0.0,-1.0, 0.0, 0.0/        00004100
0017                  DATA ANODE/4HNODE/                                               00004200
               C                                                                       00004300
               C     ** NOTE ** DURING TIME INTEGRATION Y=DISP, Z=VEL, HT=R            00004400
               C
0018                  KPB2=2*KPB
0019                  DO 100 I=1,KPB2
0020                  DV(I)=0.
0021                100 CONTINUE
0022                  IF (NQ.EQ.0 )GO TO 500
0023                  IF(NNN.NE.1) GO TO 240
               C
0024                  DO 120 I=1,NUMAT
0025                  BLKM=BLK(I)
0026                  SHRM=SHR(I)
0027                  YMOD=9.0*BLKM*SHRM/(3.*BLKM+SHRM)
0028                  U=(3.0*BLKM-2.0*SHRM)/(6.0*BLKM+2.0*SHRM)
0029                  CPS(I,1)=SQRT(YMOD*(1.-U)/(RO(I)*(1.+U)*(1.-2.*U)))
0030                  CPS(I,2)=SQRT(YMOD/(2.*RO(I)*(1.+U)))
0031                120 CONTINUE
               C
               C    CALCULATE CORRECTION TO PRESCRIBED VELOCITIES
               C
0032                240 IF (NTIME. EQ.0) GO TO 260
0033                  T=TT-DELT
0034                  VJ(1)= VELINT(T,2,V,1,150)
0035                  VJ(2)= VELINT(T,3,V,0.150)
0036                260 DO 490 I=1,NQ
0037                  I1=IQ(I,1)
0038                  I2=IQ(I,2)
0039                  I3=IQ(I,3)
0040                  I4=IQ(I,4)
0041                  I5=IQ(I,5)
               C
               C     I1= TYPE OF BOUNDARY(1,OR 2)
               C     I2= NODE NUMBER OF CORRECTER
               C     I3= NODE NUMBER OF SENSOR
               C     I4= MATERIAL PROPERTIES INDEX
               C     I5= SEQUENCE NUMBER IN KBNP ARRAY(NODAL POINT NUMBER WITH
               C     KINEMATIC INPUT FOR EXAMPLE VELOCITY OR ACCELERATION T.H )
```

```
                C
0042                  IF (I1.EQ.1) I0=0
0043                  IF (I1.NE.1)I0=275
                C
0044                  DO 480 J=1,2
0045                  DEL=ABS(R(I2+I0)-R(I3+I0))/CPS(I4,J)
0046                  T=TT-DELT-DEL
0047                  IF(T.LE.0.0) GO TO 480
0048                  IF(I1.EQ.1) N=J
0049                  IF(I1.NE.1) N=3-J
0050                  DUM1 =X1(2*I3-2+N)
0051                  DUM2=X1(2*I2-2+N)
0052                  IF(NTIME.EQ.0.) GO TO 460
0053                  IF(IV(I5).EQ.0) GO TO 460
0054                  DUM1=DUM1-VELINT(T,N+1,V,1,150)
0055                  DUM2=DUM2-VJ(N)
0056           460    DV(2*I5-2+N)=(DUM2*(DEL-DELT)+DUM1*DELT)/DEL
0057           480    CONTINUE
0058           490    CONTINUE
0059                  IF (NTIME.EQ.0) GO TO 600
                C
                C     ADD SPECIFIED VELOCITIES TO THE CORRECTION
0060           500    T=TT
0061                  VI1=VELINT(T,2,V,1,150)
0062                  VI2=VELINT(T,3,V,0,150)
0063                  DO 520 I=1,KBP
0064                  IF (IV(I).EQ.0) GO TO 520
0065                  I2=2*I
0066                  I1=I2-1
0067                  DV(I1)=DV(I1)+VI1
0068                  DV(I2)=DV(I2)+VI2
0069           520    CONTINUE
                C  CALCULATE THE INCREMENTAL DISPLACEMENTS CORRESPONDING TO THE
                C  SPECIFIED INCREMENTAL VELOCITIES
                C
0070           600    CONST=DELT/3.0
0071                  DO 620 I=1,KBP
0072                  N=KBNP(I)
0073                  I2=2* I
0074                  I1=I2-1
0075                  N2=2*N
0076                  N1=N2-1
0077                  DV(I1)=CONST*(DV(I1)+2.0*X1(N1)+0.5*DELT*X2(N1))
0078                  DV(I2)=CONST*(DV(I2)+2.0*X1(N2)+0.5*DELT*X2(N2))
0079           620    CONTINUE
                C
                C  MODIFY LOAD VECTOR FOR KINEMATIC INPUT.
                C
0080                  DO 820 I= 1,KBP
0081                  N=KBNP(I)
0082                  N2=2*N
0083                  N1=N2-1
0084                  A(N1,1)=1.0
```

```
0085                    A(N2,1)=1.0
0086                    DO 760 J=2,MBAND
0087                    J1=J-1
0088                    N11=NI+J1
0089                    N22=N2+J1
0090                    B(N11)= B(N11)-A(N1,J)*DV(2*I-1)
0091                    A(N1,J)=0.0
0092                    IF(N22.GT.NEQ) GO TO 780
0093                    B(N22)=B(N22)-A(N2,J)*DV(2*I)
0094                760 A(N2,J)=0.0
0095                780 CONTINUE
0096                    DO 800 J=2,MBAND
0097                    J1=J-1
0098                    N3=N1-J1
0099                    N4=N2-J1
0100                    B(N4)=B(N4)-A(N4,J)*DV(2*I)
0101                    A(N4,J)=0.0
0102                    IF(N3.LT.1) GO TO 820
0103                    B(N3)=B(N3)-A(N3,J)*DV(2*I-1)
0104                800 A(N3,J)=0.0
0105                820 CONTINUE
                    C SET LOADS CORRESP.TO CONSTRAINED POINTS TO SPECIFIC VALUES
                    C
0106                840 DO 890 N=1,NUMNP
0107                    KK=KODE(N)
0108                    IF (KK.EQ.0)GO TO 890
0109                    IF(KK-10) 870,880,860
0110                860 B(2*N-1)=0.
0111                870 B(2*N)=0.
0112                    GO TO 890
0113                880 B(2*N-1)=0.
0114                890 CONTINUE
                    C
0115                    DO 900 I=1,KBP
0116                    N=KBNP(I)
0117                    B(2*N-1)=DV(2*I-1)
0118                    B(2*N)  =DV(2*I )
0119                900 CONTINUE
                    C
0120                    RETURN
0121                    END
```

COMMON BLOCK /EL / MAP SIZE 84

SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION
IND	0	ICOUNT	4	NPAR	8	NPAR1	8	NUME	C
INDNL	10	ITYP2D	18	MXNODS	20	NINT	2C	MODEL	40
NUMMAT	44	NUMEG	58	NEGL	5C	NEGNL	60	IMASS	64
IDAMP	68	ISTAT	6C	NDOFDM	70	KLIN	74	IEIG	78
IMASSN	7C	IDAMPN	80						

COMMON BLOCK /SOL / MAP SIZE 20

SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION
NWK	0	NWM	4	NWC	8	NUMEST	C	MIDEST	10
MAXEST	14	NSTE	18	MA	1C				

COMMON BLOCK /DIM / MAP SIZE 3C

SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION
N1	0	N2	4	N3	8	N4	C	N5	10
N6	14	N7	18	N8	1C	N9	20	N10	24
N11	28	N12	2C	N13	30	N14	34	N15	38

COMMON BLOCK /CONST / MAP SIZE 60

SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION
DT	0	DTA	4	A0	8	A1	C	A2	10
A3	14	A4	18	A5	1C	A6	20	A7	24
A8	28	A9	2C	A10	30	A11	34	A12	38
A13	3C	A14	40	A15	44	A16	48	A17	4C
A18	50	A19	54	A20	58	IOPE	5C		

COMMON BLOCK /DAMPNG / MAP SIZE 8

SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION
ADAMP	0	BDAMP	4						

COMMON BLOCK /VAR / MAP SIZE 24

SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION
NG	0	KPRI	4	MODEX	8	KSTEP	C	ITE	10
ITEMAX	14	IREF	18	IEQREF	1C	INOCMD	20		

COMMON BLOCK /MATMOD / MAP SIZE 68

SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION
STRESS	0	STRAIN	10	D	20	IPT	60	N	64

COMMON BLOCK /TODIM / MAP SIZE 14

SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION
BET	0	THIC	4	DE	8	IEL	C	NND5	10

COMMON BLOCK /DISDER / MAP SIZE 14

SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION
DISD	0								

COMMON BLOCK /KINBC / MAP SIZE F84

SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION
KBP	0	KBNP	4	NTIME	F4	NQ	F8	IQ	FC
DV	5AC	V	78C	IV	E94				

SUBPROGRAMS CALLED

SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION
VELINT	188	SQRT	18C						

SCALAR MAP

SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION
ANODE	1FC	KPB2	200	KPB	204	I	208	NNN	20C
NUMAT	210	BLKM	214	SHRM	218	YMOD	21C	U	220
T	224	TT	228	DELT	22C	I1	230	I2	234
I3	238	I4	23C	I5	240	I0	244	J	248
DEL	24C	DUM1	250	DUM2	254	VI1	258	VI2	25C
CONST	260	MBAND	264	J1	268	NI	26C	N22	270
NEQ	274	NUMNP	278	KK	27C				

ARRAY MAP

SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION
Y	280	Z	284	HT	288	IELT	28C	IPST	290
BETA	294	THICK	298	MATP	29C	DEN	2A0	EVAL	2A4
XG	2EC	NOD5M	32C	X0	33C	X1	340	X2	344
R	348	BLK	34C	SHR	350	RO	354	B	358
KODE	35C	CPS	360	VJ	3D8	A	3E0		

STATEMENT LABEL MAP

LOCATION	STA NUM	LABEL	LOCATION	STA NUM	LABEL	LOCATION	STA NUM	LABEL
0005C0	1		0005C0	18		0005CC	19	
0005D8	20		0005E0	21	100	0005F8	22	
00060A	23		000618	24		00063C	25	
000644	26		00064C	27		00066A	28	
000696	29		0006D6	30		000704	31	120
000728	32	240	00073A	33		000746	34	
000758	35		00076A	36	260	000776	37	
00077E	38		000786	39		00078E	40	
000796	41		00079E	42		0007B4	43	
0007CA	44		0007D6	45		000828	46	
000838	47		000846	48		000860	49	
00087E	50		0008A8	51		0008CA	52	
0008F4	53		00090A	54		00092E	55	
000942	56	460	00097E	57	480	000996	58	490
0009B2	59		0009C4	60	500	0009CC	61	
0009DE	62		0009F0	63		0009FC	64	
000A0A	65		000A16	66		000A22	67	
000A3A	68		000A4E	69	520	000A6A	70	600
000A76	71		000A82	72		000A8E	73	
000A9A	74		000AA6	75		000AB6	76	
000AC2	77		000B04	78		000B4A	79	620
000B62	80		000B7E	81		000B8E	82	
000B9E	83		000BAA	84		000BBA	85	
000BCA	86		000BDA	87		000BE6	88	
000BF6	89		000C02	90		000C2C	91	
000C3C	92		000C4A	93		000C92	94	760
000CBA	96	780	000CCE	97		000CDA	98	
000CEA	99		000CF6	100		000D42	101	
000D52	102		000D60	103		000DA4	104	800
000DCC	105	820	000DF4	106	840	000E10	107	
000E18	108		000E26	109		000E3C	110	860
000E56	111	870	000E78	112		000E7E	113	880
000E98	114	890	000EB4	115		000ECC	116	
000ED8	117		000F04	118		000F3C	119	900
000F54	120							

OPTIONS IN EFFECT NOTERM,ID,EBCDIC,SOURCE,NOLIST,NODECK,LOAD,MAP,NOTEST
OPTIONS IN EFFECT NAME = VELCOR , LINECNT = 55
STATISTICS SOURCE STATEMENTS = 121,PROGRAM SIZE = 3932
STATISTICS NO DIAGNOSTICS GENERATED

```
0001                FUNCTION VELINT(T,N,V,K,NTIME)
                    C
0002                DIMENSION V(NTIME,1)
0003                IF(K.NE.1) GO TO 30
0004                J=1
0005              10 IF(T.GE.V(J,1).AND.T.LT.V(J+1,1))GO TO 20
0006                J=J+1
0007                GO TO 10
0008              20 DT1=V(J+1,1)-V(J,1)
0009                DT2=T-V(J,1)
0010              30 DV=V(J+1,N)-V(J,N)
0011                VELINT=V(J,N)+DV*DT2/DT1
                    C
0012                RETURN
0013                END
```

EQUIVALENCE DATA MAP

SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION
VELINT	B0								

SCALAR MAP

SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION
NTIME	B4	K	B8	J	BC	T	C0	DT1	C4
DT2	C8	DV	CC	N	D0				

ARRAY MAP

SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION
V	D4								

STATEMENT LABEL MAP

LOCATION	STA NUM	LABEL	LOCATION	STA NUM	LABEL	LOCATION	STA NUM	LABEL
0001CA	1		0001CA	3		0001D8	4	
0001E0	5	10	00023E	6		00024A	7	
000250	8	20	000298	9		0002B6	10	30
000308	11		000332	12				

OPTIONS IN EFFECT NOTERM,ID,EBCDIC,SOURCE,NOLIST,NODECK,LOAD,MAP,NOTEST
OPTIONS IN EFFECT NAME = VELINT , LINECNT = 55
STATISTICS SOURCE STATEMENTS = 13,PROGRAM SIZE = 826
STATISTICS NO DIAGNOSTICS GENERATED

```
0001                  SUBROUTINE STRESS(XC,YC)
                    C SUBROUTINE FOR COMPUTING FINAL STRESSES
0002                  REAL MASS
0003                  COMMON/SS/ALFA,BETA,DELT,MBAND,N,NEQ,NNN,NP,NT,NUMP,NUMEL,NUMAT,
                     *TT,VOL,DUMMY(5),A(550,36),SIG(250,7),B(550),BO(550),CC(1500),
                     *C(3,3),CH(275),CV(275),EPS(250,4),H(6000),IX(250,5),KODE(275),
                     *MASS(275),P(3,50),R(275),Z(275),S(10,10),T(275),X0(550),X1(550),
                     *X2(550)
0004                  COMMON /EL/ IND,ICOUNT,NPAR(20),NUMEG,NEGL,NEGNL,IMASS,IDAMP,ISTAT00001000
                     1           ,NDOFDM,KLIN,IEIG,IMASSN,IDAMPN                      00001100
0005                  COMMON /SOL/ NUMNP,NWK,NWM,NWC,NUMEST,MIDEST,MAXEST,NSTE,MA      00001200
0006                  COMMON /DIM/ N1,N2,N3,N4,N5,N6,N7,N8,N9,N10,N11,N12,N13,N14,N15  00001300
0007                  COMMON /CONST/ DT,DTA,A0,A1,A2,A3,A4,A5,A6,A7,A8,A9,A10,A11      00001400
                     1            ,A12,A13,A14,A15,A16,A17,A18,A19,A20,IOPE          00001500
0008                  COMMON /DAMPNG/ ADAMP,BDAMP                                     00001600
0009                  COMMON /VAR/ NG,KPRI,MODEX,KSTEP,ITE,ITEMAX,IREF,IEQREF,INOCMD  00001700
                    C     COMMON /EM/ S(136),XM(16),B(4,16),NOD(8),NODM(8),RE(16),    00001800
                    C    1            EDIS(16),AEM(204),XX(16)                       00001900
0010                  COMMON /MATMOD/ D(4,4),IPT                                      00002000
0011                  COMMON /TODIM/ BET,THIC,DE,IEL,NND5                             00002100
0012                  COMMON /DISDER/ DISD(5)                                         00002200
0013                  COMMON /ELPLS/STRESS(4),STRAIN(4),STRINC(4),BM,G,ALPH,CEE,B11,
                     *B2,B3,U1,U2
                    C     COMMON A(1)                                               00002300
                    C                                                               00002400
0014                  DIMENSION Y(1), HT(1),                                        00002500
                     1          IELT(1), IPST(1), BETA(1), THICK(1), MATP(1),       00002600
                     2          DEN(1)                                             00002700
0015                  DIMENSION EVAL(9,2),XG(4,4),NOD5M(4),XC(1),YC(1)              00002900
                    C                                                               00003000
0016                  EQUIVALENCE (NPAR(2),NUME),(NPAR(16),NUMMAT),(NPAR(15),MODEL) 00003100
                     1          ,(NPAR(10),NINT),(NPAR(1),NPAR1),(NPAR(7),MXNODS)  00003200
                     2          ,(NPAR(5),ITYP2D),(NPAR(3),INDNL)                  00003300
                    C                                                               00003400
0017                  DATA XG /     0.,              0.,              0.,          0., 00003500
                     1 -.5773502691896, .5773502691896,              0.,          0., 00003600
                     2 -.7745966692415, .0000000000000, .7745966692415,          0., 00003700
                     3 -.8611363115941,-.3399810435849, .3399810435849, .8611363115941/ 00003800
                    C                                                               00003900
0018                  DATA EVAL / 1.0,-1.0,-1.0, 1.0, 0.0,-1.0, 0.0, 1.0, 0.0,      00004000
                     1            1.0, 1.0,-1.0,-1.0, 1.0, 0.0,-1.0, 0.0, 0.0/     00004100
0019                  DATA ANODE/4HNODE/                                            00004200
                    C                                                               00004300
                    C     ** NOTE ** DURING TIME INTEGRATION Y=DISP, Z=VEL, HT=R     00004400
                    C
                    C     COMPUTE FINAL ELEMENT STRESSES
                    C
0020                  DO 500 N=1,NUMEL
                    C
0021                  CALL CANDDE (B,H,N,IX,NUMEL,SIG,EPS,CC)
0022                  DO 100 J=1,4
0023                  SIG(N,J)=STRESS(J)
0024                  EPS(N,J)=STRAIN(J)
```

