To the Graduate Council:

I am submitting herewith a thesis written by Douglas Walter Stillman entitled "INGRID: A Three-Dimensional Mesh Generator Based on an Expanded Index Notation." I recommend that it be accepted in partial fulfillment of the requirements for the degree of Master of Science, with a major in Engineering Science.

Jerry E. Stoneking, Major Professor

We have read this thesis
and recommend its acceptance:

John E Akin

Bruce R Dewey

Accepted for the Council:

L Evans Roth

Vice Chancellor
Graduate Studies and Research

INGRID: A THREE-DIMENSIONAL MESH GENERATOR BASED ON AN EXPANDED INDEX NOTATION

A Thesis

Presented for the

Master of Science

Degree

The University of Tennessee, Knoxville

Douglas Walter Stillman

December 1981

ACKNOWLEDGEMENT

INGRID was partially funded by Los Alamos National Laboratory, Contract No. 4-L-11-4308R-1.

The author would like to express his appreciation to Dr. J. E. Stoneking and Dr. J. E. Akin for their advice in the development of this program. The author also appreciates the time and effort that Eunice Hinkle and Angela Burleson spent in preparing this report.

Finally, the author would like to express his thankfulness to Ying Chi Wu for her support and encouragement in the completion of this project.

ABSTRACT

This report presents a program named INGRID which aids in the preparation of models for finite element programs. An expanded I, J, K indexing scheme is provided to minimize input data. Algorithms for duplicate node removal, structure plotting, and hidden surface removal are presented to improve computational efficiency.

TABLE OF CONTENTS

SECTION PAGE

APPENDICES (Continued)

LIST OF TABLES

LIST OF FIGURES

FIGURE PAGE

I. INTRODUCTION

Automatic mesh generation is a useful tool for preparing finite element models, and numerous computer programs are available which have mesh generation capability.[1] An efficient index mesh generation technique was developed by Cook.[2] This author expanded Cook's index mesh generation scheme and developed a more general index notation for defining and manipulating three dimensional computer models of objects. The expanded index scheme is implemented in a computer program designated INGRID.

The use of generalized index notation enables INGRID to perform isoparametric mapping, drag mesh, replication, and warping of structures. Complex combinations of mapping, drag mesh, replication, and warping occur naturally when using index notation and do not require any special programming.

INGRID's features include:

- Seven functions which permit a wide variety of mesh generation techniques.
- Interactive plotting.
- Calculation of desired boundary conditions for combinations of plate, beam, and solid elements.
- Generated output can be selected to be compatible with SAP5,[3] SAP6,[4] PAFEC,[5] and MOVIE.BYU.[6]

Two dimensional index mesh generation was first implemented by Becker and Brisbane.[7] Three dimensional indexing is based on body-oriented coordinate systems such as is shown in Figure 1.[8] This body can be defined in the (X,Y,Z) coordinate system; however, an easier approach is to use the (α, β, γ) coordinate system to generate

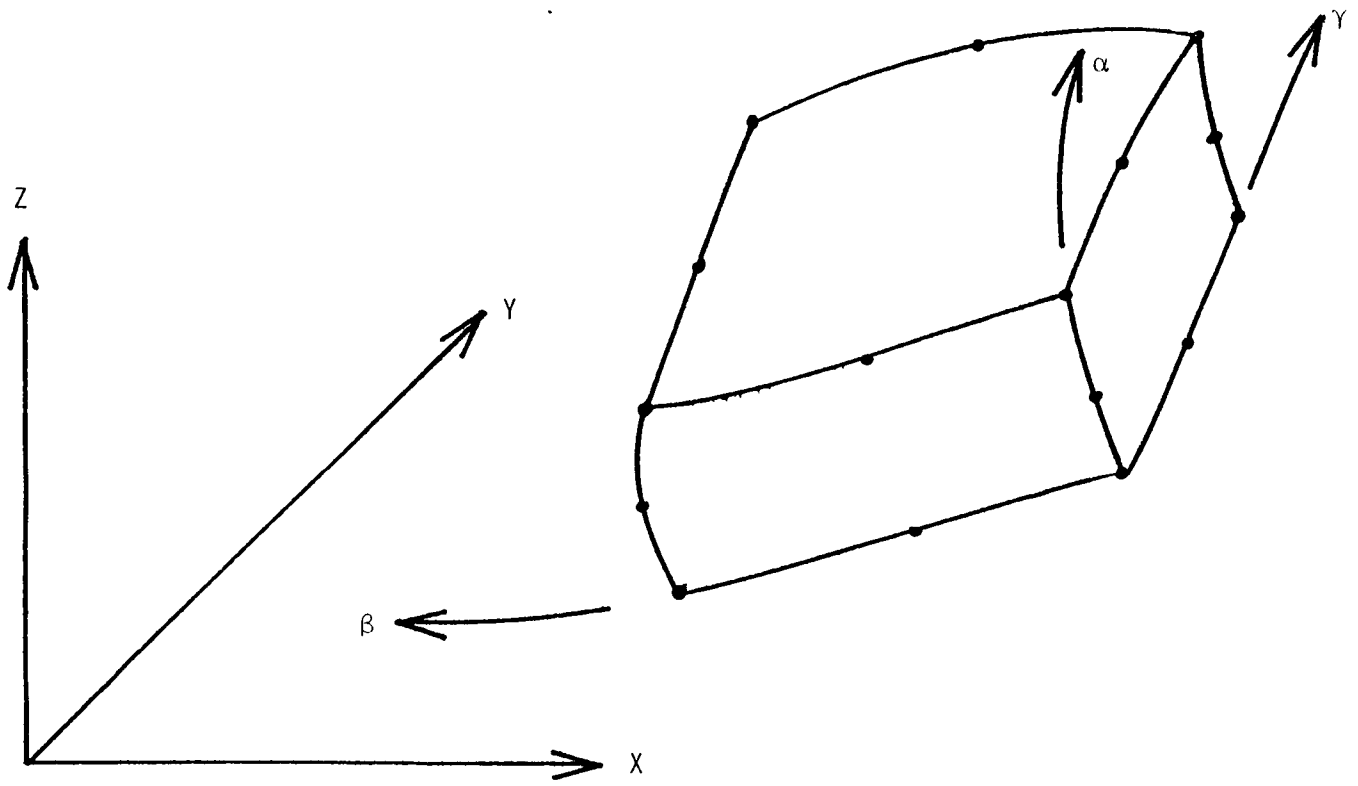


Figure 1. Body Oriented Coordinates

the mesh and then change to rectangular coordinates. If positions along the (α, β, γ) axes are assigned integer values, then the coordinate transformation is a mapping from index space to the object. The structure can be manipulated in index space and then mapped into its final form in cartesian coordinates. A detailed description of index notation is in Appendix A.

Two computer programs which use index notation are INGEN[2] and SLIC.[3] These programs provide several methods for manipulating meshes in the index space, but these methods have limitations. Their ability to transform from integer space to cartesian coordinates is minimal and, as a consequence, much of the usefulness of the notation is lost. The index space can be an obstacle when generating meshes since different parts of a structure require different body oriented coordinate systems.

INGRID avoids these problems by modifying the integer space. Structures are divided into parts which each have their own natural body-oriented coordinates. The parts are individually generated in their own integer space and mapped into their final location in the object space. This not only allows parts to be generated in their own coordinate system, but also allows parts to be generated completed independent of each other.

Table I shows the different types of functions available in INGRID. All mesh generation programs use the Type 1 functions. Type 1 functions include assigning coordinates to a node, generating nodes along lines or arcs, and generating nodes in the interior of plane or solid regions. The Type 2 function changes a single coordinate at a given node point. This is used, for example, to place groups of nodes along the same line or plane. Type 3 functions are used to warp planes

TABLE I

FUNCTIONS

Type	Function	Description
1	$\bar{P} = \bar{F}(\alpha, \beta, \gamma)$	Point Generation
2	$Q = F(\alpha, \beta, \gamma)$	Coordinate Generation
3	$\bar{P} = \bar{F}(\alpha, \beta, \gamma) + \bar{P}$	Warping Function

P ~ Position of point

Q ~ Position of point (α, β, γ) along the X, W, or Z axis.

or solids to match boundaries. Cylindrical surfaces can be made by using a single warping function.