```
0025              100 CONTINUE
                  C
0026                  J=IX(N,2)
0027                  K=IX(N,3)
0028                  L=IX(N,4)
0029                  IF(K.NE.L) GO TO 200
0030                  I=IX(N,1)
0031                  XC(N)=(R(I)+R(J)+R(K))*0.3333
0032                  YC(N)=(Z(I)+Z(J)+Z(K))*0.3333
0033                  GO TO 300
0034              200 XC(N)=(R(J)+R(L))*0.5
0035                  YC(N)=(Z(J)+Z(L))*0.5
                  C
                  C     CALCULATE PRINCIPAL STRESSES
                  C
0036              300 CB=(SIG(N,1)+SIG(N,2))*0.5
0037                  BC=(SIG(N,1)-SIG(N,2))*0.5
0038                  CR=SQRT(BC**2+ IG(N,4)**2)
0039                  SIG(N,5)=CB+CR
0040                  SIG(N,6)=CB-CR
0041                  IF((BC.EQ.0.0).AND.(SIG(N,4).EQ.0.0)) GO TO 400
0042                  ANGLE=ATAN2(SIG(N,4),BC)*0.5
0043                  SIG(N,7)=57.296*ANGLE
0044                  GO TO 500
0045              400 SIG(N,7)=0.0
0046              500 CONTINUE
                  C
0047                  RETURN
0048                  END
```

COMMON BLOCK /SS / MAP SIZE 23674

SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION
ALFA	0	BETA	4	DELT	8	MBAND	C	N	10
NEQ	14	NNN	18	NP	1C	NT	20	NUMP	24
NUMEL	28	NUMAT	2C	TT	30	VOL	34	DUMMY	38
A	4C	SIG	135AC	B	15104	BO	1599C	CC	16234
C	179A4	CH	179C8	CV	17E14	EPS	18260	H	19200
IX	1EFC0	KODE	20348	MASS	20794	P	20BE0	R	20E38
Z	21284	S	216D0	T	21860	X0	21CAC	X1	22544
X2	22DDC								

COMMON BLOCK /EL / MAP SIZE 84

SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION
IND	0	ICOUNT	4	NPAR	8	NPAR1	8	NUME	C
INDNL	10	ITYP2D	18	MXNODS	20	NINT	2C	MODEL	40
NUMMAT	44	NUMEG	58	NEGL	5C	NEGNL	60	IMASS	64
IDAMP	68	ISTAT	6C	NDOFDM	70	KLIN	74	IEIG	78
IMASSN	7C	IDAMPN	80						

COMMON BLOCK /SOL / MAP SIZE 24

SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION
NUMNP	0	NWK	4	NWM	8	NWC	C	NUMEST	10
MIDEST	14	MAXEST	18	NSTE	1C	MA	20		

COMMON BLOCK /DIM / MAP SIZE 3C

SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION
N1	0	N2	4	N3	8	N4	C	N5	10
N6	14	N7	18	N8	1C	N9	20	N10	24
N11	28	N12	2C	N13	30	N14	34	N15	38

COMMON BLOCK /CONST / MAP SIZE 60

SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION
DT	0	DTA	4	A0	8	A1	C	A2	10
A3	14	A4	18	A5	1C	A6	20	A7	24
A8	28	A9	2C	A10	30	A11	34	A12	38
A13	3C	A14	40	A15	44	A16	48	A17	4C
A18	50	A19	54	A20	58	IOPE	5C		

COMMON BLOCK /DAMPNG / MAP SIZE 8

SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION
ADAMP	0	BDAMP	4						

COMMON BLOCK /VAR / MAP SIZE 24

SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION
NG	0	KPRI	4	MODEX	8	KSTEP	C	ITE	10
ITEMAX	14	IREF	18	IEQREF	1C	INOCMD	20		

COMMON BLOCK /MATMOD / MAP SIZE 44

SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION
D	0	IPT	40						

COMMON BLOCK /TODIM / MAP SIZE 14

SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION
BET	0	THIC	4	DE	8	IEL	C	NND5	10

COMMON BLOCK /DISDER / MAP SIZE 14

SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION
DISD	0								

COMMON BLOCK /ELPLS / MAP SIZE 54

SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION
STRESS	0	STRAIN	10	STRINC	20	BM	30	G	34
ALPH	38	CEE	3C	B11	40	B2	44	B3	48
U1	4C	U2	50						

SUBPROGRAMS CALLED

SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION
CANDDE	110	IG	114	SQRT	118	ATAN2	11C		

SCALAR MAP

SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION
ANODE	154	J	158	K	15C	L	160	I	164
CB	168	BC	16C	CR	170	ANGLE	174		

ARRAY MAP

SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION
Y	178	HT	17C	IELT	180	IPST	184	THICK	188
MATP	18C	DEN	190	EVAL	194	XG	1DC	NOD5M	21C
XC	22C	YC	230						

STATEMENT LABEL MAP

LOCATION	STA NUM	LABEL	LOCATION	STA NUM	LABEL	LOCATION	STA NUM	LABEL
000302	1		000302	20		00032A	21	
000338	22		00034A	23		000352	24	
00035A	25	100	000376	26		000382	27	
00038E	28		00039A	29		0003A8	30	
0003B4	31		0003E8	32		000418	33	
00041E	34	200	000446	35		00045A	36	300
00046A	37		00047A	38		0004C8	39	
0004D8	40		0004E8	41		00051E	42	
000540	43		000550	44		000556	45	400
000562	46	500	000596	47				

OPTIONS IN EFFECT NOTERM,ID,EBCDIC,SOURCE,NOLIST,NODECK,LOAD,MAP,NOTEST
OPTIONS IN EFFECT NAME = STRESS , LINECNT = 55
STATISTICS SOURCE STATEMENTS = 48,PROGRAM SIZE = 1438
STATISTICS NO DIAGNOSTICS GENERATED

```
0001                  SUBROUTINE  CANDDE (B,H,N,IX,NEL,SIG,EPS,CC)
                      C
                      C   SUBROUTINE FOR INCREMENTAL CORNER DISPLACEMENTS  AND INCREMENTAL
                      C   STRAINS DUE TO WAVE PROPAGATION
                      C
0002                        COMMON/MATARG/BLK(12),SHR(12),RO(12),ALPHA(12),CELL(12),B11(12),
                           *B22(12),B33(12),U11(12),U22(12)
0003                        COMMON/ELPLS/STRESS(4),STRAIN(4),STRINC(4),BM,G,ALPH,CEE,B1,
                           *B2,B3,U1,U2
0004                        DIMENSION B(1),H(1),IX(250,1),BB(8),EPS(250,1),C(4,4),CC(1),
                           *SIG(250,1)
                      C
                      C     READY FOR INCREMENTAL CORNER DISPLACEMENTS
                      C
0005                        DO 100 I=1,3
0006                        I2=I+1
0007                        J2=2*IX(N,I)
0008                        BB(I2-1)=B(J2-1)
0009                        BB(I2)=B(J2)
0010                  100   CONTINUE
0011                        J24=2*IX(N,4)
0012                        IF(J24.EQ.J2) GO TO 120
0013                        BB(7)=B(J24-1)
0014                        BB(8)=B(J24)
0015                        NF=8
0016                        GO TO 140
0017                  120   NF=6
                      C
                      C   NOW COMPUTE INCREMENTAL STRAINS
                      C
0018                  140   DO 160 I=1,3
0019                        STRINC(I)=0.0
0020                  160   CONTINUE
                      C
0021                        NO=N*24-32
0022                        DO 200 I=1,3
0023                        NO=NO+8
0024                        DO 180 J=1,NF
0025                        K=NO+J
0026                        STRINC(I)=STRINC(I)+H(K)*BB(J)
0027                  180   CONTINUE
0028                  200   CONTINUE
0029                        STRINC(4)=STRINC(3)
0030                        STRINC(3)=0.0
                      C
                      C     WE NOW GO TO ELASTIC- PLASTIC ANALYSIS (PREPARATION)
                      C
0031                        DO 220 I=1,4
0032                        STRESS(I)=SIG(N,I)
0033                        STRAIN(I)=EPS(N,I)
0034                  220   CONTINUE
0035                        MTYPE=IX(N,5)
0036                        BM=BLK(MTYPE)
```

```
0037                   G =SHR(MTYPE)
0038                   ALPH=ALPHA(MTYPE)
0039                   B1=B11(MTYPE)
0040                   B2=B22(MTYPE)
0041                   B3=B33(MTYPE)
0042                   U1=U11(MTYPE)
0043                   U2=U22(MTYPE)
                     C
0044                   CALL ELPAL(C,N)
                     C
                     C     CONSTITUTIVE LAWS FOR ELASTIC-PLASTIC ANALYSIS IS OBTAINED
                     C     FROM ""NONSAP"" AND INTRODUCED IN ""ELPAL""
                     C
                     C     STORE (C) MATRIX
0045                   NO=6*(N-1)
0046                   CC(NO+1)=C(1,1)
0047                   CC(NO+2)=C(1,2)
0048                   CC(NO+3)=C(1,4)
0049                   CC(NO+4)=C(2,2)
0050                   CC(NO+5)=C(2,4)
0051                   CC(NO+6)=C(4,4)
                     C
0052                   RETURN
0053                   END
```

COMMON BLOCK /MATARG / MAP SIZE 1E0

SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION
BLK	0	SHR	30	RO	60	ALPHA	90	CELL	C0
B11	F0	B22	120	B33	150	U11	180	U22	1B0

COMMON BLOCK /ELPLS / MAP SIZE 54

SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION
STRESS	0	STRAIN	10	STRINC	20	BM	30	G	34
ALPH	38	CEE	3C	B1	40	B2	44	B3	48
U1	4C	U2	50						

SUBPROGRAMS CALLED

SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION
ELPAL	C0								

SCALAR MAP

SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION
I	CC	I2	D0	J2	D4	N	D8	J24	DC
NF	E0	NO	E4	J	E8	K	EC	MTYPE	F0
NEL	F4								

ARRAY MAP

SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION
B	F8	H	FC	IX	100	BB	104	EPS	124
C	128	CC	168	SIG	16C				

STATEMENT LABEL MAP

LOCATION	STA NUM	LABEL	LOCATION	STA NUM	LABEL	LOCATION	STA NUM	LABEL
00026E	1		00026E	5		00027E	6	
00028A	7		00029E	8		0002BE	9	
0002C6	10	100	0002DE	11		0002F6	12	
000304	13		00031C	14		000324	15	
00032C	16		000332	17	120	00033A	18	140
000346	19		00034E	20	160	000366	21	
000376	22		000382	23		00038E	24	
00039A	25		0003A6	26		0003C6	27	180
0003DE	28	200	0003F6	29		000402	30	
00040A	31		000426	32		000436	33	
000446	34	220	000466	35		00047A	36	
000496	37		00049E	38		0004A6	39	
0004AE	40		0004B6	41		0004BE	42	
0004C6	43		0004CE	44		0004DC	45	
0004EE	46		000502	47		00050A	48	
000512	49		00051A	50		000522	51	
00052A	52							

OPTIONS IN EFFECT NOTERM,ID,EBCDIC,SOURCE,NOLIST,NODECK,LOAD,MAP,NOTEST
OPTIONS IN EFFECT NAME = CANDDE , LINECNT = 55
STATISTICS SOURCE STATEMENTS = 53,PROGRAM SIZE = 1330
STATISTICS NO DIAGNOSTICS GENERATED

```
0001                  SUBROUTINE ELPAL(COEF,NE)
0002                  DIMENSION COEF(4,4),DS(4),DE(4),STROLD(4),STRSIN(4)
0003                  COMMON/ELPLS/STRESS(4),STRAIN(4),STRINC(4),B0,G,ALPH,CEE,B1,
                     *B2,B3,U1,U2
0004                  IREG=1
0005                  DO 1 I=1,4
0006                  STROLD(I)=STRESS(I)
0007                1 CONTINUE
0008                  CALL ELFUN(B,G,B1,B2,B3,U1,U2,STRAIN(I),NE,B0)
0009                  CALL ELAST(B,G,COEFF(1,1))
0010                  CALL COMPUT(COEFF(1,1),STRINC(1),STRESS(1),STRSIN(1))
0011                  CALL TSTYLD(IREG,STRESS(1),1,ALPHA,CEE)
0012                  IF(IREG.EQ.1) GO TO 30
0013                  DO 20 I=1,4
0014                  STRESS(I)=STRESS(I)-STRSIN(I)
0015               20 CONTINUE
0016                  CALL YLDFUN(STRESS(1),COEFF(1,1),B,G,ALPHA,CEE,IREG)
0017                  CALL COMPUT(COEFF(1,1),STRINC(1),STRESS(1),STRSIN(1))
                   C
                   C     THIS CALL TO YLDFUN ADJUSTS THE DEVIATORS SO THAT STRESS
                   C     EXACTLY SATISFIES THE YIELD CONDITION
                   C
0018                  CALL TSTYLD(IREG,STRESS(1),2,ALPHA,CEE)
0019                  IF(IREG.EQ.1)CALL ELAST(B,G,COEFF(1,1))
0020               30 CONTINUE
0021                  DO 40 I=1,4
0022                  DS(I)=STRESS(I)-STROLD(I)
0023               40 CONTINUE
                   C
0024                  DP=(DS(1)+DS(2)+DS(3))*.3333
                   C
0025                  DO 50 I=1,3
0026                  DE(I)=(0.3333*DP/B)+(0.5* (DS(I)-DP)/G)
0027               50 CONTINUE
0028                  STRAIN(4)=STRAIN(4)+(DS(4)/G)
0029                  RETURN
0030                  END
```

COMMON BLOCK /ELPLS / MAP SIZE 54

SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION
STRESS	0	STRAIN	10	STRINC	20	B0	30	G	34
ALPH	38	CEE	3C	B1	40	B2	44	B3	48
U1	4C	U2	50						

SUBPROGRAMS CALLED

SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION
ELFUN	B4	ELAST	B8	COEFF	BC	COMPUT	C0	TSTYLD	C4
YLDFUN	C8								

SCALAR MAP

SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION
IREG	198	I	19C	B	1A0	NE	1A4	ALPHA	1A8
DP	1AC								

ARRAY MAP

SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION
COEF	1B0	DS	1B4	DE	1C4	STROLD	1D4	STRSIN	1E4

STATEMENT LABEL MAP

LOCATION	STA NUM	LABEL	LOCATION	STA NUM	LABEL	LOCATION	STA NUM	LABEL
000298	1		000298	4		0002A0	5	
0002B0	6		0002B8	7	1	0002D4	8	
0002F6	9		000316	10		000352	11	
000368	12		000376	13		000386	14	
000392	15	20	0003AE	16		0003DA	17	
000416	18		00042C	19		00045A	21	30
00046A	22		000476	23	40	000492	24	
0004A6	25		0004B2	26		0004D8	27	50
0004F0	28		000504	29				

```
 *OPTIONS IN EFFECT*  NOTERM,ID,EBCDIC,SOURCE,NOLIST,NODECK,LOAD,MAP,NOTEST
 *OPTIONS IN EFFECT*  NAME = ELPAL    , LINECNT =       55
 *STATISTICS*    SOURCE STATEMENTS =       30,PROGRAM SIZE =     1292
 *STATISTICS*  NO DIAGNOSTICS GENERATED
```

```
0001                  SUBROUTINE YLDFUN(S,COEFF,B,ALPHA,CEE,IREG)
0002                  DIMENSION S(4),COEFF(4,4),F(4)
0003                  DUM=0.1667*((S(1)-S(2))**2+(S(2)-S(3))**2+(S(3)-S(1))**2)+S(4)**2
0004                  DUM=SQRT(ABS(DUM))
0005                  F(1)=(0.1667*(2.0*S(1)-S(2)-S(3))/DUM)-ALPHA
0006                  F(2)=(0.1667*(2.0*S(2)-S(1)-S(3))/DUM)-ALPHA
0007                  F(3)=(0.1667*(2.0*S(3)-S(1)-S(2))/DUM)-ALPHA
0008                  F(4)=S(4)/DUM
0009                  CALL PLAST(F(1),B,G,COEFF(1,1))
0010                  RETURN
0011                  END
```

SUBPROGRAMS CALLED

SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION
PLAST	90	SQRT	94						

SCALAR MAP

SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION
DUM	B0	ALPHA	B4	B	B8	G	BC	CEE	C0
IREG	C4								

ARRAY MAP

SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION
S	C8	COEFF	CC	F	D0				

STATEMENT LABEL MAP

LOCATION	STA NUM	LABEL	LOCATION	STA NUM	LABEL	LOCATION	STA NUM	LABEL
0001B2	1		0001B2	3		0001E8	4	
000204	5		000224	6		000244	7	
000264	8		000270	9		000296	10	

```
 *OPTIONS IN EFFECT*  NOTERM,ID,EBCDIC,SOURCE,NOLIST,NODECK,LOAD,MAP,NOTEST
 *OPTIONS IN EFFECT*  NAME = YLDFUN  , LINECNT =       55
 *STATISTICS*    SOURCE STATEMENTS =       11,PROGRAM SIZE =      670
 *STATISTICS*  NO DIAGNOSTICS GENERATED
```

```
0001                  SUBROUTINE COMPUT(COEF,STRINC,STRESS,STRSIN)
0002                  DIMENSION COEFF(4,4),STRINC(4),STRESS(4),STRSIN(4)
0003               35 DO 50 I=1,4
0004                  DUM=0.
0005                  DO 40 J =1,4
0006                  DUM =COEFF(I,J)*STRINC(J)+DUM
0007               40 CONTINUE
0008                  STRSIN(I)=DUM
0009               50 CONTINUE
                    C
0010                  DO 60 I=1,4
0011                  STRESS(I)=STRESS(I)+STRSIN(I)
0012               60 CONTINUE
0013                  RETURN
0014                  END
```

SCALAR MAP

SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION
I	9C	DUM	A0	J	A4	COEF	A8		

ARRAY MAP

SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION
COEFF	AC	STRINC	EC	STRESS	F0	STRSIN	F4		

STATEMENT LABEL MAP

LOCATION	STA NUM	LABEL	LOCATION	STA NUM	LABEL	LOCATION	STA NUM	LABEL
0001A0	1		0001A0	3	35	0001B4	4	
0001BC	5		0001CE	6		0001DE	7	40
0001FA	8		000202	9	50	00021E	10	
000236	11		000242	12	60	00025E	13	

OPTIONS IN EFFECT NOTERM,ID,EBCDIC,SOURCE,NOLIST,NODECK,LOAD,MAP,NOTEST
OPTIONS IN EFFECT NAME = COMPUT , LINECNT = 55
STATISTICS SOURCE STATEMENTS = 14,PROGRAM SIZE = 614
STATISTICS NO DIAGNOSTICS GENERATED

```
0001                  SUBROUTINE TSTYLD(IREG,STRESS,I,ALPHA,CEE)
0002                  DIMENSION STRESS(4)
                C
                C     THIS SUBROUTINE TESTS IF THE STRESSES HAS BEEN PROPERLY
                C     COMPUTED BY SR YLDFUN
                C
0003                  S1=STRESS(1)
0004                  S2=STRESS(2)
0005                  S3=STRESS(3)
0006                  S4=STRESS(4)
                C
                C
0007                  DUM1=0.1667*((S1-S2)**3+(S2-S3)**2+(S3-S1)**2)+S4**2
0008                  DUM2=S1+S2+S3
0009                  DUM1=SQRT(DUM)
0010                  P=ABS(DUM2/3.)
0011               40 CONTINUE
0012                  IF(I.EQ.2) GO TO 50
0013                  F=DUM1-0.95*(ALPHA*DUM2+CEE)
0014                  IF(F.LE.0) GO TO 10
                C
                C     PLASTIC DEFORMATION
0015                  IREG=2
0016                  IF(I.EQ.1) RETURN
                C
                C     STRESS HAS BEEN PROPERLY COMPUTED BY YLDFUN
0017               20 DMAX=DD
0018                  IF(ALPHA.EQ.0) GO TO 50
0019                  DD=-0.95*CEE/ALPHA
0020                  IF(DUM2.LT.DD) GO TO 50
0021                  DMAX=DD
0022                  IREG=1
0023               50 FACTOR=0.
0024                  IF(DUM1.NE.0.) FACTOR=ABS(ALPHA*DMAX+CEE)/DUM1
0025                  IF(FACTOR.EQ.0.) IREG=1
0026                  DO 30 J=1,3
0027               30 STRESS(J)=(STRESS(J)-DUM2/3.)*FACTOR+DMAX/3.
0028                  STRESS(4)=STRESS(4)*FACTOR
0029                  RETURN
0030               10 IREG=1
0031                  RETURN
0032                  END
```

SUBPROGRAMS CALLED

SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION
SQRT	C0								

SCALAR MAP

SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION
S1	D0	S2	D4	S3	D8	S4	DC	DUM1	E0
DUM2	E4	DUM	E8	P	EC	I	F0	F	F4
ALPHA	F8	CEE	FC	IREG	100	DMAX	104	DD	108
FACTOR	10C	J	110						

ARRAY MAP

SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION
STRESS	114								

STATEMENT LABEL MAP

LOCATION	STA NUM	LABEL	LOCATION	STA NUM	LABEL	LOCATION	STA NUM	LABEL
0001EA	1		0001EA	3		0001F6	4	
0001FE	5		000206	6		00020E	7	
000244	8		000254	9		000266	10	
000274	12	40	000282	13		00029C	14	
0002AA	15		0002B2	16		0002C8	17	20
0002D0	18		0002DE	19		0002F0	20	
0002FE	21		000306	22		00030E	23	50
000316	24		00033A	25		000350	26	
000360	27	30	000396	28		0003A6	29	
0003AE	30	10	0003B6	31				

OPTIONS IN EFFECT NOTERM,ID,EBCDIC,SOURCE,NOLIST,NODECK,LOAD,MAP,NOTEST
OPTIONS IN EFFECT NAME = TSTYLD , LINECNT = 55
STATISTICS SOURCE STATEMENTS = 32,PROGRAM SIZE = 958
STATISTICS NO DIAGNOSTICS GENERATED

```
0001                  SUBROUTINE PLAST(F,B,G,C)
0002                  DIMENSION F(4),C(4,4)
                C
                C     THIS SUBROUTINE COMPUTES LAME'S PARAMETER AND PLASTICITY MATRIX
0003                  F1=F(1)
0004                  F2=F(2)
0005                  F3=F(3)
0006                  F4=F(4)
                C
                C     LAME'S PARAMETER
0007                  X=B-0.667*G
                C
0008                  FF=F1+F2+F3
0009                  S=F1**2+F2**2+F3**2+2.0*F4**2
0010                  D=X*FF**2+2.0*G*S
0011                  A1=X*FF+2.0*G*F1
0012                  A2=X*FF+2.0*G*F2
0013                  A3=X*FF+2.0*G*F3
                C
                C     C(1,1)=(X+2.*G)-(A1*A2/D)
0014                  C(1,2)=X-(A1*A2/D)
0015                  C(1,3)=X-(A1*A3/D)
0016                  C(1,4)=-2.*G*F4*A1/D
0017                  C(2,2)=X+2.0*G-(A2**2/D)
0018                  C(2,3)=X-(A2*A3/D)
0019                  C(2,4)= -2.*G*F4*A2/D
0020                  C(3,3)=X+2.*G-(A3*A3/D)
0021                  C(3,4)=-2.0*G*F4*A3/D
0022                  C(4,4)=G-(4.*G**2*F4**2/D)
0023                  C(2,1)=C(1,2)
0024                  C(3,1)=C(1,3)
0025                  C(3,2)=C(2,3)
0026                  C(4,1)=C(1,4)
0027                  C(4,2)=C(2,4)
0028                  C(4,3)=C(3,4)
0029               20 RETURN
0030                  END
```

SCALAR MAP

SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION
F1	90	F2	94	F3	98	F4	9C	X	A0
B	A4	G	A8	FF	AC	S	B0	D	B4
A1	B8	A2	BC	A3	C0				

ARRAY MAP

SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION
F	C4	C	C8						

STATEMENT LABEL MAP

LOCATION	STA NUM	LABEL	LOCATION	STA NUM	LABEL	LOCATION	STA NUM	LABEL
00017A	1		00017A	3		000186	4	
00018E	5		000196	6		00019E	7	
0001B0	8		0001C0	9		0001E6	10	
000202	11		00021C	12		000236	13	
000250	14		00026A	15		000280	16	
00029A	17		0002B8	18		0002CE	19	
0002E8	20		000308	21		000322	22	
000342	23		00034A	24		000352	25	
00035A	26		000362	27		00036A	28	
000372	29	20						

```
 *OPTIONS IN EFFECT*  NOTERM,ID,EBCDIC,SOURCE,NOLIST,NODECK,LOAD,MAP,NOTEST
 *OPTIONS IN EFFECT*  NAME = PLAST   , LINECNT =       55
 *STATISTICS*    SOURCE STATEMENTS =       30,PROGRAM SIZE =      890
 *STATISTICS*  NO DIAGNOSTICS GENERATED
```

```
0001                  SUBROUTINE ELFUN(B,G,B1,B2,B3,U1,U2,STRAIN,NE,B0)
0002                  DIMENSION UELMAX(250),STRAIN(4)
0003                  EQUIVALENCE(DUM,UCRNT)
0004                  E1=STRAIN(1)
0005                  E2=STRAIN(2)
0006                  E3=STRAIN(3)
0007                  E4=STRAIN(4)
0008                  UCRNT=ABS((1.+E1+E2+ E1*E2-E4*E4)*(1.E3)-1.)
                C
0009                  UMAX=UELMAX(NE)
0010                  IF(UCRNT.GE.UMAX)GO TO 10
                C     UNLOADING AND RELOADING
0011                  BP=B2+(B1-B2)*UMAX/U1
0012                  IF(BP.GE.B1) BP=B1
0013                  B=BP+(B1-BP)*DUM/U1
0014                  GO TO 100
                C
0015                10 B=B1-(B1-B2)*EXP(-DUM/U2)
0016                  UELMAX(NE)=UCRNT
0017               100 CONTINUE
0018               300 CONTINUE
0019                  B=B0
0020                  RETURN
0021                  END
```

SUBPROGRAMS CALLED

SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION
EXP	A8								

EQUIVALENCE DATA MAP

SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION
DUM	B8	UCRNT	B8						

SCALAR MAP

SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION
E1	BC	E2	C0	E3	C4	E4	C8	UMAX	CC
NE	D0	BP	D4	B2	D8	B1	DC	U1	E0
B	E4	U2	E8	B0	EC	G	F0	B3	F4

ARRAY MAP

SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION
UELMAX	F8	STRAIN	4E0						

STATEMENT LABEL MAP

LOCATION	STA NUM	LABEL	LOCATION	STA NUM	LABEL	LOCATION	STA NUM	LABEL
000618	1		000618	4		000624	5	
00062C	6		000634	7		00063C	8	
00066A	9		00067A	10		000688	11	
0006A0	12		0006B6	13		0006CE	14	
0006D4	15	10	00070A	16		00071A	17	100
00071A	19	300	000722	20				

OPTIONS IN EFFECT NOTERM,ID,EBCDIC,SOURCE,NOLIST,NODECK,LOAD,MAP,NOTEST
OPTIONS IN EFFECT NAME = ELFUN , LINECNT = 55
STATISTICS SOURCE STATEMENTS = 21,PROGRAM SIZE = 1834
STATISTICS NO DIAGNOSTICS GENERATED

```
0001                  SUBROUTINE ELAST(B,G,C)
0002                  DIMENSION C(4,4)
                    C
                    C     FORMATION OF ELASTICITY MATRIX
0003                  C(1,1)=B+1.333*G
0004                  C(1,2)=B-0.6666*G
0005                  C(4,4)=G
0006                  C(2,1)=C(1,2)
0007                  C(1,3)=C(1,2)
0008                  C(1,4)=0.
0009                  C(2,4)=0.
0010                  C(3,1)=C(1,2)
0011                  C(3,2)=C(1,2)
0012                  C(3,3)=C(1,1)
0013                  C(3,4)=0.
0014                  C(4,1)=0.
0015                  C(4,2)=0.
0016                  C(4,3)=0.
0017                  RETURN
0018                  END
```

SCALAR MAP

SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION
B	90	G	94						

ARRAY MAP

SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION
C	98								

STATEMENT LABEL MAP

LOCATION	STA NUM	LABEL	LOCATION	STA NUM	LABEL	LOCATION	STA NUM	LABEL
00013A	1		00013A	3		00014E	4	
000160	5		000168	6		000170	7	
000178	8		000180	9		000188	10	
000190	11		000198	12		0001A0	13	
0001A8	14		0001B0	15		0001B8	16	
0001C0	17							

OPTIONS IN EFFECT NOTERM,ID,EBCDIC,SOURCE,NOLIST,NODECK,LOAD,MAP,NOTEST
OPTIONS IN EFFECT NAME = ELAST , LINECNT = 55
STATISTICS SOURCE STATEMENTS = 18,PROGRAM SIZE = 456
STATISTICS NO DIAGNOSTICS GENERATED