The Type 2 and Type 3 functions are not unique to index mesh generators; however, they are natural to use with the body-oriented coordinates and give index mesh generation a significant advantage over other methods. INGEN provides only Type 1 functions, and thus much of the advantage in using the index space is not utilized.

Another problem with index mesh generation schemes is the amount of data necessary to describe regions in the index space. To generate a cube using INGEN or SLIC requires 16 cards - eight cards to define the eight vertices, six cards to define the six surfaces, one card to generate the nodes, and one card to generate the elements. Input data requirements are simplified in INGRID. The program reads a command to

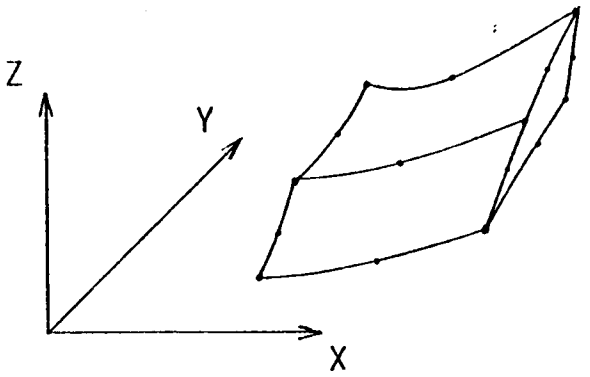
generate elements in a region and internally generates the other required fifteen commands. Further reductions in input data are achieved by using index progressions which are defined on page 51 of Appendix A. The index progression provides a simple method for defining multiple regions in the index space. INGRID uses the expanded notation to minimize the amount of input data necessary for complex structures.

Comparisons with Other Mesh Generation Schemes

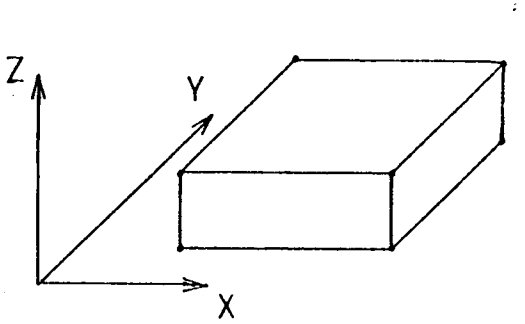
A comparison of the input data requirements for the computer programs PAFEC, INGEN, and INGRID is given using the examples in Figure 2. TABLE II shows the amount of data required by each of these programs for the cases in Figure 2. PAFEC uses node numbering to label nodes while INGEN and INGRID use indexing. PAFEC was chosen for the comparison since PAFEC's mesh generation scheme is similar to many other programs.

For Case 1, Table II shows that INGRID substantially reduces the amount of data required by INGEN to define the mesh. PAFEC requires less data than INGRID; however, the indexing system has further advantages over the node numbering system used in PAFEC. Boundary conditions, material numbers, and other properties within a region can be specified easily using indexing, whereas node numbering provides very little control of mesh properties.

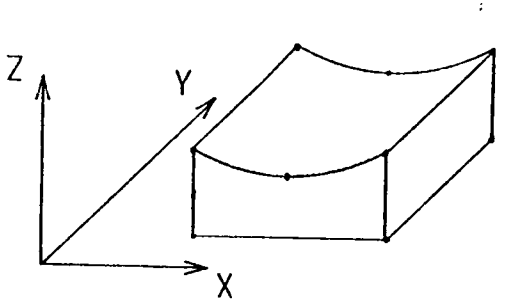
Many mesh generation problems have convenient body oriented coordinate systems such as are used in Case 2 of Figure 2. The mesh in Case 2 can be mapped onto an object by performing a simple coordinate transformation. Although INGEN is designed to take advantage of body oriented coordinate systems, Table II shows very little savings of



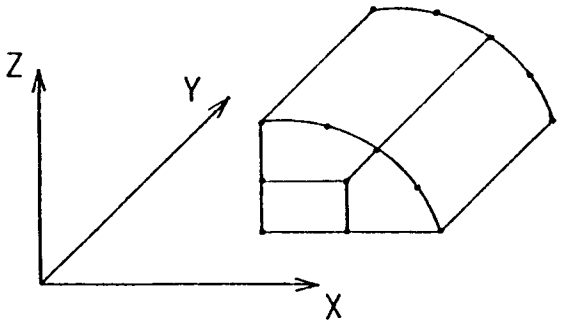
Case 1 - General Solid Region



Case 2 - Region with Body Oriented Coordinates



Case 3 - Slightly Irregular Region



Case 4 - Multiple Regions

Figure 2. Mesh Generation Problems

TABLE II

COMPARISON OF INPUT DATA REQUIREMENTS

Case	PAFEC			INGEN			INGRID		
	a[1]	b[2]	c[3]	a[1]	b[2]	c[3]	a[1]	b[2]	c[3]
1	43	60	23	144	60	28	102	60	23
2	19	24	12	144	24	28	6	6	6
3	23	30	14	144	30	28	18	9	8
4	28	36	17	222	42	42	44	14	10
4[4]	48	54	24						

[1]a ~ number of integers required

[2]b ~ number of floating point numbers required

[3]c ~ number of input cards required

[4]Case 4 is a special case for PAFEC. This is the amount of data that would be required for similar programs that did not have this as a special case.

input data for Case 2. The reductions in input data for PAFEC are the result of curved edges in Case 1 becoming straight in Case 2. INGRID, on the other hand, shows a 93% decrease in input data which is due primarily to using body-oriented coordinates. Cases 3 and 4 demonstrate that this advantage is not lost even if the body-oriented coordinates are not completely natural for the structure.

Many other generation techniques have been developed to solve special problems in mesh generation. If the mesh has certain properties, the input data can be reduced significantly. Some of these techniques which require special programming arise naturally when using body-oriented coordinates and the modified index system. An example of this is the drag mesh[10] generation system used in the UNISTRUC program.

The basic assumption of the drag mesh is that for certain objects the cross sectional area remains the same along some curvilinear axis such as the object in Figure 3. The cross section is first defined and the object is "dragged" along a circular arc to produce the mesh. This scheme simplifies mesh generation since the nodes which must be input are the ones which lie on the original cross section. Using the example in Figure 3, INGRID can generate the same mesh in the same manner as UNISTRUC. First, a cylindrical coordinate system is chosen for the body-oriented coordinate system. Functions are used which assign constant (R, Z) values to arcs of the mesh. Finally, the completed mesh is generated with less data than is required by UNISTRUC.