```
0001                  SUBROUTINE SYMSOL (KKK)
0002                  REAL MASS
0003                  COMMON  ALFA,BETA,DELT,MBAND,N,NEQ,NNN,NP,NT,NUMNP,NUMEL,NUMAT,
                     1 TT,VOL,DUMMY(5),A(550,36),SIG(250,7),B(550),B0(550),CC(1500),
                     2 C(3,3),CH(275),CV(275),EPS(250,4),H(6000),IX(250,5),KODE(275),
                     3 MASS(275),P(3,50),R(275),Z(275),S(10,10),T(275),X0(550),XI(550),
                     4 X2(550)
                C
0004                    GO TO (1000,2000),KKK
                C
                C  REDUCE MATRIX
                C
0005             1000  DO 280 N=1,NEQ
0006                   DO 260 L=2,MBAND
0007                   CONST= A(N,L)/A(N,1)
0008                   I=N+L-1
0009                   IF (NEQ-1) 260,240,240
0010               240 J=0
0011                   DO 250 K=L,MBAND
0012                   J=J+1
0013               250 A(I,J)=A(I,J)-CONST*A(N,K)
0014               260 A(N,L)=CONST
0015               280 CONTINUE
0016                   GO TO 500
                C
                C
                C   REDUCE VECTOR
                C
0017             2000  DO 290 N=1,NEQ
0018                   DO 285 L=2,MBAND
0019                   I=N+L-1
0020                   IF (NEQ-I)290,285,285
0021               285 B(I)=B(I)-A(N,L)*B(N)
0022               290 B(N)=B(N)/A(N,1)
                C
                C    BACK SUBSTITUTION
                C
0023                   N=NEQ
0024               300 N=N-1
0025                   IF(N) 350,500,350
0026               350 DO 400 K=2,MBAND
0027                   L=N+K-1
0028                   IF(NEQ-L) 400,370,370
0029               370 B(N)= B(N)-A(N,K)*B(L)
0030               400 CONTINUE
0031               500 RETURN
0032                   END
```

COMMON BLOCK / / MAP SIZE 23674

SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION
ALFA	0	BETA	4	DELT	8	MBAND	C	N	10
NEQ	14	NNN	18	NP	1C	NT	20	NUMNP	24
NUMEL	28	NUMAT	2C	TT	30	VOL	34	DUMMY	38
A	4C	SIG	135AC	B	15104	B0	1599C	CC	16234
C	179A4	CH	179C8	CV	17E14	EPS	18260	H	19200
IX	1EFC0	KODE	20348	MASS	20794	P	20BE0	R	20E38
Z	21284	S	216D0	T	21860	X0	21CAC	XI	22544
X2	22DDC								

SCALAR MAP

SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION
KKK	120	L	124	CONST	128	I	12C	J	130
K	134								

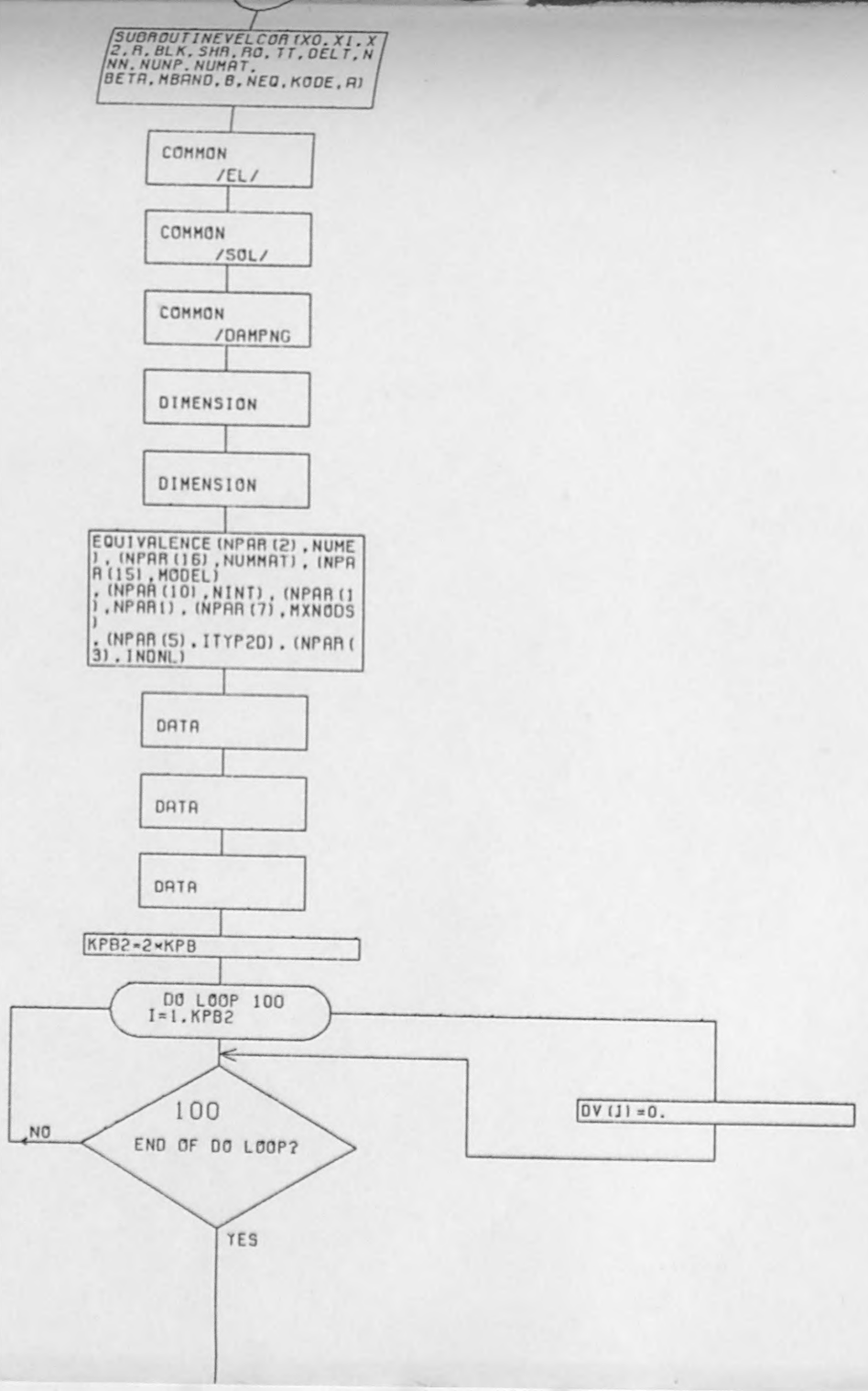
STATEMENT LABEL MAP

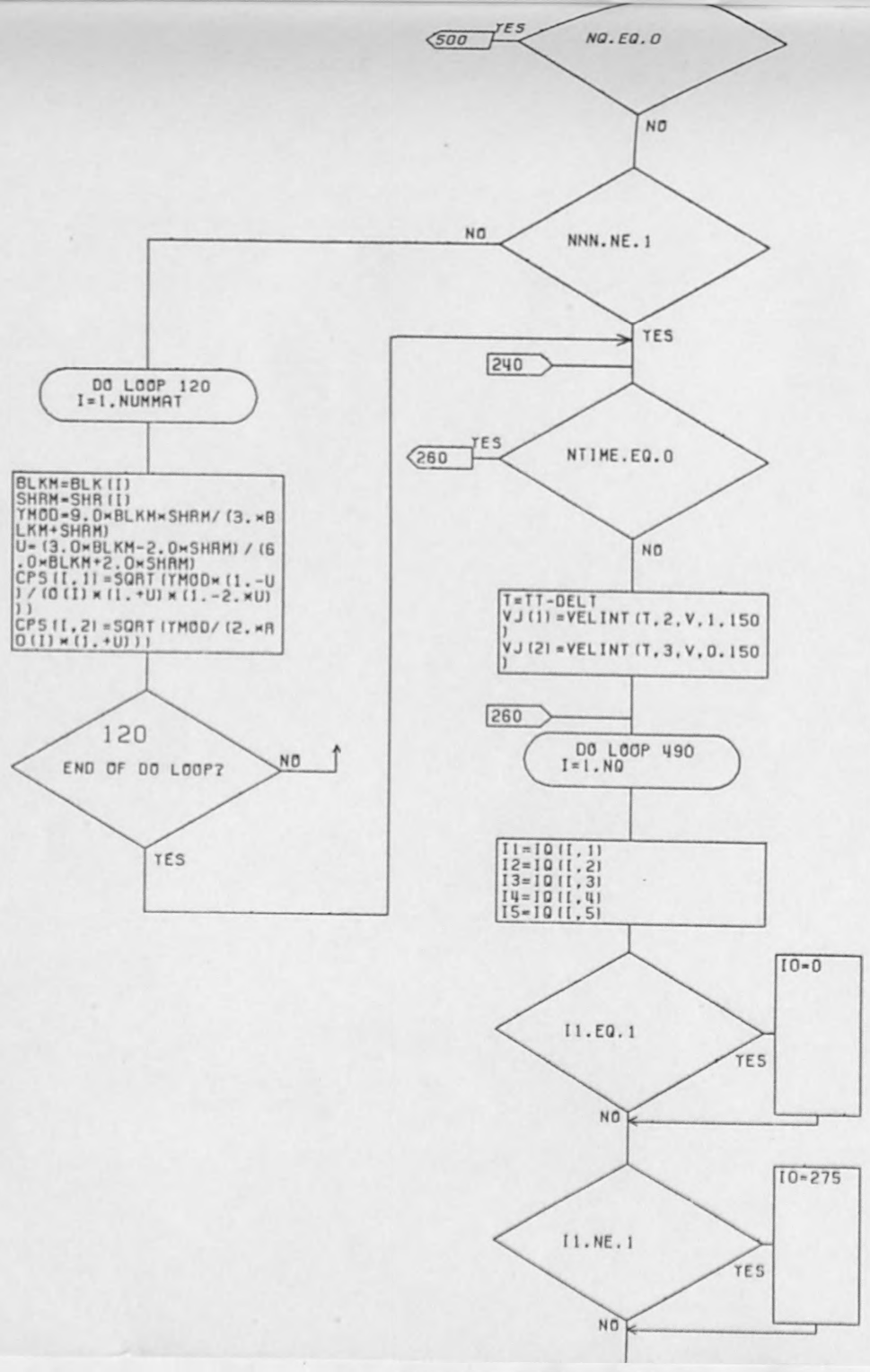
LOCATION	STA NUM	LABEL	LOCATION	STA NUM	LABEL	LOCATION	STA NUM	LABEL
0001DC	1		0001DC	4		000208	5	1000
00021C	6		000226	7		000232	8	
000246	9		000256	10	240	00025E	11	
000272	12		00027E	13	250	0002E0	14	260
000304	15	280	000324	16		00032A	17	2000
00033E	18		000348	19		00035C	20	
00036C	21	285	0003AA	22	290	0003D2	23	
0003DE	24	300	0003EA	25		0003F6	26	350
000402	27		000416	28		000426	29	370
000460	30	400	00047C	31	500			

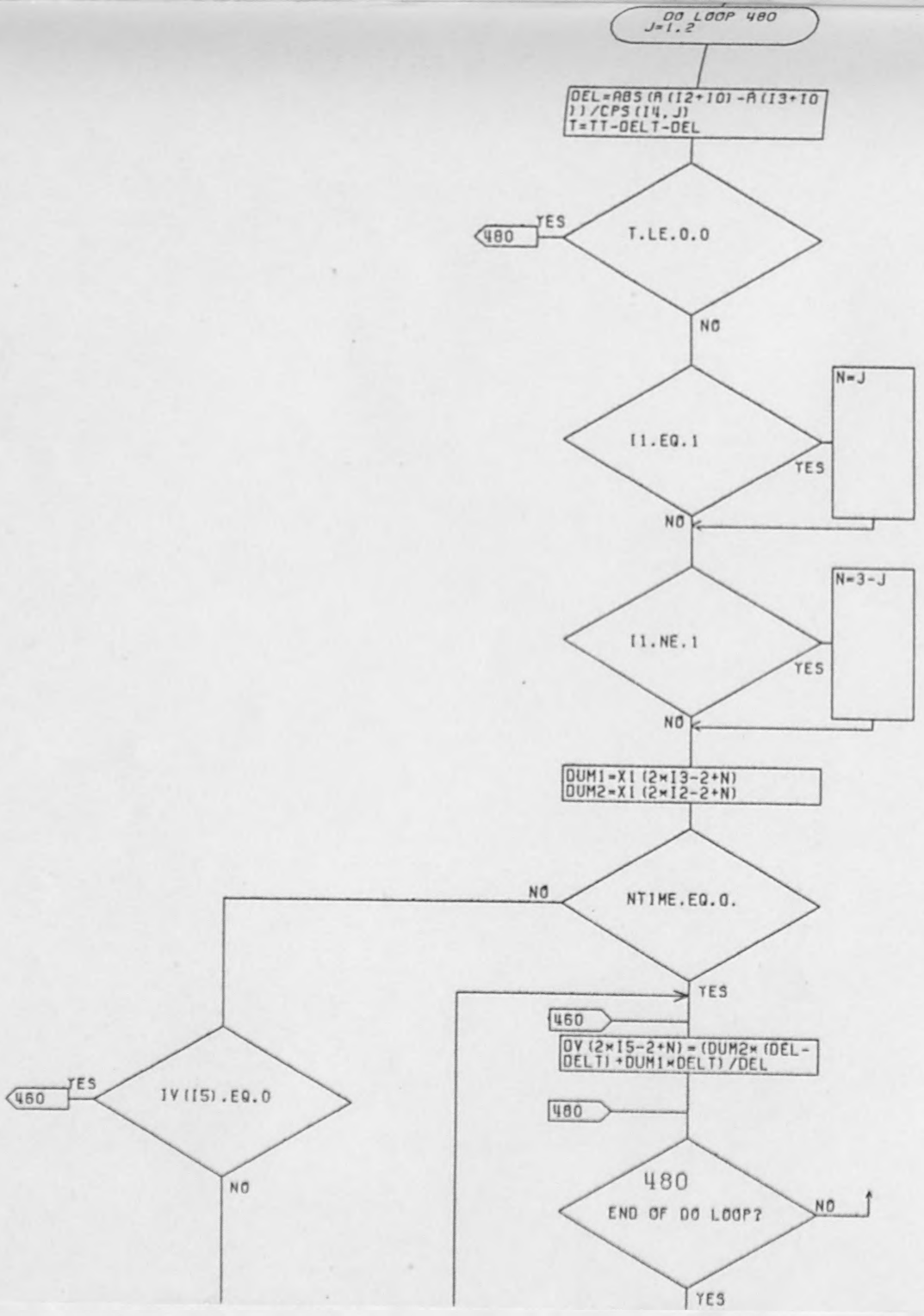
OPTIONS IN EFFECT NOTERM,ID,EBCDIC,SOURCE,NOLIST,NODECK,LOAD,MAP,NOTEST
OPTIONS IN EFFECT NAME = SYMSOL , LINECNT = 55
STATISTICS SOURCE STATEMENTS = 32,PROGRAM SIZE = 1156
STATISTICS NO DIAGNOSTICS GENERATED

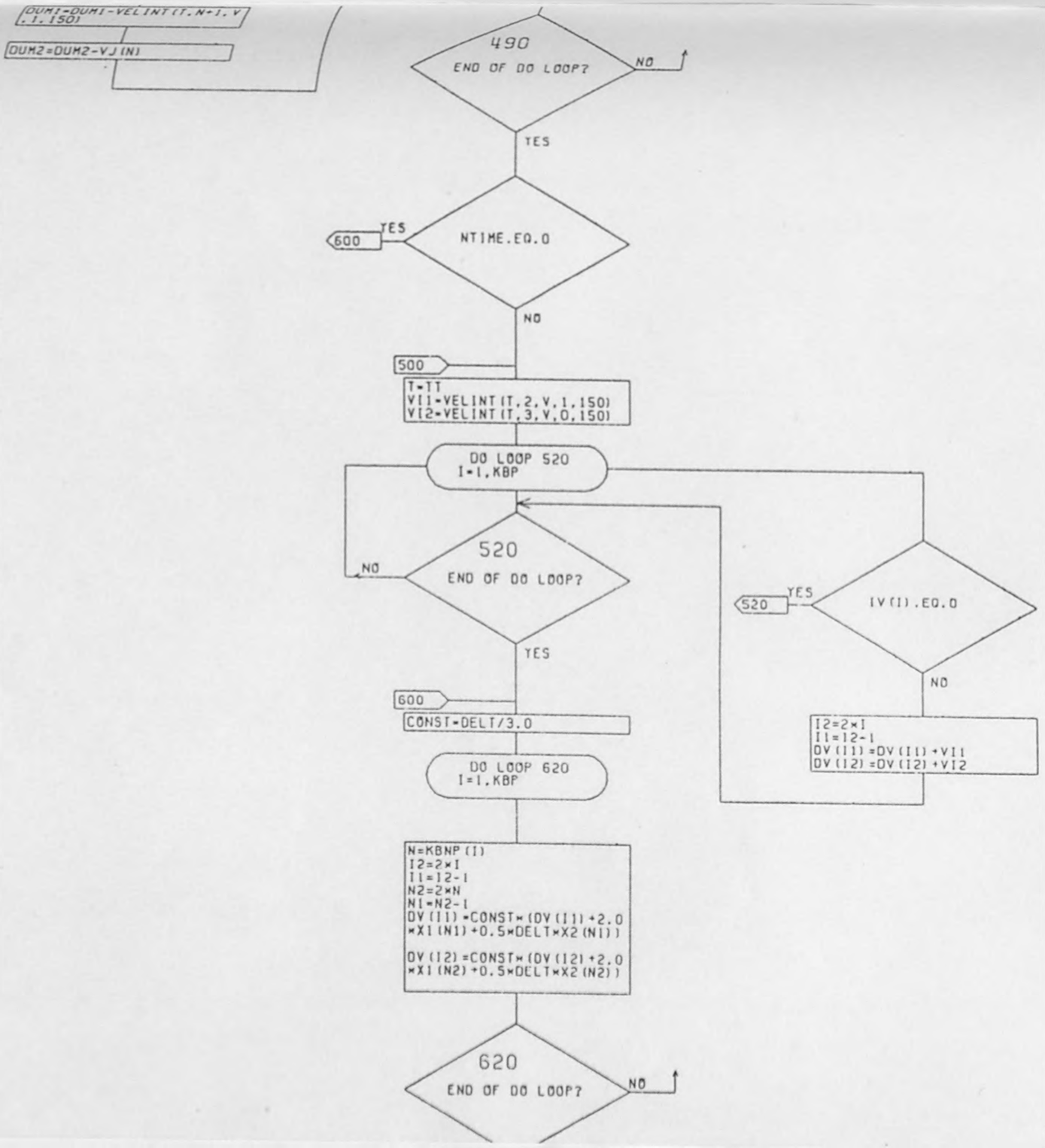
STATISTICS NO DIAGNOSTICS THIS STEP

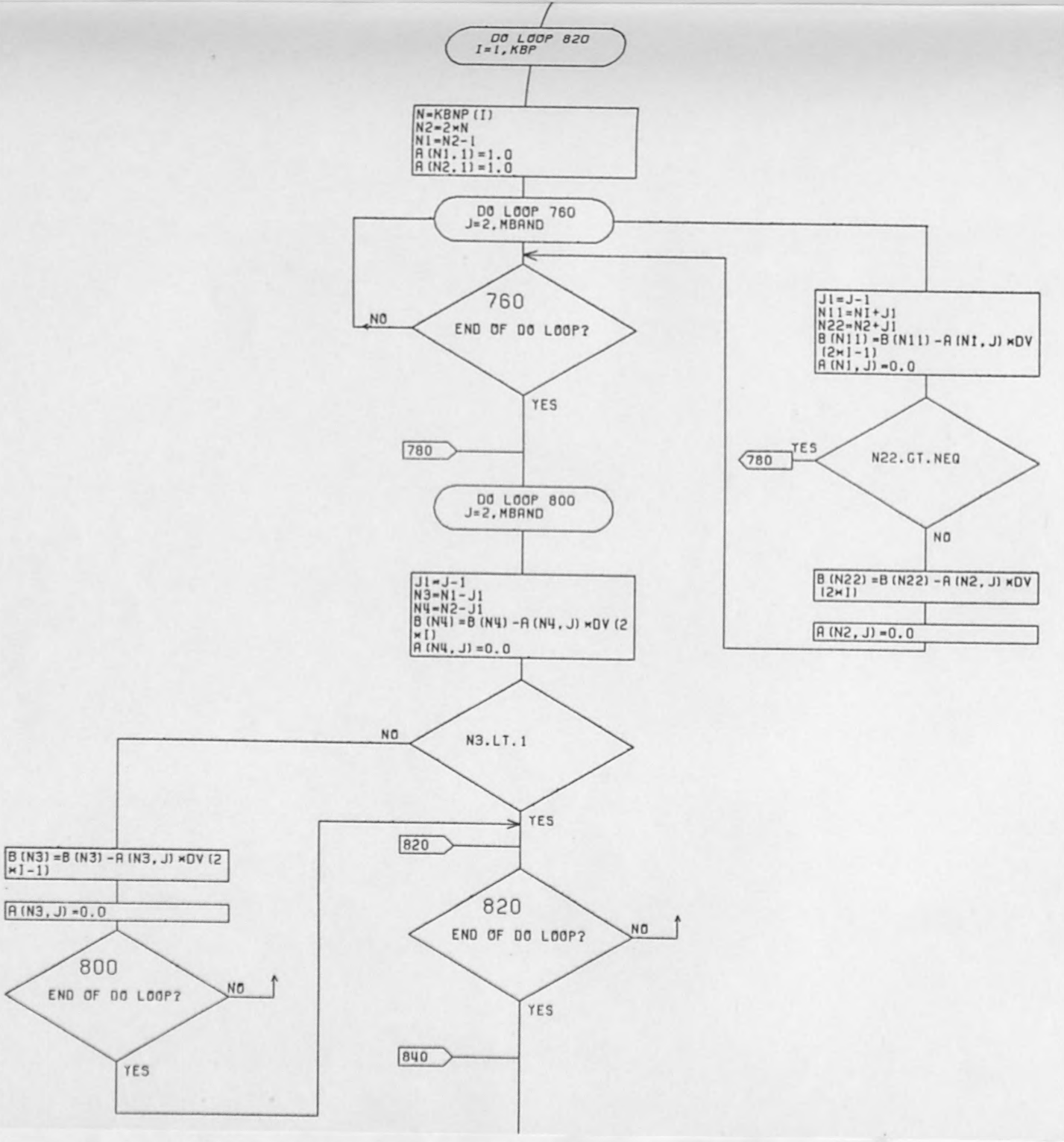
```
      C  SUBROUTINE FOR SUPPRESSING UNWANTED REFLECTED VELOCITY FROM THE
      C  BOUNDARY.(VELOCITY CORRECTION=VELCOR)
      C  PROGRAMMED BY S.N.MUKHERJEE.BBC,BADEN AUG,1977
      C UPDATED: MARCH,1978. SEPT,1980.
      C NODAL POINT DATA AND ELEMENT PROPERTIES ARE READ IN NONSAP.
      C ELEMENT STIFFNESS MATRIX AND ASSEMBLED GLOBAL STIFFNESS MATRIX,
      C MASS MATRIX ARE FORMED IN NONSAP AND USED IN VELCOR.
      C**************************************************************
      C TO USE VELCOR WITH ANY OTHER FINITE ELEMENT PROGRAM WHICH HAS
      C THE CAPBILITIES OF DYNAMIC ANALYSIS THE FOLLOWING IN PUT DATA ARE
      C REQUIRED UNLESS OR  OTHERWISE AVAILABLE.
      C  I = MATERIAL IDENTIFICATION NUMBER
      C  BLK(I) =BULK MODULUS.
      C  SHR(I) = SHEAR MODULUS
      C  ALPHA(I) = ANGLE OF FRICTION
      C  CELL(I)  = COHESION
      C  KBP = NUMBER OF NODAL POINTS WHERE  KINEMATIC INPUT PRESCRIBED
      C  FORMAT I3
      C  (KBP(I),I=1,KBP) =NODAL POINT NUMBER WITH KINEMATIC INPUT
      C  NQ =NUMBER OF NODAL POINTS TO BE QUIETED FORMAT(I3)
      C  A BLANK CARD FOR NQ=0
      C  IF AN INPUT IS GIVEN IN NQ THEN THE FOLLOWING CARDS MUST BE SUPPLIED.
      C
      C  ((IQ(I,J),J=1,5,I=NQ)      FORMAT (5I5)
      C  IQ(I,1) = TYPE OF BOUNDARY I=1 VERTICAL BOUNDARY, I=2 HORIZONTAL
      C  BOUNDARY
      C  IQ(1,2) = NODAL POINT NUMBER OF CORRECTOR
      C  IQ(1,3) = NODAL POINT NUMBER OF SENSOR
      C  IQ(1,4) = MATERIAL PROPERTY INDEX
      C  IQ(1,5) =SEQUENCE NUMBER OF THE POINT IN THE KBNP ARRAY.
      C**************************************************************
      C
0001        SUBROUTINE VELCOR(X0,X1,X2,R,BLK,SHR,RO,TT,DELT,NNN,NUMNP,NUMAT,
           1BETA,MBAND,B,NEQ,KODE,A)
0002        COMMON /EL/ IND,ICOUNT,NPAR(20),NUMEG,NEGL,NEGNL,IMASS,IDAMP,ISTAT00001000
           1           ,NDOFDM,KLIN,IEIG,IMASSN,IDAMPN                      00001100
0003        COMMON /SOL/ NWK,NWM,NWC,NUMEST,MIDEST,MAXEST,NSTE,MA           00001200
0004        COMMON /DIM/ N1,N2,N3,N4,N5,N6,N7,N8,N9,N10,N11,N12,N13,N14,N15 00001300
0005        COMMON /CONST/ DT,DTA,A0,A1,A2,A3,A4,A5,A6,A7,A8,A9,A10,A11     00001400
           1             ,A12,A13,A14,A15,A16,A17,A18,A19,A20,IOPE          00001500
0006        COMMON /DAMPNG/ ADAMP,BDAMP                                     00001600
0007        COMMON /VAR/ NG,KPRI,MODEX,KSTEP,ITE,ITEMAX,IREF,IEQREF,INOCMD  00001700
      C     COMMON /EM/ S(136),XM(16),B(4,16),NOD(8),NODM(8),RE(16),        00001800