Program Features

INGRID was designed as a preprocessor for PAFEC, SAP5, and SAP6. Data from INGRID can be formatted to be compatible with any of these

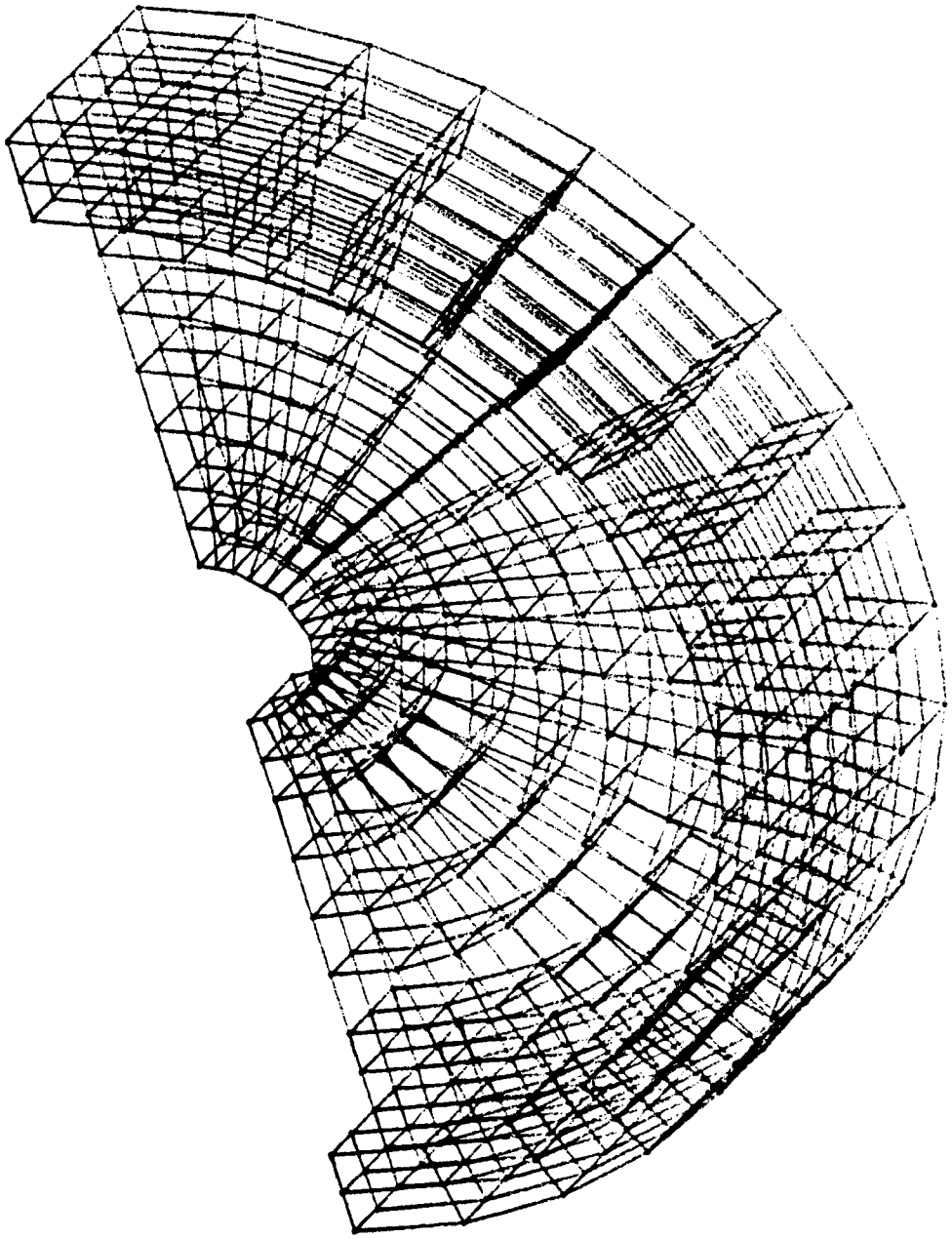


Figure 3. Example of a "Drag Mesh"

three programs. Further processing is performed to insure that the data are completely compatible with the program chosen. Solid elements are placed in either a right- or left-handed coordinate system depending on the program to be used. Midside nodes on 8-node plates are also moved so that they lie in the plane of the plate.

To increase the usefulness of INGRID, a boundary condition generator is included which automatically generates any necessary boundary conditions for any configuration of plates, beams, and solids. Since PAFEC has its own boundary condition generation, they are not generated for PAFEC. Boundary conditions can also be input to INGRID and generated in the final output.

INGRID provides an efficient plotting package to verify the accuracy of meshes. The plotting package takes advantage of information created during mesh generation to minimize the number of lines which are plotted. If each element is plotted individually as is done in many mesh generators, then a line which is common to several elements is plotted several times. INGRID tries to avoid plotting lines several times by generating plot commands directly from input information. INGRID also checks for consecutive plot vectors which lie in a line. These are plotted as one pen command to increase plotting speed. Curved lines are plotted for edges of quadratic elements.

A utility program, MOVE, reads special output data from INGRID and processes the data to be compatible with the MOVIE.BYU[6] plotting program Since MOVIE.BYU is primarily used to plot only the visible surface of a structure, MOVE attempts to remove any surfaces which cannot be seen. The solid element data from INGRID is read and each solid

element is divided into six faces, as required by MOVIE.BYU. If two solid elements are adjacent, then they share one face in common which is not visible. These faces are removed from the input to MOVIE.BYU to minimize the amount of computer time required for plotting.

Although the index notation presented is not adequate for all mesh generation problems, a large number of cases can be solved using this notation. Further additions to the notation can be made to increase the generality of the notation and reduce the amount of data necessary for many problems.

II. PROGRAM ORGANIZATION

A partial outline of subroutines in INGRID is shown in Figure 4. The program is divided into four segments. Subroutine INIT reads the job control cards and initializes INGRID. Subroutine DLEE generates nodes, elements, and a plot file. Interactive plotting is controlled by subroutine INPLOT and the final data processing and formatting is performed by subroutine OUTPUT.

Table III contains a brief description of the subroutines listed in Figure 4. Descriptions of the algorithms used to perform the various operations are also given.

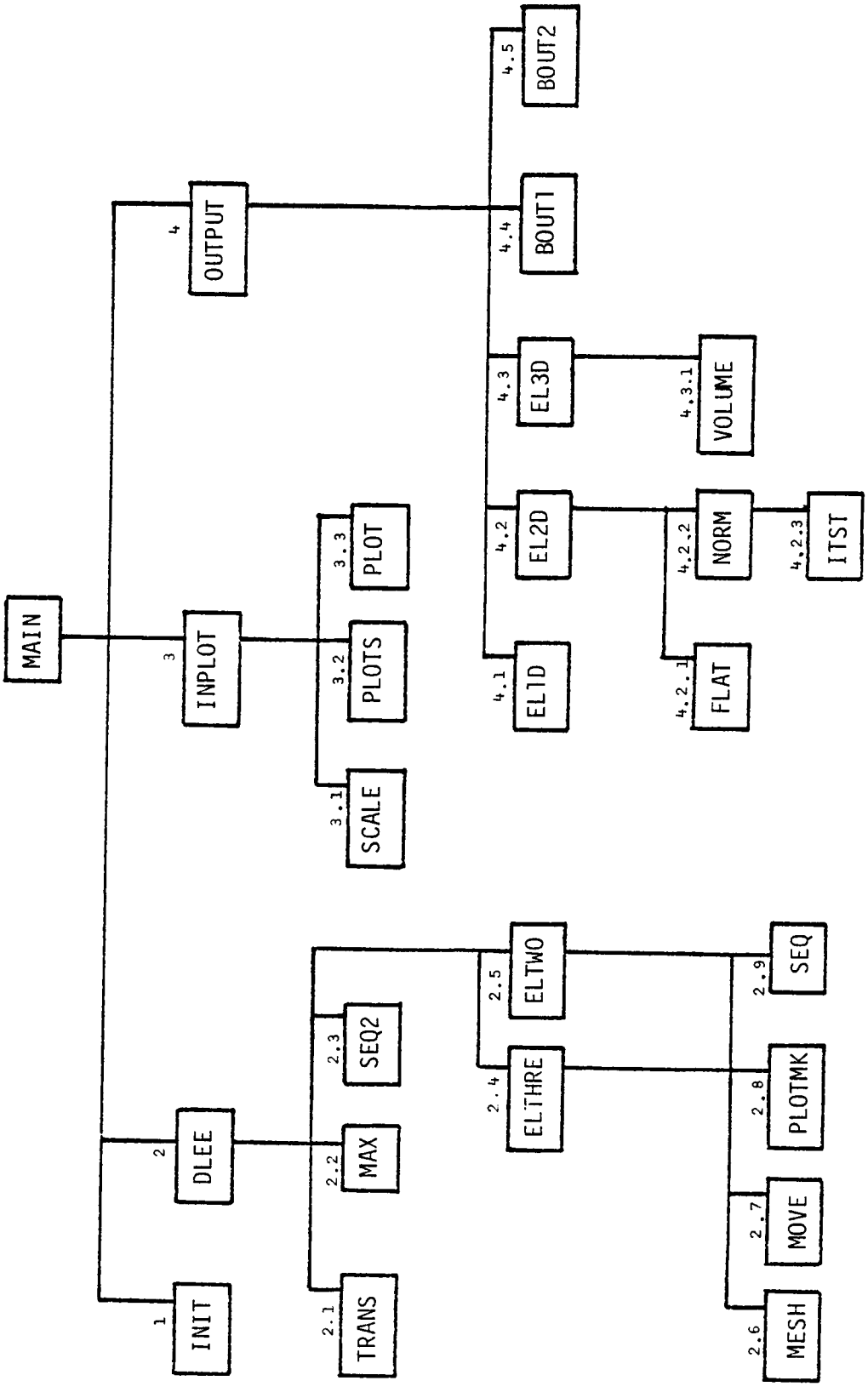


Figure 4. Program Organization

TABLE III

SUBROUTINE DESCRIPTIONS

Subroutine		Descriptions
2.1	TRANS	Mesh modules are transferred from the input data file onto unit 11. Comments are removed and commands to repeat the module are processed.
2.2	MAX	The dimensions of the system matrices are determined for the mesh module on unit 11.
2.3	SEQ2	Functions are read and executed. All vertices of regions are calculated.
2.4	ELTHRE	Nodes, elements, and a plot file are created for solid elements.
2.5	ELTWO	Nodes, elements, and a plot file are created for shell elements.
2.6	MESH	Locations of nodes within a region are determined by linear interpolation from the vertices.
2.7	MOVE	Vectors are added to nodes according to the arc commands in the input file.
2.8	PLOTMK	The plotfile is created for a region.
2.9	SEQ	The mesh module's index progression is divided into the regions defined.
3.1	SCALE	The limits of the plot data are determined and scaled to fit a Tektronix 4010 terminal.
3.2	PLOTS	Initialize plotting (similar to CALCOMP subroutine PLOTS).
3.3	PLOT	Plot a vector (Similar to CALCOMP subroutine PLOT).
4.1	EL1D	Reformatting and boundary conditions are determined for beam elements.
4.2	EL2D	Reformatting and boundary conditions are determined for plate elements.
4.3	EL3D	Reformatting and boundary conditions are determined for solid elements.

TABLE III (continued)

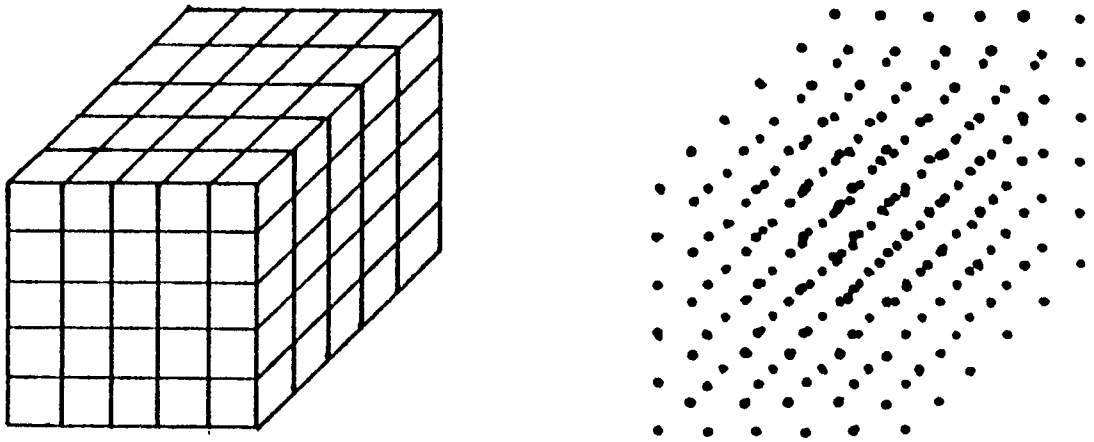
Subroutine		Descriptions
4.4	BOUT1	If the rotational constraints lie along a coordinate axis, then the appropriate global degrees of freedom are constrained.
4.5	BOUT2	Boundary elements are output to constrain rotations normal to plate elements.
4.2.1	FLAT	Midside nodes of a plate element are moved parallel to the elements' normal vector until they lie in the plane of the element.
4.2.2	NORM	The normal vector to a plate is determined.
4.2.3.	ITST	For each node of a plate element, the normal vector is compared to previously defined normal vectors of adjacent plate elements.
4.3.1	VOLUME	The approximate volume of a solid element is calculated to determine if it is defined in a left-handed coordinate system.

III. REGION GENERATION

Region generation is performed by subroutine SEQ and involves the interpretation of an index progression. The first time SEQ is called, it reads the index progression cards which are in the mesh module on unit 11. Progressions are stored using three matrices which are then optimized to decrease storage by subroutine DOUBLE. Subroutine DOUBLE also multiplies all indices by two when quadratic elements are requested instead of linear elements. The first region found is returned to the calling routine.

Subsequent calls to SEQ each return a design until all regions are determined. Finally the program returns a flag which indicates that no more data are available.

For each index progression, there can be defined a vertices matrix such as is shown in Figure 5 where each vertice is the vertice with the minimum indices for a possible region. SEQ compares all of the points in the vertices matrix for each of four possibilities. The vertice could be the minimum point for a solid region, or a plane of constant i, j, or k. Each time SEQ is called, a pointer is updated which points to the next possible region. If the region tested for does not exist at this vertice, the pointer is updated until either a region is found or there is no more data.



(a) (b)

Figure 5. Vertices of Regions

IV. NODE GENERATION

Node generation in INGRID is performed by subroutines MESH and MOVE. The initial coordinates of nodes are determined by linear interpolation in subroutine MESH. Displacement vectors are added to nodal points in subroutine MOVE to match curved boundaries.

Subroutine MESH calculates nodal point coordinates for each region by interpolating between vertices of the region. The following equations are used:

```
WI = (IX - I)/(IX - IM)
WJ = (JX - J)/(JX - JM)
WK = (KX - K)/(KX - KM)

X(I, J, K) = X(IM, JM, KM) * WI * WJ * WK + X (IX, JM, KM)
     * (1.0 - WI) * WJ * WK + X(IM, JX, KM) * WI * (1.0 - WJ) * WK
     + X(IM, JM, KX) * WI * WJ * (1.0 - WK) + X(IX, JX, KM)
     * (1.0 - WI) * (1.0 - WJ) * WK + X(IX, JM, KX) * (1.0 - WI) * WJ
     * (1.0 - WK) + X(IM, JX, KX) * WI * (1.0 - WJ) * (1.0 - WK)
     + X(IX, JX, KX) * (1.0 - WI) * (1.0 - WJ) * (1.0 - WK)

Y(I, J, K) = Y(IM, JM, KM) * WI * WJ * WK + Y(IX, JM, KM) * (1.0 - WI)
     * WI * WK + Y(IM, JX, KM) * WI * (1.0 - WJ) * WK + Y(IM, JM, KX)
     * WI * WJ * (1.0 - WK) + Y(IX, JX, KM) * (1.0 - WI) * (1.0 - WJ)
     * WK + Y(IX, JM, KX) * (1.0 - WI) * WJ * (1.0 - WK) + Y(IM, JX, KX)
     * WI * (1.0 - WJ) * (1.0 - WK) + Y(IX, JX, KX) * (1.0 - WI)
     * (1.0 - WJ) * (1.0 - WK)

Z(I, J, K) = Z(IM, JM, KM) * WI * WJ * WK + Z(IX, JM, KM)
     * (1.0 - WI) * WJ * WK + Z(IM, JX, KM) * WI * (1.0 - WJ) * WK
     + Z(IM, JM, KX) * WI * WJ * (1.0 - WK) + Z(IX, JX, KM)
     * (1.0 - WI) * (1.0 - WJ) * WK + Z(IX, JM, KX) * (1.0 - WI)
     * WJ * (1.0 - WK) + Z(IM, JX, KX) * WI * (1.0 - WJ) * (1.0 - WK)
     + Z(IX, JX, KX) * (1.0 - WI) * (1.0 - WJ) * (1.0 - WK)
```

where

IM ~ minimum i index of region
JM ~ minimum j index of region
KM ~ minimum k index of region
IX ~ maximum i index of region
JX ~ maximum j index of region
KX ~ maximum k index of region