      C    1            EDIS(16),AEM(204),XX(16)                          00001900
0008        COMMON /MATMOD/ STRESS(4),STRAIN(4),D(4,4),IPT,N                00002000
0009        COMMON /TODIM/ BET,THIC,DE,IEL,NND5                             00002100
0010        COMMON /DISDER/ DISD(5)                                         00002200
0011        COMMON/KINBC/KBP,KBNP(60),NTIME,NQ,IQ(60,5),DV(120),V(150,3),
           &IV(60)
      C     COMMON A(1)                                                     00002300
      C                                                                     00002400
0012        DIMENSION Y(1), Z(1), HT(1),                                    00002500
           1          IELT(1), IPST(1), BETA(1), THICK(1), MATP(1),        00002600
```

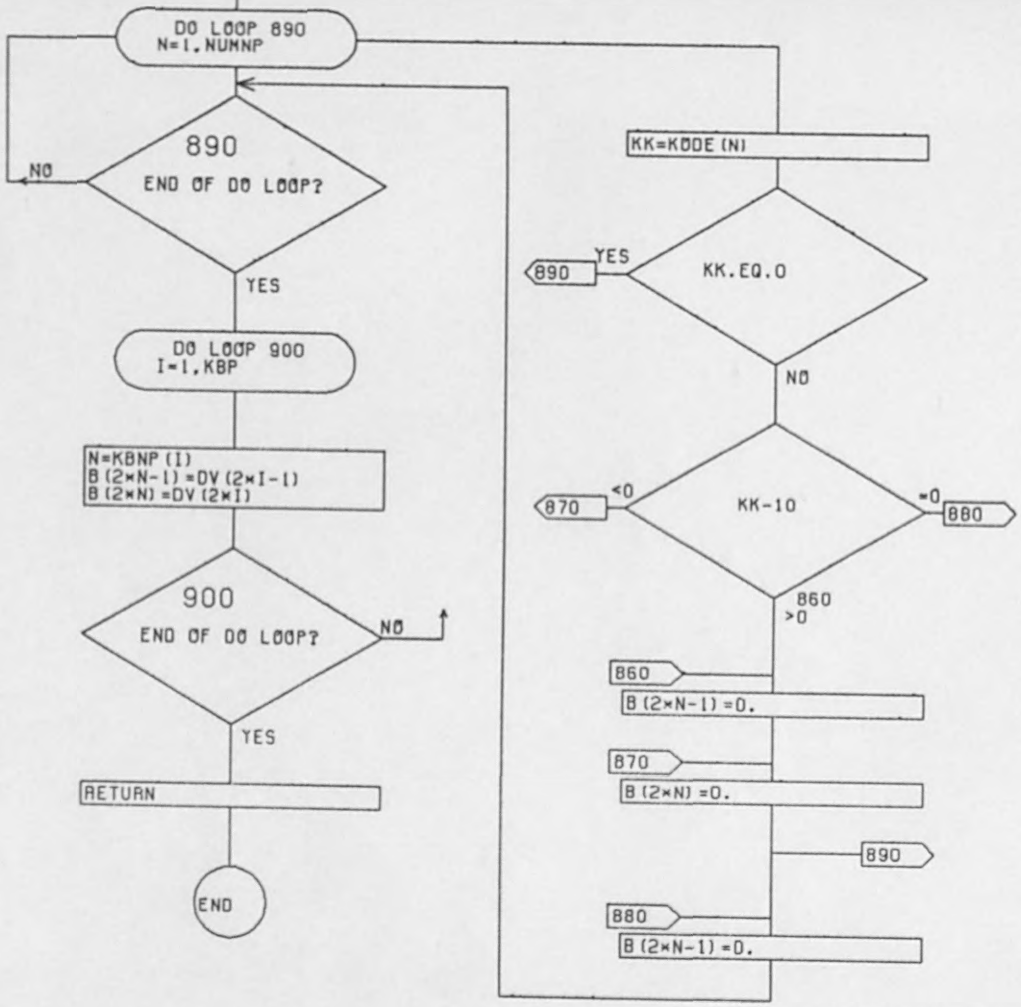

SUBROUTINEVELCOR(XO, X1, X2, R, BLK, SHR, RO, TT, DELT, NNN, NUNP, NUMAT, BETA, MBAND, B, NEQ, KODE, A)
COMMON /EL/
COMMON /SOL/
COMMON /DAMPNG
DIMENSION
DIMENSION
EQUIVALENCE (NPAR(2), NUME), (NPAR(16), NUMMAT), (NPAR(15), MODEL), (NPAR(10), NINT), (NPAR(1), NPAR1), (NPAR(7), MXNODS), (NPAR(5), ITYP2D), (NPAR(3), INONL)
DATA
DATA
DATA
KPB2=2*KPB
DO LOOP 100 I=1, KPB2
100
END OF DO LOOP?
NO
YES
DV(J)=0.

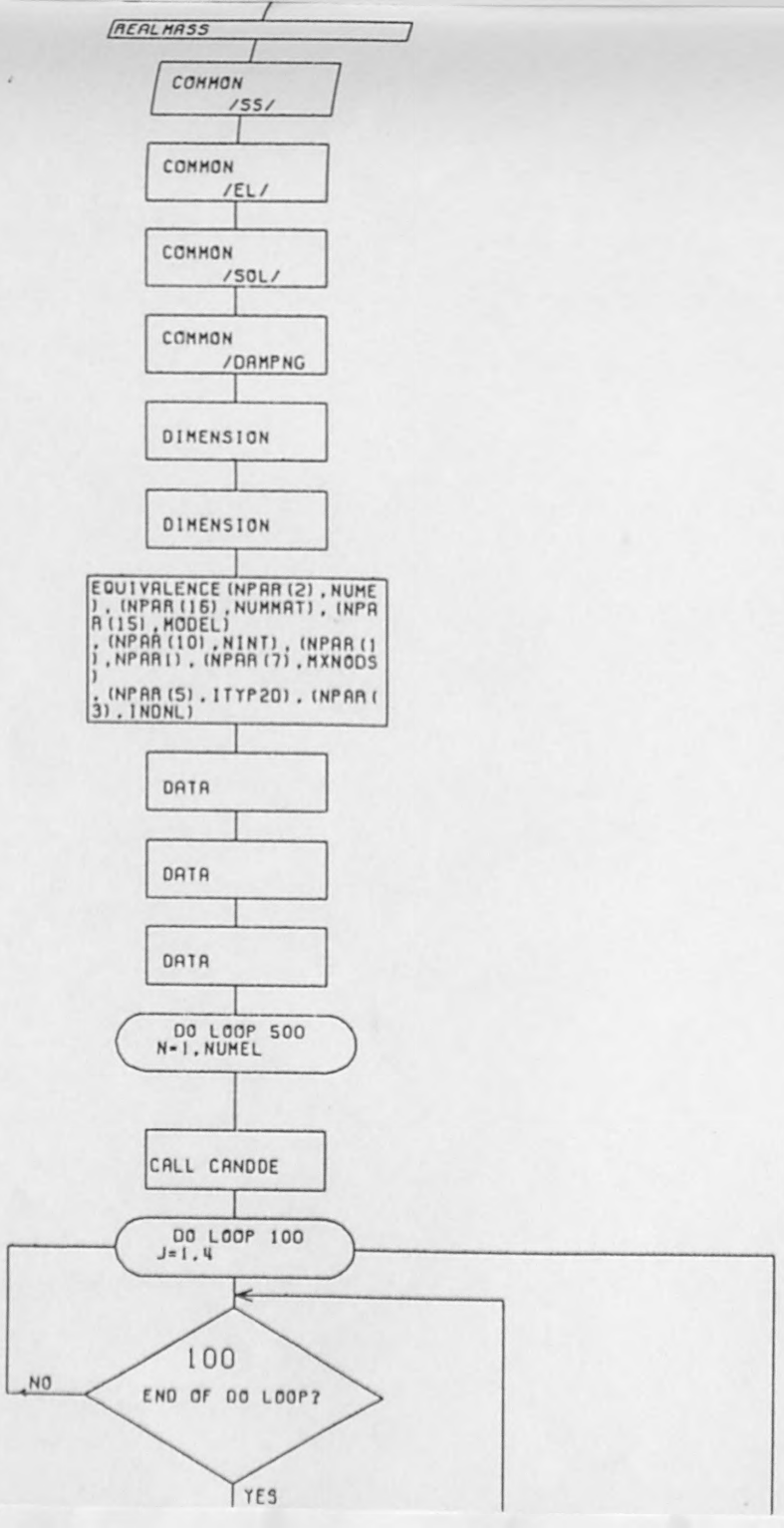

500
YES
NO.EQ.0
NO
NNN.NE.1
NO
YES
DO LOOP 120
I=1,NUMMAT
BLKM=BLK(I)
SHRM=SHR(I)
YMOD=9.0*BLKM*SHRM/(3.*BLKM+SHRM)
U=(3.0*BLKM-2.0*SHRM)/(6.0*BLKM+2.0*SHRM)
CPS(I,1)=SQRT(YMOD*(1.-U)/(D(I)*(1.+U)*(1.-2.*U)))
CPS(I,2)=SQRT(YMOD/(2.*RO(I)*(1.+U)))
120
END OF DO LOOP?
NO
YES
240
260
YES
NTIME.EQ.0
NO
T=TT-DELT
VJ(1)=VELINT(T,2,V,1.150)
VJ(2)=VELINT(T,3,V,0.150)
260
DO LOOP 490
I=1,NQ
I1=IQ(I,1)
I2=IQ(I,2)
I3=IQ(I,3)
I4=IQ(I,4)
I5=IQ(I,5)
I1.EQ.1
YES
IO=0
NO
I1.NE.1
YES
IO=275
NO

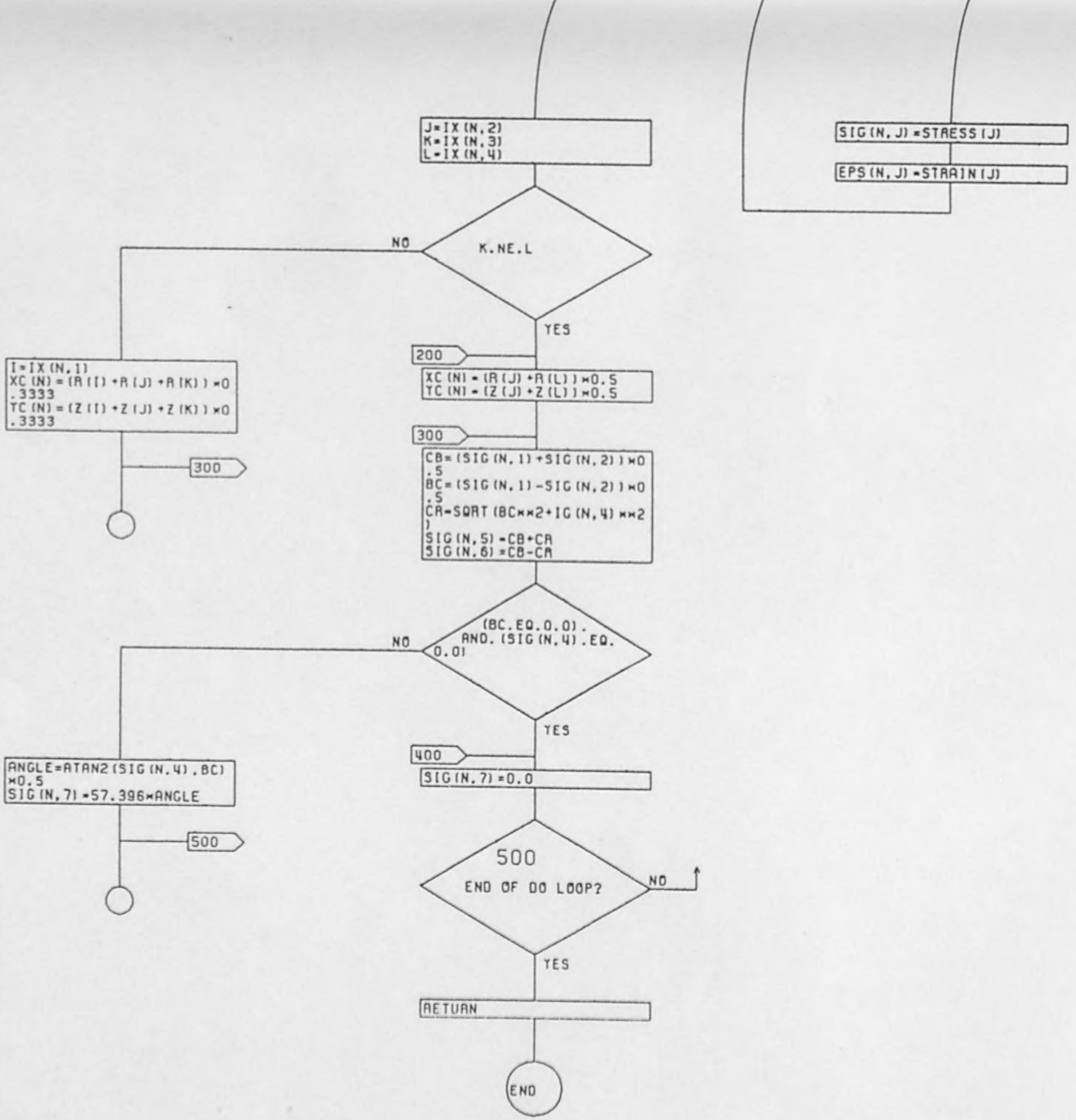

DO LOOP 480
J=1,2
DEL=ABS(A(I2+I0)-A(I3+I0))/CPS(I4,J)
T=TT-DELT-DEL
T.LE.0.0
YES
480
NO
I1.EQ.1
YES
N=J
NO
I1.NE.1
YES
N=3-J
NO
DUM1=XI(2*I3-2+N)
DUM2=XI(2*I2-2+N)
NTIME.EQ.0.
NO
YES
460
DV(2*I5-2+N)=(DUM2*(DEL-DELT)+DUM1*DELT)/DEL
480
480
END OF DO LOOP?
NO
YES
IV(I5).EQ.0
YES
460
NO

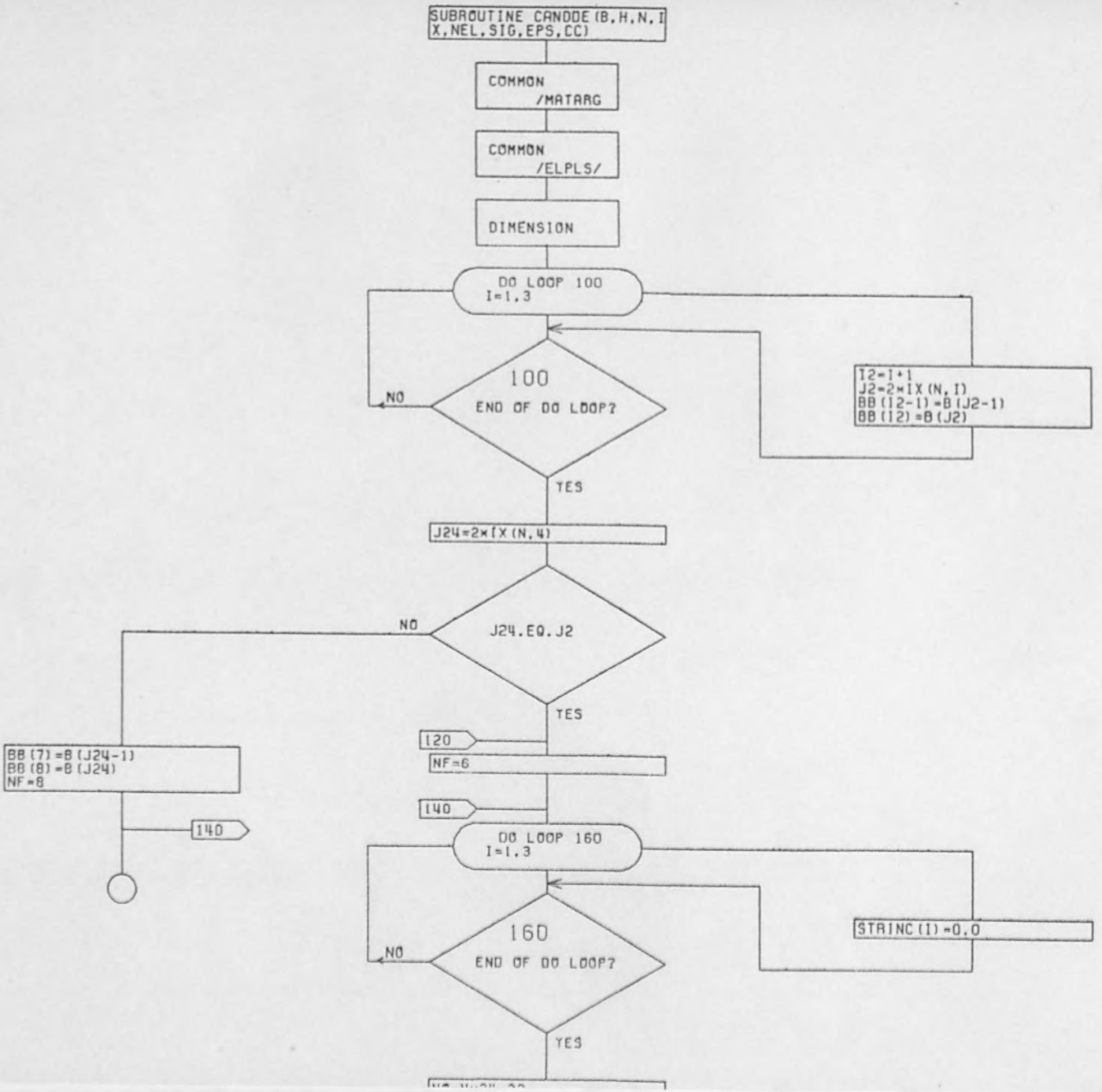

DUM1=DUM1-VELINT(T,N+1,V,1,150)
DUM2=DUM2-VJ(N)
490
END OF DO LOOP?
NO
YES
600
YES
NTIME.EQ.0
NO
500
T=TT
VI1=VELINT(T,2,V,1,150)
VI2=VELINT(T,3,V,0,150)
DO LOOP 520
I=1,KBP
520
END OF DO LOOP?
NO
YES
520
YES
IV(I).EQ.0
NO
I2=2*I
I1=I2-1
DV(I1)=DV(I1)+VI1
DV(I2)=DV(I2)+VI2
600
CONST=DELT/3.0
DO LOOP 620
I=1,KBP
N=KBNP(I)
I2=2*I
I1=I2-1
N2=2*N
N1=N2-1
DV(I1)=CONST*(DV(I1)+2.0*X1(N1)+0.5*DELT*X2(N1))
DV(I2)=CONST*(DV(I2)+2.0*X1(N2)+0.5*DELT*X2(N2))
620
END OF DO LOOP?
NO

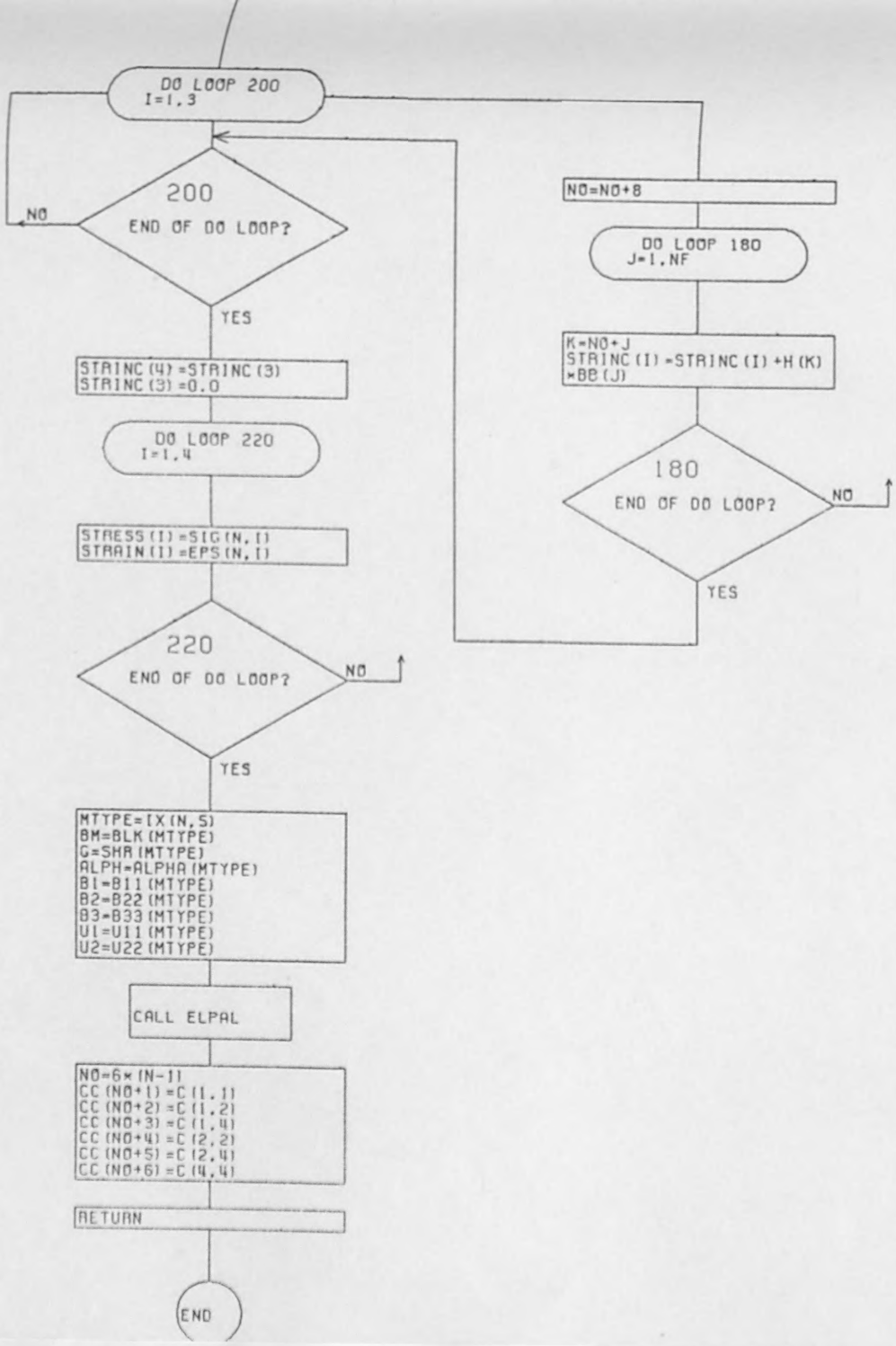

DO LOOP 820
I=1,KBP
N=KBNP(I)
N2=2*N
N1=N2-1
A(N1,1)=1.0
A(N2,1)=1.0
DO LOOP 760
J=2,MBAND
760
END OF DO LOOP?
NO
YES
780
DO LOOP 800
J=2,MBAND
J1=J-1
N3=N1-J1