I ~ i index of node
J ~ j index of node
K ~ k index of node

The number of calculations are reduced by using:

```
FA = WI * WJ          F4 = FA - F1
FB = WJ * WK          F5 = WK - FC - F2
FC = WI * WK          F6 = WJ - FB - F4
F1 = FA * WK          F7 = WI - FA - F3
F2 = FB - F1          F8 = 1 + FA - WI - WJ - F5
F3 = FC - F1
```

```
X(I, J, K) = X(IM, JM, KM) * F1 + X(IX, JM, KM) * F2 + X(IM, JX, KM) * F3
     + X(IM, JM, KX) * F4 + X(IX, JX, KM) * F5 + X(IX, JM, KX) * F6
     + X(IM, JX, KX) * F7 + X(IX, JX, KX) * F8

Y(I, J, K) = Y(IM, JM, KM) * F1 + Y(IX, JM, KM) * F2 + Y(IM, JX, KM) * F3
     + Y(IM, JM, KX) * F4 + Y(IX, JX, KM) * F5 + Y(IX, JM, KX) * F6
     + Y(IM, JX, KX) * F7 + Y(IX, JX, KX) * F8

Z(I, J, K) = Z(IM, JM, KM) * F1 + Z(IX, JM, KM) * F2 + Z(IM, JX, KM) * F3
     + Z(IM, JM, KX) * F4 + Z(IX, JX, KM) * F5 + Z(IX, JM, KX) * F6
     + Z(IM, JX, KX) * F7 + Z(IX, JX, KX) * F8
```

Similar equations are used for node generation in plane regions.

Curved boundaries are matched by adding a displacement vector to every node within a region which has a curved boundary. Figure 6 shows a typical curved boundary. Point 1 and point 2 are vertices of a region and point 3 lies somewhere along the curve. Subroutine SEQ2 reads commands to form a circular arc through three points using either a parabola or a circle. Subroutines PARABO and CIRCLE reduce this information to a set of constants which are stored until they are

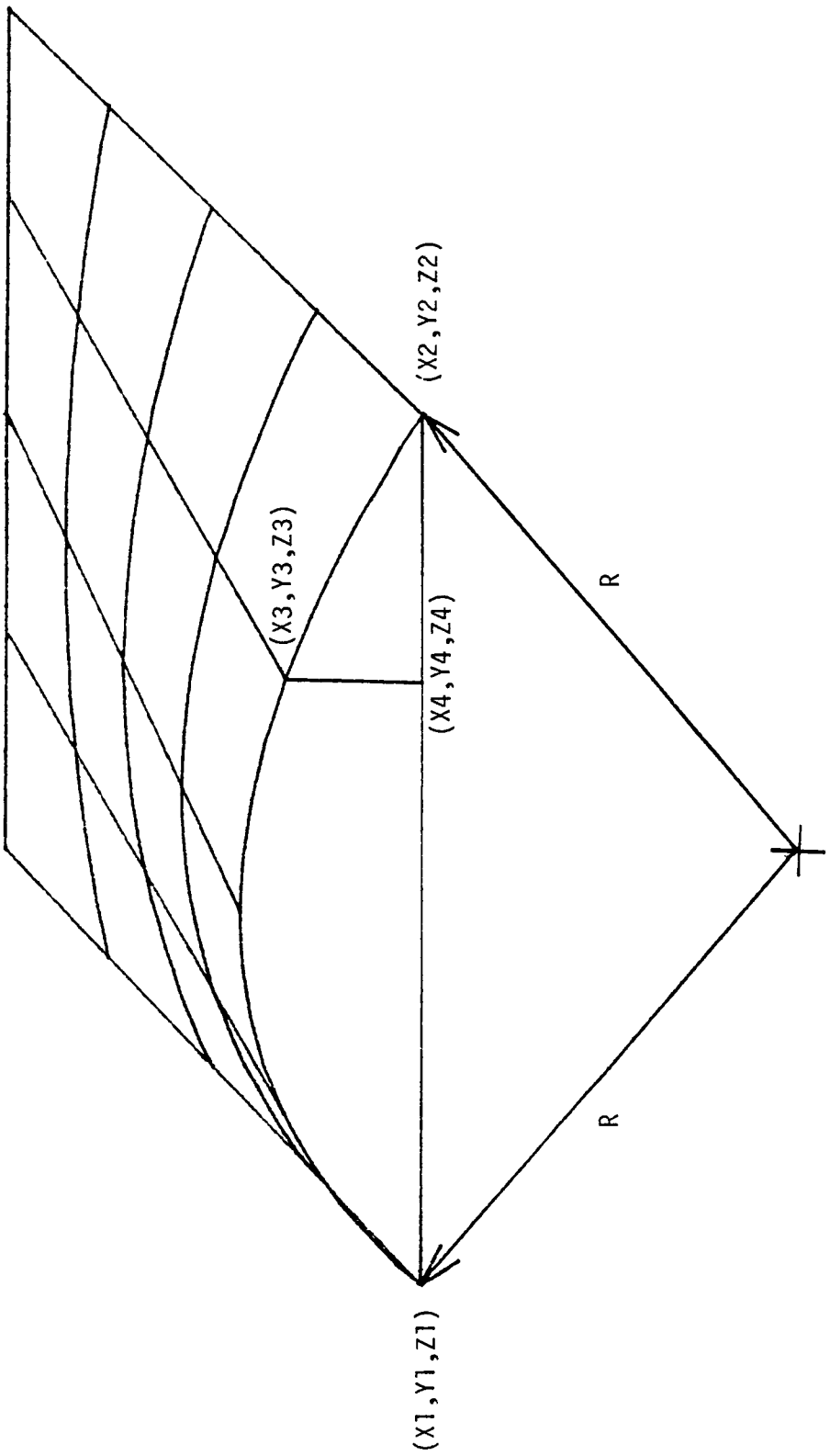


Figure 6. Forming a Curved Boundary

required in subroutine MOVE. These constants are used to determine the displacement vectors.

The equation for a parabola through three points is:

$$\begin{pmatrix} X \\ Y \\ Z \end{pmatrix} = \begin{vmatrix} X_1 & X_2 & X_3 \\ Y_1 & Y_2 & Y_3 \\ Z_1 & Z_2 & Z_3 \end{vmatrix} \cdot \begin{pmatrix} 3T - 1 - 2T^2 \\ 4(T - T^2) \\ 2T^2 - T \end{pmatrix}$$

where T varies from 0.0 at point 1 to 1.0 at point 2. The displacement vector is equal to zero at points 1 and 2 so the equation of the line through points 1 and 2 is subtracted from the parabola to give the displacement vector:

$$\begin{pmatrix} XV \\ YV \\ ZV \end{pmatrix} = \begin{vmatrix} X_1 & X_2 & X_3 \\ Y_1 & Y_2 & Y_3 \\ Z_1 & Z_2 & Z_3 \end{vmatrix} \cdot \left(\begin{vmatrix} 2T^2 + 1 - 3T \\ 4(T - T^2) \\ 2T^2 - T \end{vmatrix} - \begin{vmatrix} 1 - T \\ 0 \\ T \end{vmatrix} \right)$$

$$= \begin{vmatrix} X_1 & X_2 & X_3 \\ Y_1 & Y_2 & Y_3 \\ Z_1 & Z_2 & Z_3 \end{vmatrix} \cdot \begin{pmatrix} 2(T^2 - T) \\ 4(T^2 - T) \\ 2(T^2 - T) \end{pmatrix}$$

These equations are reduced as follows:

$$XV = (4X_3 - 2X_1 - 2X_2)(T - T^2) = C_1(T - T^2)$$
$$YV = (4Y_3 \quad 2Y_1 - 2Y_2)(T - T^2) = C_2(T - T^2)$$
$$ZV = (4Z_3 - 2Z_1 - 2Z_2)(T - T^2) = C_3(T - T^2)$$

$$C_1 = 4X_3 - 2X_1 - 2X_2$$
$$C_2 = 4Y_3 - 2Y_1 - 2Y_2$$
$$C_3 = 4Z_3 - 2Z_1 - 2Z_2$$

C_1, C_2, and C_3 are stored for later use in subroutine MOVE.