N4=N2-J1
B(N4)=B(N4)-A(N4,J)*DV(2*I)
A(N4,J)=0.0
N3.LT.1
NO
YES
820
B(N3)=B(N3)-A(N3,J)*DV(2*I-1)
A(N3,J)=0.0
800
END OF DO LOOP?
NO
YES
820
END OF DO LOOP?
NO
YES
840
J1=J-1
N11=N1+J1
N22=N2+J1
B(N11)=B(N11)-A(N1,J)*DV(2*I-1)
A(N1,J)=0.0
780
N22.GT.NEQ
YES
NO
B(N22)=B(N22)-A(N2,J)*DV(2*I)
A(N2,J)=0.0

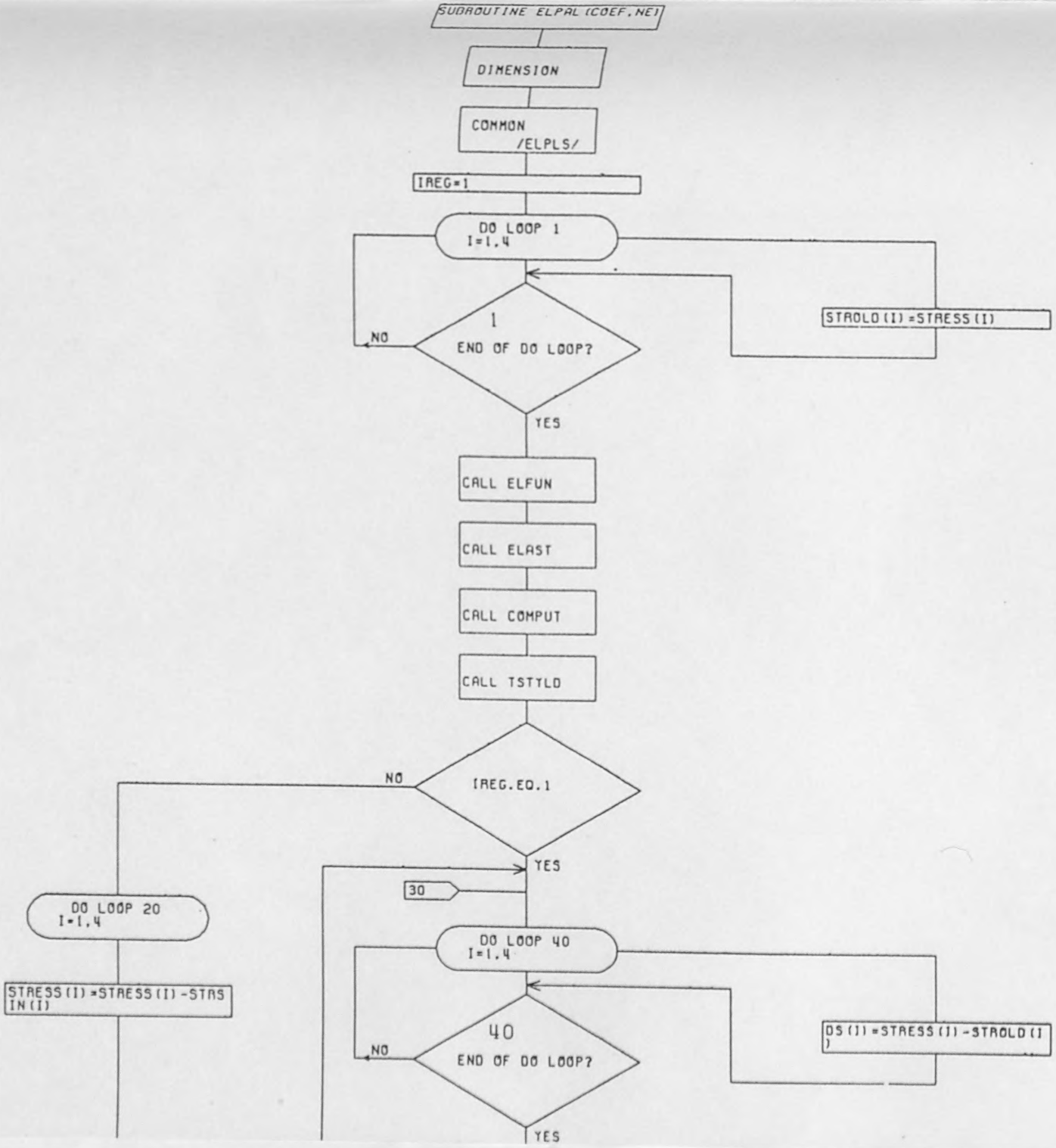

DO LOOP 890
N=1,NUMNP
890
END OF DO LOOP?
NO
YES
DO LOOP 900
I=1,KBP
N=KBNP(I)
B(2*N-1)=DV(2*I-1)
B(2*N)=DV(2*I)
900
END OF DO LOOP?
NO
YES
RETURN
END
KK=KODE(N)
890
YES
KK.EQ.0
NO
870
<0
KK-10
=0
880
860
>0
860
B(2*N-1)=0.
870
B(2*N)=0.
890
880
B(2*N-1)=0.

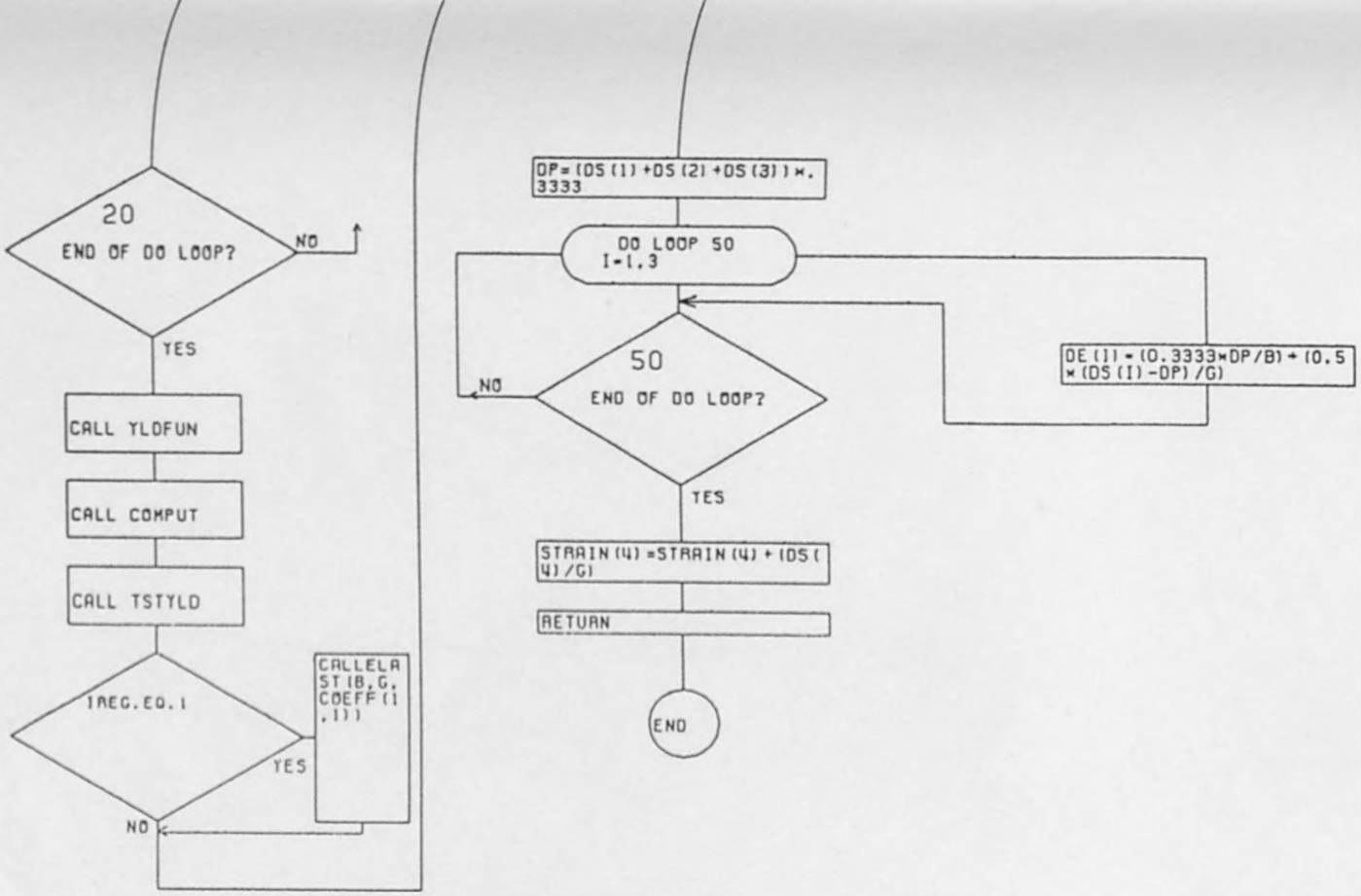

REALMASS
COMMON /SS/
COMMON /EL/
COMMON /SOL/
COMMON /DAMPNG
DIMENSION
DIMENSION
EQUIVALENCE (NPAR(2),NUME), (NPAR(16),NUMMAT), (NPAR(15),MODEL), (NPAR(10),NINT), (NPAR(1)),NPAR1), (NPAR(7),MXNODS), (NPAR(5),ITYP20), (NPAR(3),INDNL)
DATA
DATA
DATA
DO LOOP 500 N=1,NUMEL
CALL CANDDE
DO LOOP 100 J=1,4
100 END OF DO LOOP?
NO
YES

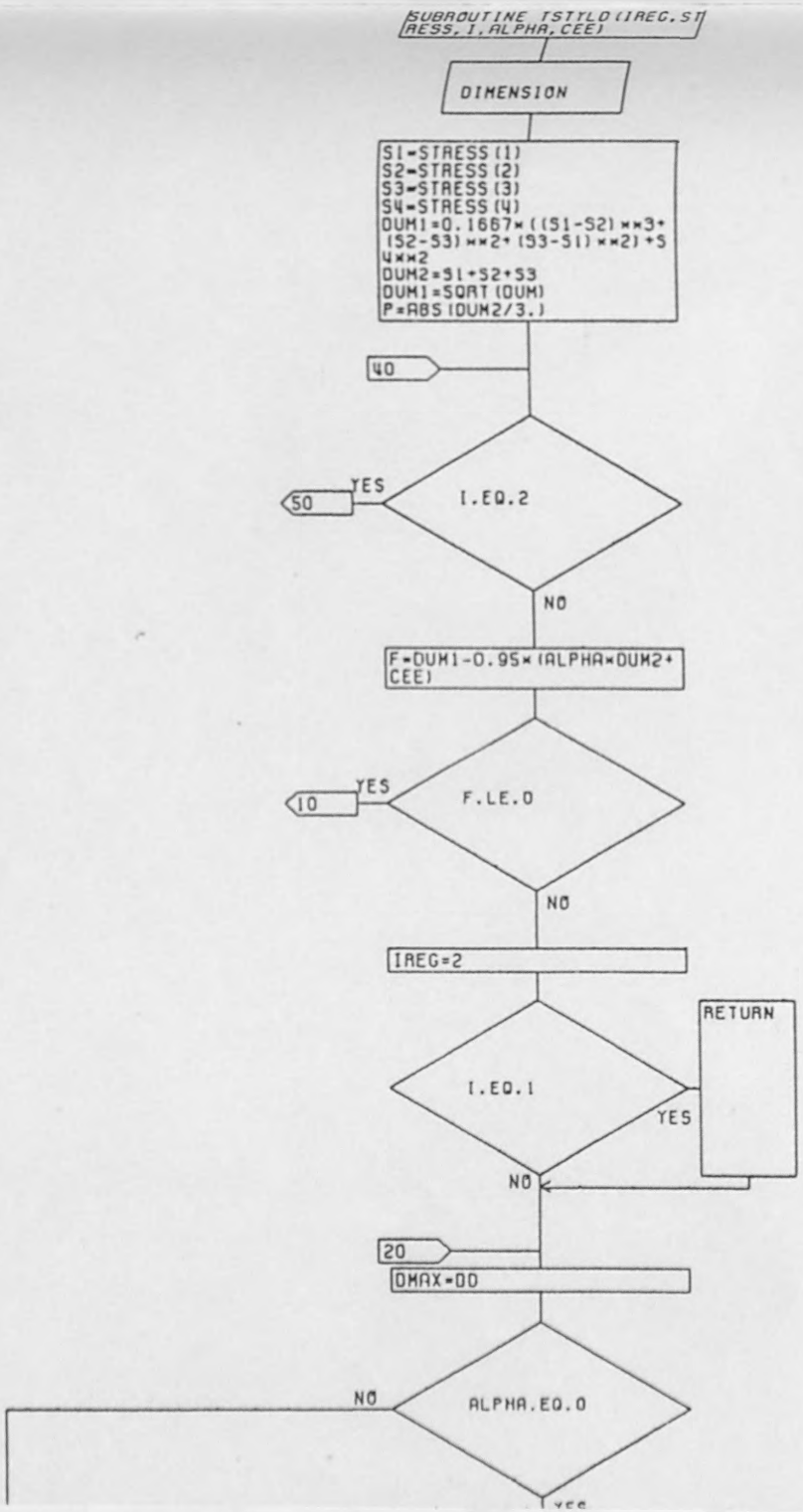

J=IX(N,2)
K=IX(N,3)
L=IX(N,4)
K.NE.L
NO
YES
I=IX(N,1)
XC(N)=(R(I)+R(J)+R(K))*0
.3333
TC(N)=(Z(I)+Z(J)+Z(K))*0
.3333
300
200
XC(N)=(R(J)+R(L))*0.5
TC(N)=(Z(J)+Z(L))*0.5
300
CB=(SIG(N,1)+SIG(N,2))*0
.5
BC=(SIG(N,1)-SIG(N,2))*0
.5
CR=SQRT(BC**2+IG(N,4)**2
)
SIG(N,5)=CB+CR
SIG(N,6)=CB-CR
(BC.EQ.0.0).
AND.(SIG(N,4).EQ.
0.0)
NO
YES
ANGLE=ATAN2(SIG(N,4),BC)
*0.5
SIG(N,7)=57.296*ANGLE
500
400
SIG(N,7)=0.0
500
END OF DO LOOP?
NO
YES
RETURN
END
SIG(N,J)=STRESS(J)
EPS(N,J)=STRAIN(J)

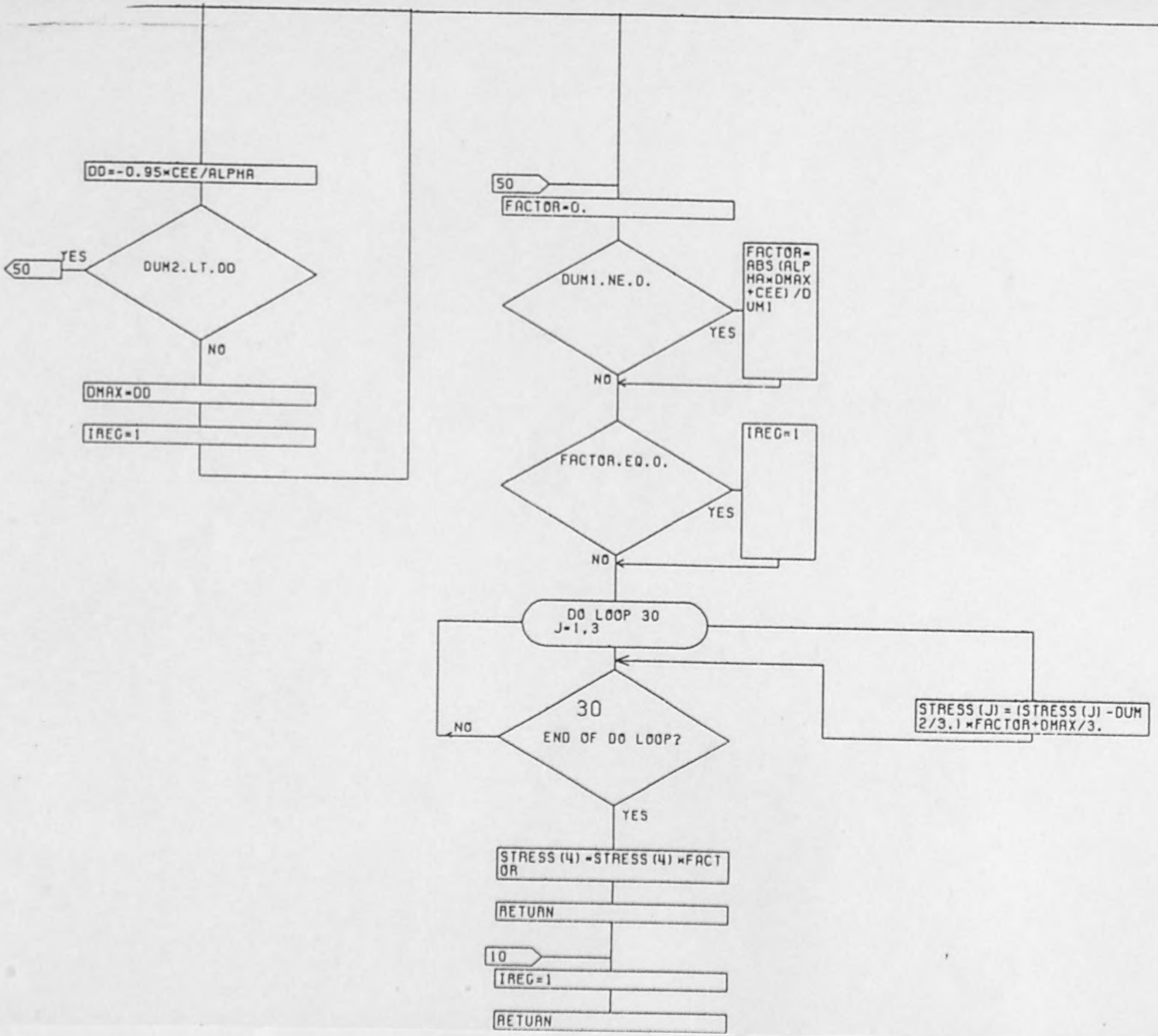

SUBROUTINE CANDDE(B,H,N,IX,NEL,SIG,EPS,CC)
COMMON /MATARG
COMMON /ELPLS/
DIMENSION
DO LOOP 100 I=1,3
100 END OF DO LOOP?
NO
YES
I2=I+1
J2=2*IX(N,I)
BB(I2-1)=B(J2-1)
BB(I2)=B(J2)
J24=2*IX(N,4)
J24.EQ.J2
NO
YES
BB(7)=B(J24-1)
BB(8)=B(J24)
NF=8
140
120
NF=6
140
DO LOOP 160 I=1,3
160 END OF DO LOOP?
NO
YES
STRINC(I)=0.0

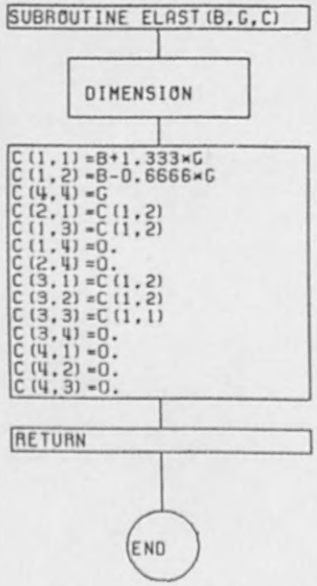

DO LOOP 200
I=1,3
200
END OF DO LOOP?
NO
YES
STRINC(4)=STRINC(3)
STRINC(3)=0.0
DO LOOP 220
I=1,4
STRESS(I)=SIG(N,I)
STRAIN(I)=EPS(N,I)
220
END OF DO LOOP?
NO
YES
MTYPE=IX(N,5)
BM=BLK(MTYPE)
G=SHR(MTYPE)
ALPH=ALPHA(MTYPE)
B1=B11(MTYPE)
B2=B22(MTYPE)
B3=B33(MTYPE)
U1=U11(MTYPE)
U2=U22(MTYPE)
CALL ELPAL
NO=6*(N-1)
CC(NO+1)=C(1,1)
CC(NO+2)=C(1,2)
CC(NO+3)=C(1,4)
CC(NO+4)=C(2,2)
CC(NO+5)=C(2,4)
CC(NO+6)=C(4,4)
RETURN
END
NO=NO+8
DO LOOP 180
J=1,NF
K=NO+J
STRINC(I)=STRINC(I)+H(K)
*BB(J)
180
END OF DO LOOP?
NO
YES

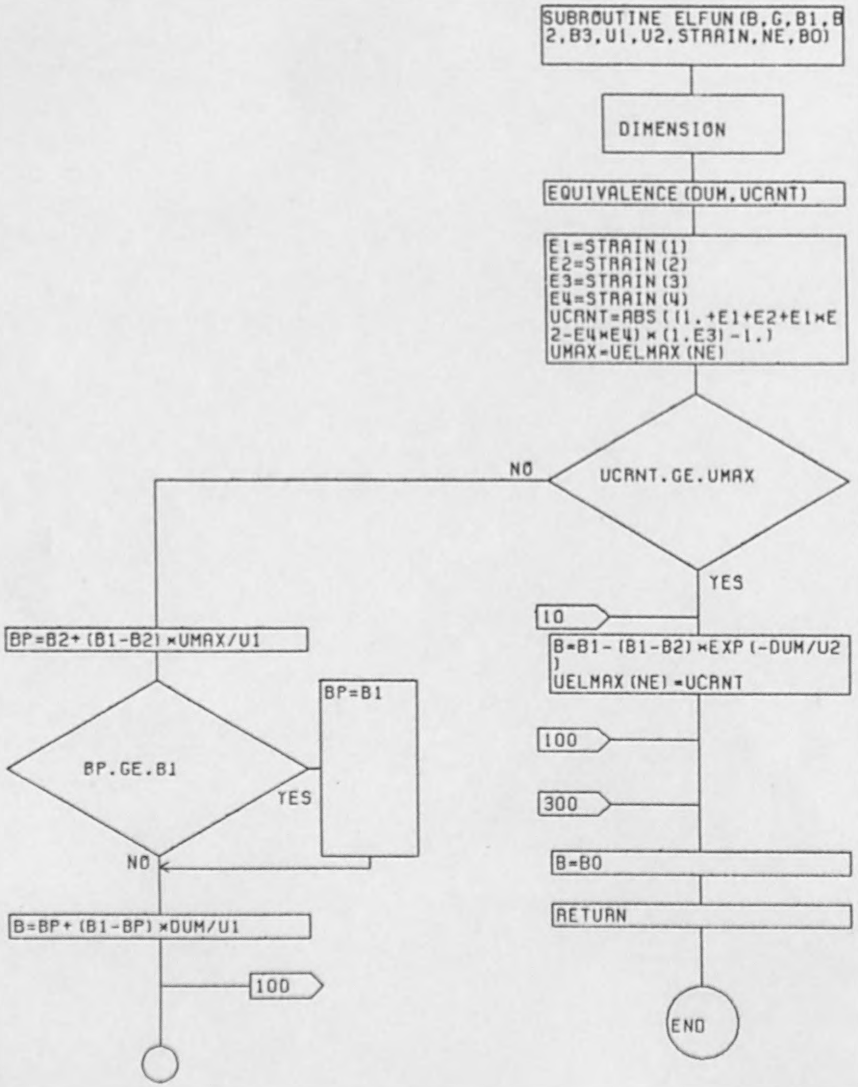

SUBROUTINE ELPAL(COEF,NE)
DIMENSION
COMMON /ELPLS/
IREG=1
DO LOOP 1 I=1,4
1
END OF DO LOOP?
NO
YES
STROLD(I)=STRESS(I)
CALL ELFUN
CALL ELAST
CALL COMPUT
CALL TSTYLD
IREG.EQ.1
NO
YES
30
DO LOOP 20 I=1,4
STRESS(I)=STRESS(I)-STRAIN(I)
DO LOOP 40 I=1,4
40
END OF DO LOOP?
NO
YES
DS(I)=STRESS(I)-STROLD(I)

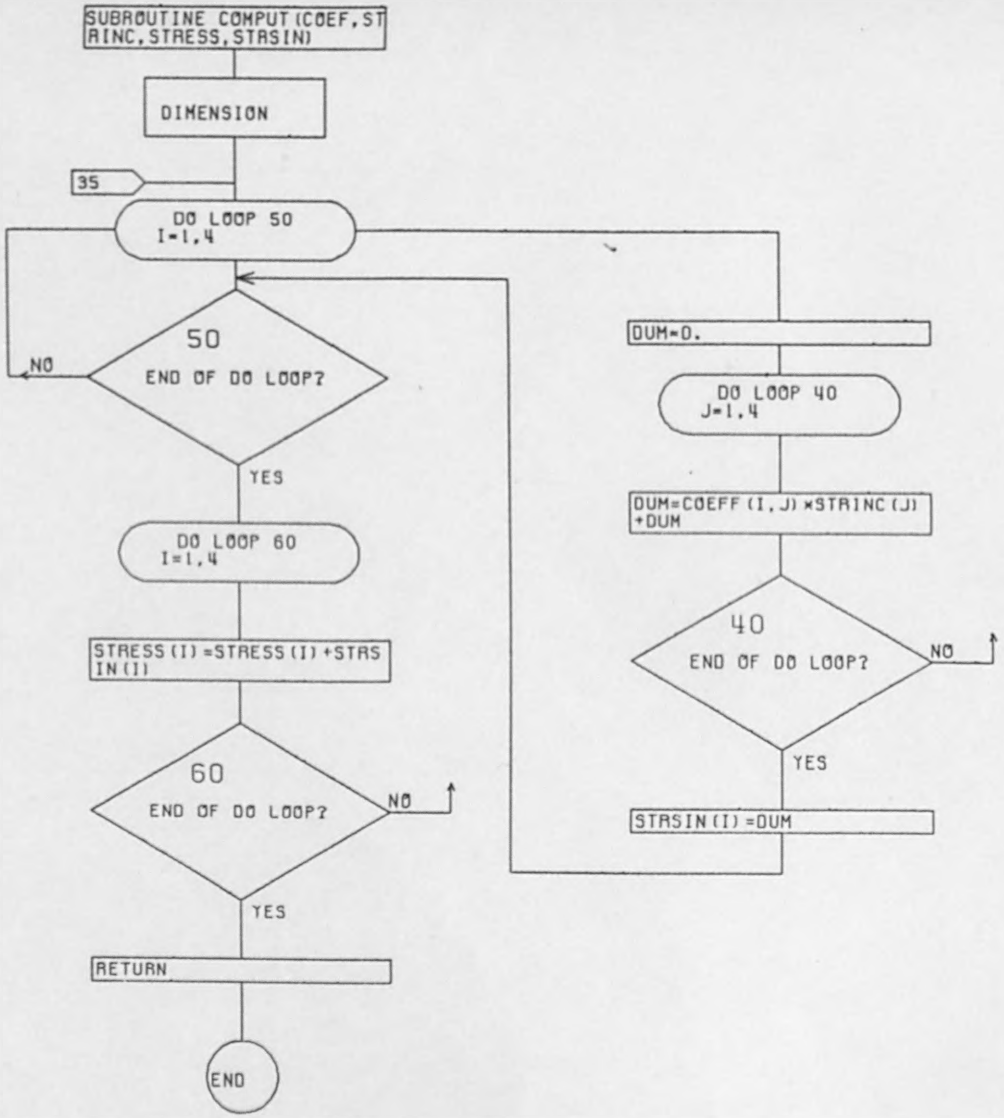

20
END OF DO LOOP?
NO
YES
CALL YLDFUN