For a circular boundary, the constants stored are the unit vector normal to the line through points 1 and 2, the radius of the arc, and the distance from point 1 to point 2. These parameters are derived from Figure 7. Since angle θ equals angle A, sin θ = a/2R = sinA. Since

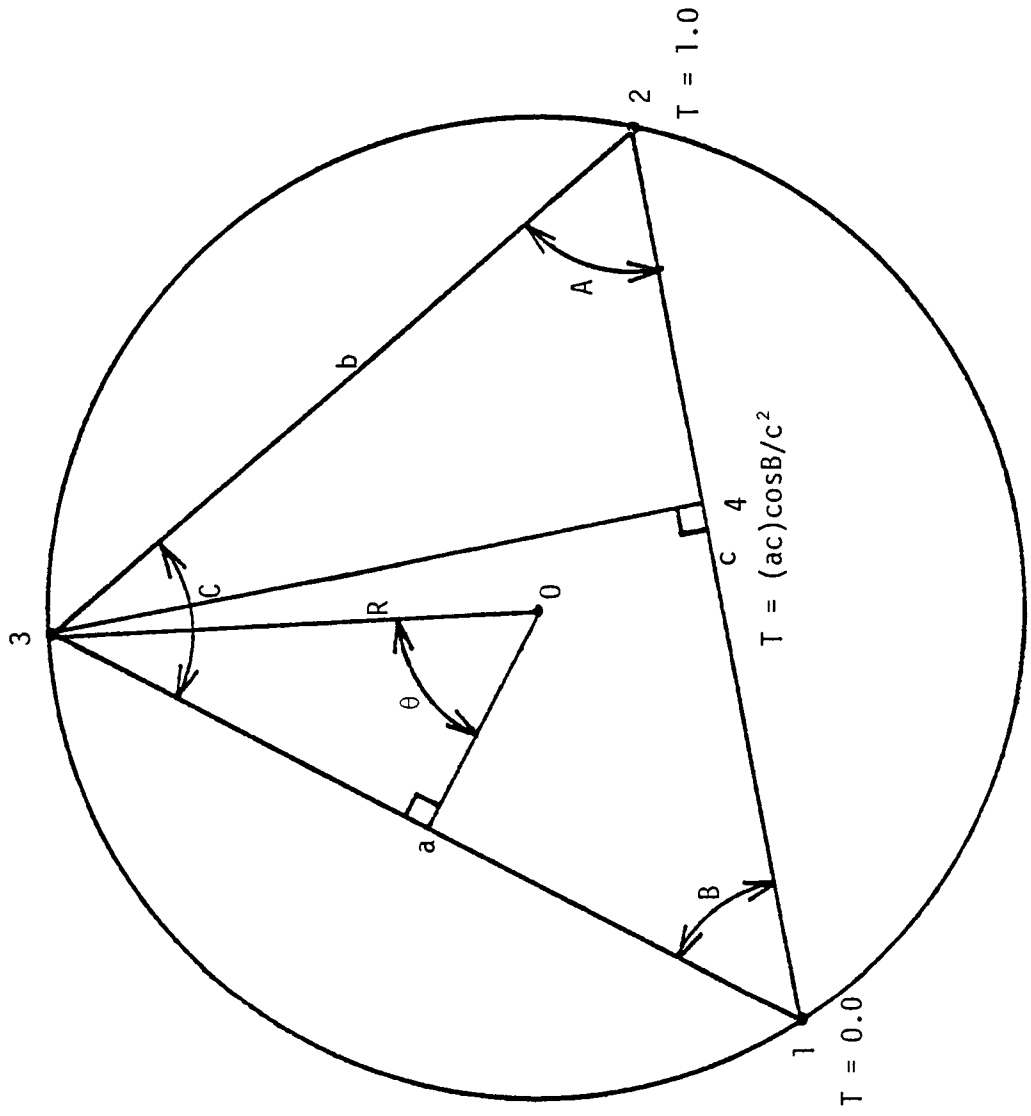


Figure 7. A Circular Boundary

AREA = bc $\sin^A/2.0$, b/2R = (2/ac)AREA or R = abc/(4.AREA). The area is determined using Hero's formula

$$\text{AREA} = \sqrt{s(s-a)(s-b)(s-c)} \quad \text{where } s = (a+b+c)/2.$$

The distance from point 1 to point 4 is

$$a\cos B = (ac)\cos B/c$$

The ratio of the distance from 1 to 4 divided by the distance from 1 to 2 is $(ac)\cos B/c^2$. From the law of cosines, $b^2 = a^2 + c^2 - 2ac\cos B$ therefore

$$\begin{aligned} T &= (ac)\cos B/c^2 \\ &= (a^2 + c^2 - b^2)/2c^2 \end{aligned}$$

and

$$\begin{aligned} X_4 &= (X_2 - X_1) * T + X_1 \\ X_4 &= (Y_2 - Y_1) * T + Y_1 \\ Z_4 &= (Z_2 - Z_1) * T + Z_1 \end{aligned}$$

$$\hat{\mu} = \frac{(X_3 - X_4)\hat{i} + (Y_3 - Y_4)\hat{j} + (Z_3 - Z_4)\hat{k}}{\sqrt{(X_3 - X_4)^2 + (Y_3 - Y_4)^2 + (Z_3 - Z_4)^2}}$$

The five constants, $\hat{\mu}$, R, and C are stored. Figure 8 shows how the point P is moved to point P' on the arc. The vector ys $\hat{\mu}$ is added to P. Using the parameter T, XS = C * (T - 0.5) and

$$D_1^2 = R^2 - XM^2, \quad D_2^2 = R^2 - XS^2$$

Therefore

$$YS = \sqrt{R^2 - XS^2} - \sqrt{R^2 - XM^2}$$

Each point throughout the regions adjacent to the arc is moved by a displacement vector. For a solid region, the displacement vector for a node depends on it's location in the regions as follows:

$$\overline{X}(I, J, K) = \overline{X}(I, J, K) + \overline{F}(T) * W\alpha * W\beta$$