CALL COMPUT
CALL TSTYLD
IREG.EQ.1
YES
NO
CALLELAST(B,G,COEFF(1,1))
DP=(DS(1)+DS(2)+DS(3))*.3333
DO LOOP 50
I=1,3
50
END OF DO LOOP?
NO
YES
DE(I)=(0.3333*DP/B)+(0.5*(DS(I)-DP)/G)
STRAIN(4)=STRAIN(4)+(DS(4)/G)
RETURN
END

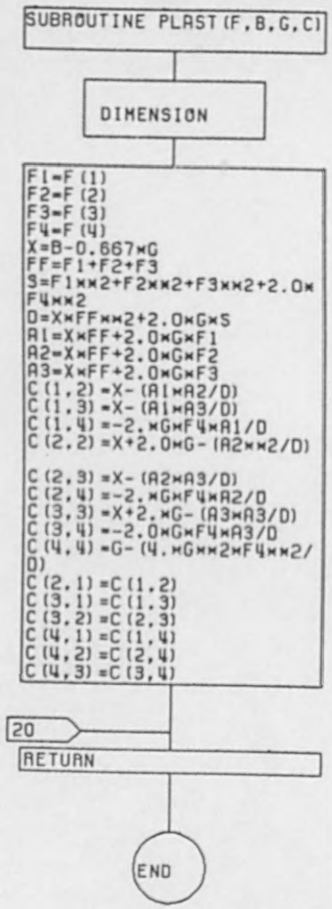

SUBROUTINE ISTILD(IREG,ST
RESS,I,ALPHA,CEE)
DIMENSION
S1=STRESS(1)
S2=STRESS(2)
S3=STRESS(3)
S4=STRESS(4)
DUM1=0.1667*((S1-S2)**3+
(S2-S3)**2+(S3-S1)**2)+S
4**2
DUM2=S1+S2+S3
DUM1=SQRT(DUM)
P=ABS(DUM2/3.)
40
I.EQ.2
YES
50
NO
F=DUM1-0.95*(ALPHA*DUM2+
CEE)
F.LE.0
YES
10
NO
IREG=2
I.EQ.1
YES
RETURN
NO
20
DMAX=00
ALPHA.EQ.0
NO
YES

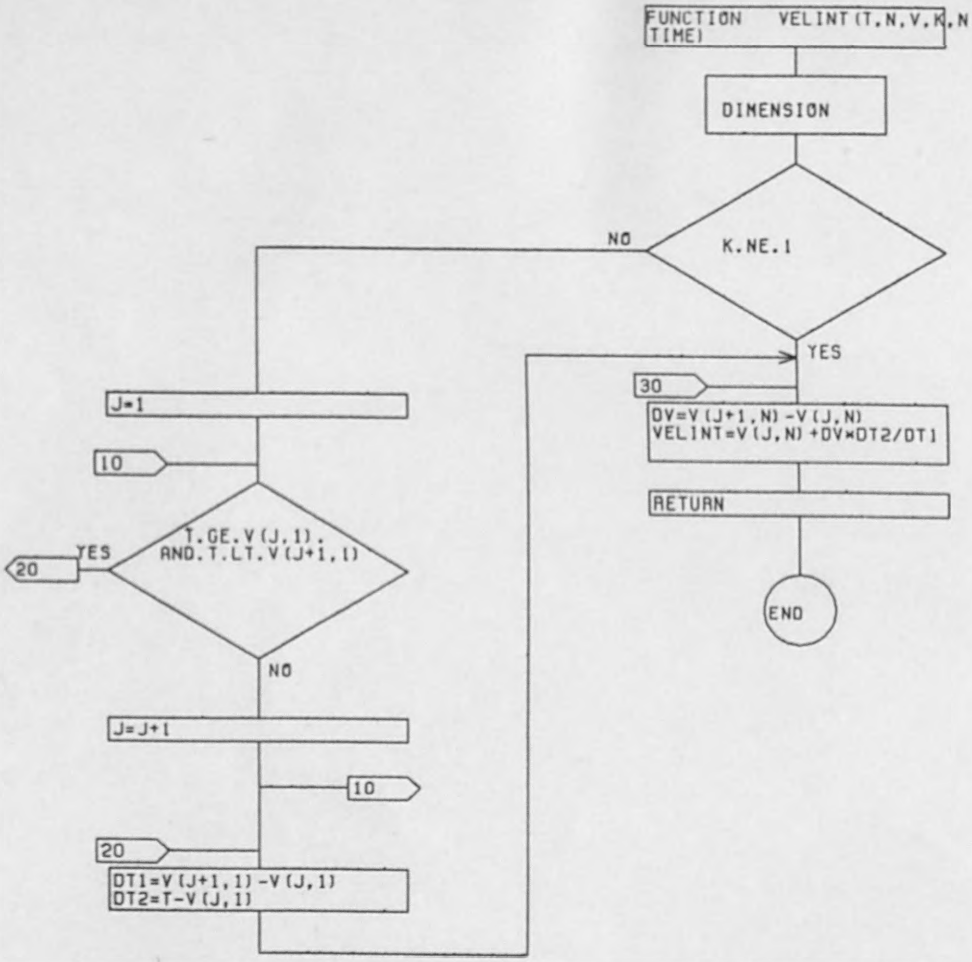

DD=-0.95*CEE/ALPHA
DUM2.LT.DD
YES
50
NO
DMAX=DD
IREG=1
50
FACTOR=0.
DUM1.NE.0.
YES
FACTOR=ABS(ALPHA*DMAX+CEE)/DUM1
NO
FACTOR.EQ.0.
YES
IREG=1
NO
DO LOOP 30
J=1,3
30
END OF DO LOOP?
NO
STRESS(J)=(STRESS(J)-DUM2/3.)*FACTOR+DMAX/3.
YES
STRESS(4)=STRESS(4)*FACTOR
RETURN
10
IREG=1
RETURN

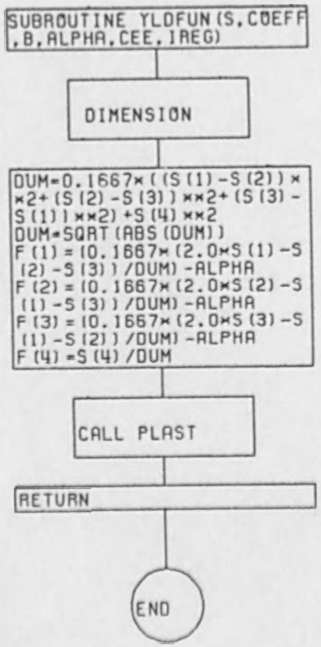

SUBROUTINE ELAST(B,G,C)
DIMENSION
C(1,1)=B+1.333*G
C(1,2)=B-0.6666*G
C(4,4)=G
C(2,1)=C(1,2)
C(1,3)=C(1,2)
C(1,4)=0.
C(2,4)=0.
C(3,1)=C(1,2)
C(3,2)=C(1,2)
C(3,3)=C(1,1)
C(3,4)=0.
C(4,1)=0.
C(4,2)=0.
C(4,3)=0.
RETURN
END


SUBROUTINE ELFUN(B,G,B1,B2,B3,U1,U2,STRAIN,NE,BO)
DIMENSION
EQUIVALENCE(DUM,UCRNT)
E1=STRAIN(1)
E2=STRAIN(2)
E3=STRAIN(3)
E4=STRAIN(4)
UCRNT=ABS((1.+E1+E2+E1*E2-E4*E4)*(1.E3)-1.)
UMAX=UELMAX(NE)
UCRNT.GE.UMAX
NO
YES
10
B=B1-(B1-B2)*EXP(-DUM/U2)
UELMAX(NE)=UCRNT
100
300
B=BO
RETURN
END
BP=B2+(B1-B2)*UMAX/U1
BP.GE.B1
YES
BP=B1
NO
B=BP+(B1-BP)*DUM/U1
100


SUBROUTINE COMPUT(COEF,STRINC,STRESS,STRSIN)
DIMENSION
35
DO LOOP 50
I=1,4
50
END OF DO LOOP?
NO
YES
DO LOOP 60
I=1,4
STRESS(I)=STRESS(I)+STRSIN(I)
60
END OF DO LOOP?
NO
YES
RETURN
END
DUM=0.
DO LOOP 40
J=1,4
DUM=COEFF(I,J)*STRINC(J)+DUM
40
END OF DO LOOP?
NO
YES
STRSIN(I)=DUM


SUBROUTINE PLAST(F,B,G,C)
DIMENSION
F1=F(1)
F2=F(2)
F3=F(3)
F4=F(4)
X=B-0.667*G
FF=F1+F2+F3
S=F1**2+F2**2+F3**2+2.0*
F4**2
D=X*FF**2+2.0*G*S
A1=X*FF+2.0*G*F1
A2=X*FF+2.0*G*F2
A3=X*FF+2.0*G*F3
C(1,2)=X-(A1*A2/D)
C(1,3)=X-(A1*A3/D)
C(1,4)=-2.*G*F4*A1/D
C(2,2)=X+2.0*G-(A2**2/D)
C(2,3)=X-(A2*A3/D)
C(2,4)=-2.*G*F4*A2/D
C(3,3)=X+2.*G-(A3*A3/D)
C(3,4)=-2.0*G*F4*A3/D
C(4,4)=G-(4.*G**2*F4**2/
D)
C(2,1)=C(1,2)
C(3,1)=C(1,3)
C(3,2)=C(2,3)
C(4,1)=C(1,4)
C(4,2)=C(2,4)
C(4,3)=C(3,4)
20
RETURN
END


FUNCTION VELINT(T,N,V,K,N
TIME)
DIMENSION
K.NE.1
NO
YES
30
DV=V(J+1,N)-V(J,N)
VELINT=V(J,N)+DV*DT2/DT1
RETURN
END
J=1
10
T.GE.V(J,1).
AND.T.LT.V(J+1,1)
YES
20
NO
J=J+1
10
20
DT1=V(J+1,1)-V(J,1)
DT2=T-V(J,1)


SUBROUTINE YLDFUN(S,COEFF
,B,ALPHA,CEE,IREG)
DIMENSION
DUM=0.1667*((S(1)-S(2))*
*2+(S(2)-S(3))**2+(S(3)-
S(1))**2)+S(4)**2
DUM=SQRT(ABS(DUM))
F(1)=(0.1667*(2.0*S(1)-S
(2)-S(3))/DUM)-ALPHA
F(2)=(0.1667*(2.0*S(2)-S
(1)-S(3))/DUM)-ALPHA
F(3)=(0.1667*(2.0*S(3)-S
(1)-S(2))/DUM)-ALPHA
F(4)=S(4)/DUM
CALL PLAST
RETURN
END