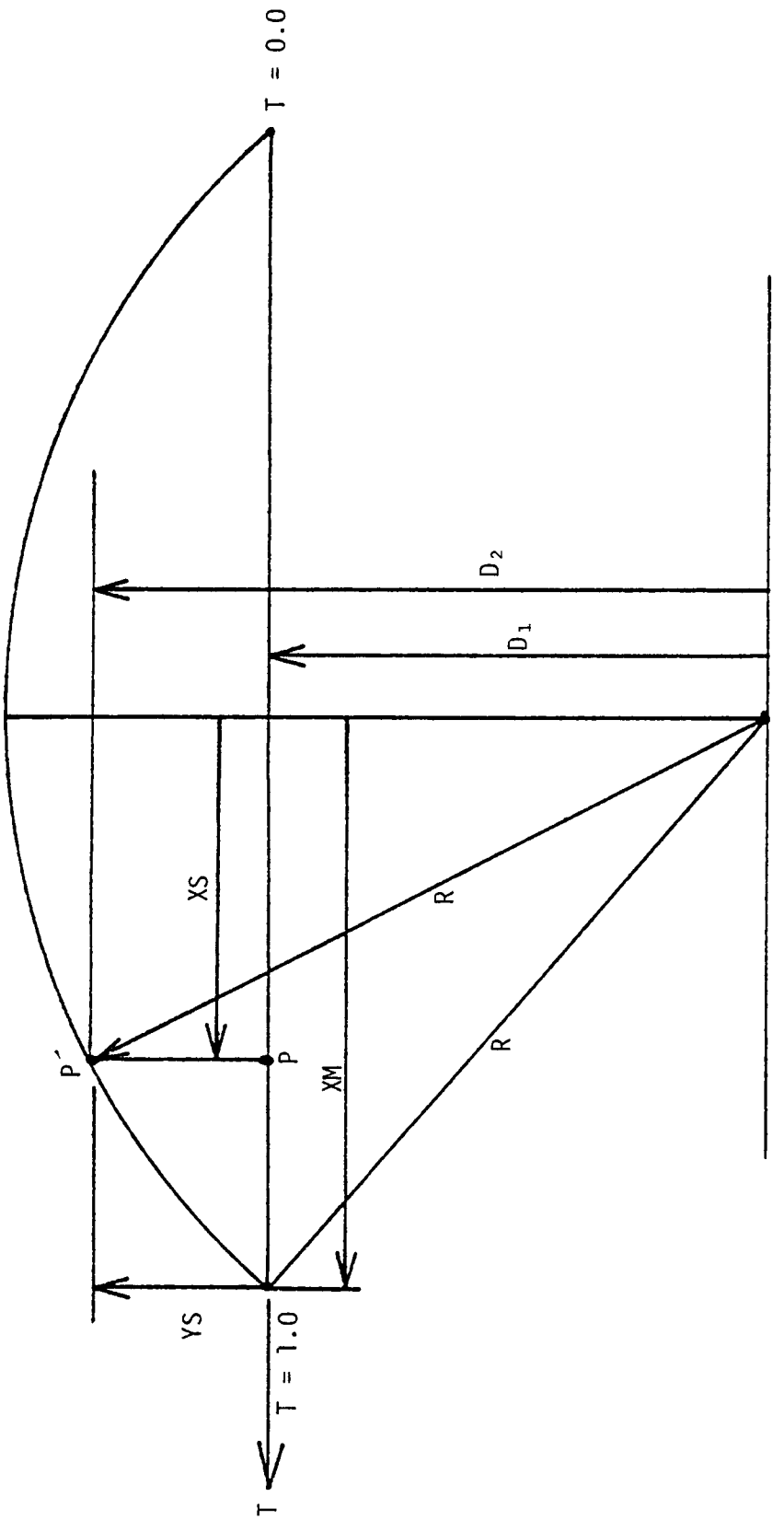


Figure 8. Forming a Circular Boundary

Where $\overline{F}(T)$ is the displacement vector for either a circle or a parabola, $W\alpha$ and $W\beta$ are scale factors determined by the distance along the axes perpendicular to T from the boundary to the point of interest as in Figure 9.

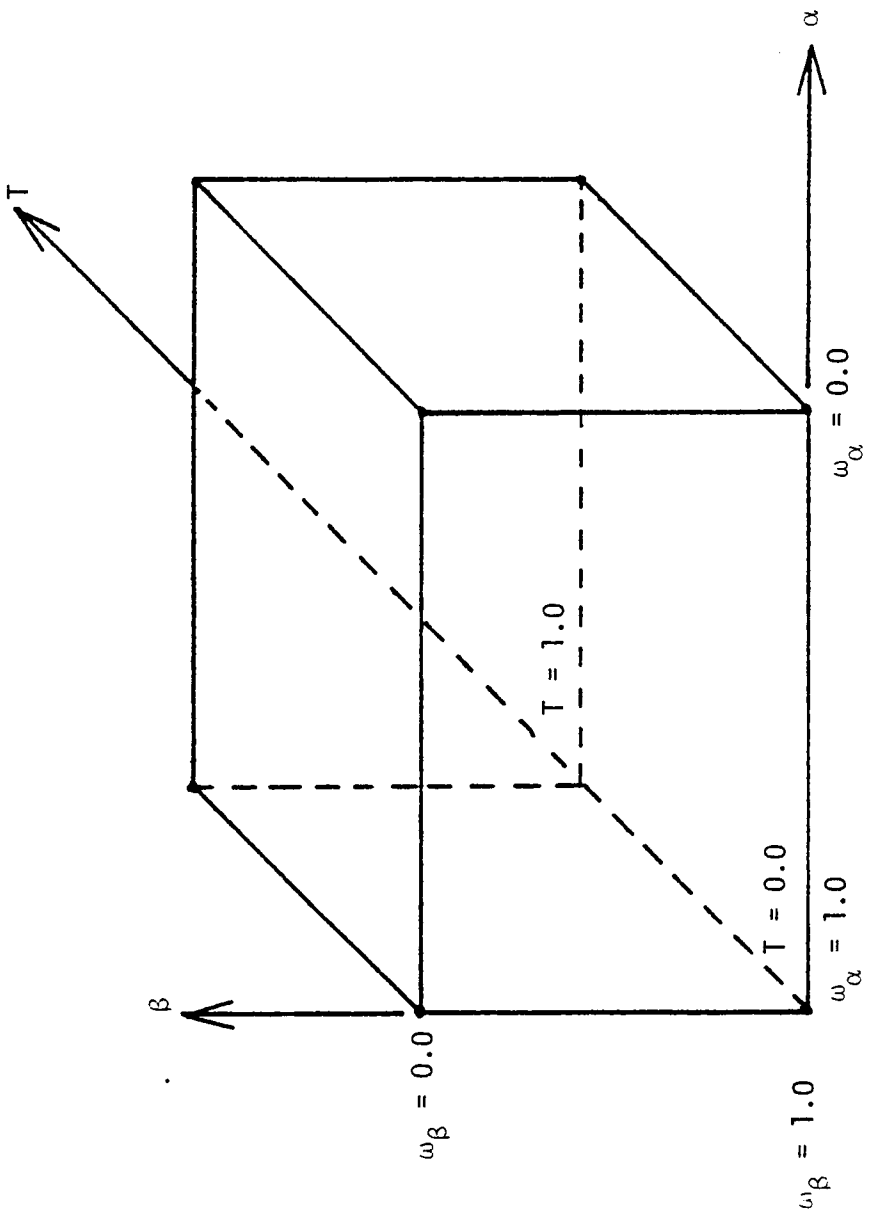


Figure 9. Scale Factors

V. STRUCTURE PLOTTING

Plotting is performed in two steps. First, a plot file is created during mesh generation which has a series of pen commands. Each command consists of three node numbers which represent the beginning, the midpoint, and the endpoint of an arc. Second, these commands are processed by subroutine INPLOT to produce the final plot.

Plot File Creation

The plot file is produced by subroutine PLOTMK. For every element that is generated, subrouting PLOTMK produces the pen commands which plot the edges of the element. Plate elements each have four edges and require four plot commands, and solid elements require twelve plot commands. Since edges can be adjacent to more than one element, the amount of plotting can be reduced by eliminating duplicate edges.

Figure 10(a) shows a region which has 16 elements and requires 40 plot commands. Since each plot command requires both a command to position the pen at the beginning of a line, and a command to draw a line, the plotting speed can be increased by using a special ordering of plot commands. Figures 10(b) and 10(c) show a series of plot commands which can be superimposed to give the plot in Figure 10(a). In this case the number of times that the pen is repositioned is reduced to ten while the ordering of the plot commands allows straight lines to be detected.

Each plot command consists of three node numbers which define beginning and end points of an element edge along with a midside node. For linear elements that do not have midside nodes, the midside node

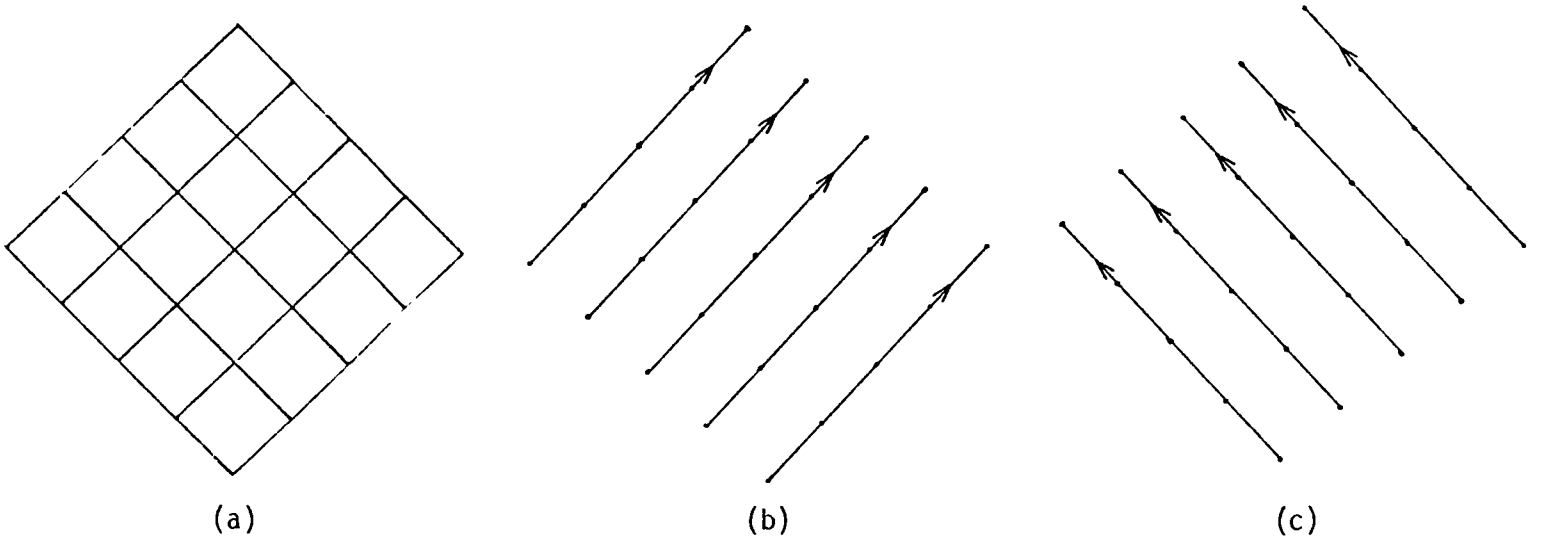


Figure 10. Plot Vectors

number is zero. If the midside node does not exist, then a straight line is drawn from the beginning to the end points. Otherwise, a parabole is plotted through the three node points.

Node Projection

The screen coordinates of nodes are determined by mapping the coordinates of an object on to a plane normal to the viewing direction. These coordinates are then scaled so that the plot will fit properly on a graphics terminal screen. The coordinate transformation used in INGRID is shown in Figure 11. The equations for this transformation are:

$$B = -X \sin\beta + Y \cos\alpha$$
$$A = -X \cos\beta \sin\alpha - Y \sin\beta \sin\alpha + Z \cos\alpha$$

Each node point is transformed using these equations. During mesh generation, the maximum and minimum coordinates for each part are stored by subroutine MAXMIN. This data is used to determine the scale factors for plotting. A list of parts is input by the user in subroutine PART. From this list of parts the maximum and minimum coordinates of the collection of parts are determined using the data created by MAXMIN. The parts are then enclosed in a sphere from which the final scaling equations are derived:

$$A = (A_i - A_o) \cdot \left(\frac{RU}{2R}\right) + C_A$$

$$B = (B_i - B_o) \cdot \left(\frac{RU}{2R}\right) + C_B$$

where

A_i, B_i = are the coordinates of a node to be plotted
A, B are the image of the scaled node

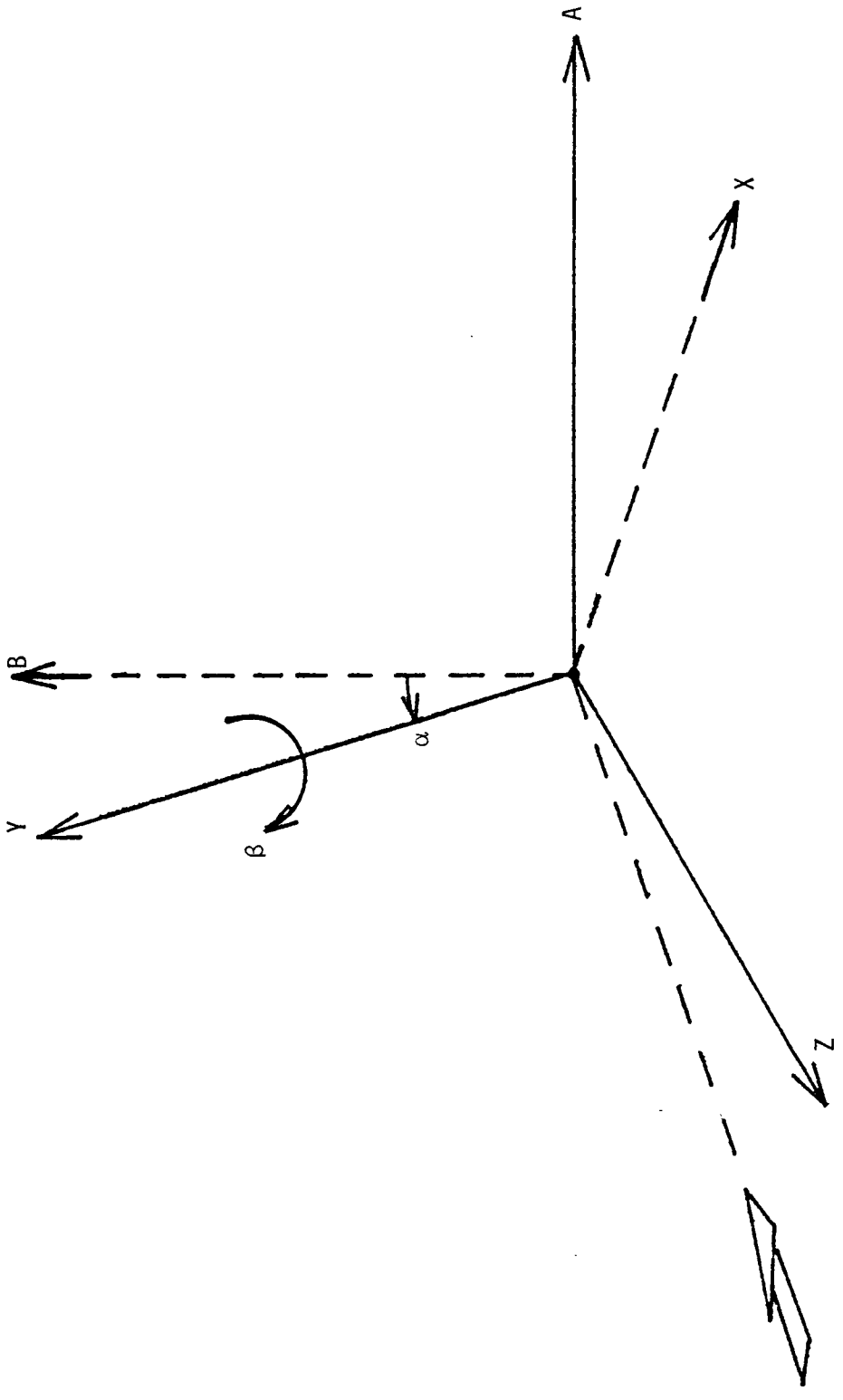


Figure 11. Project of Coordinates for Viewing

A_O, B_O are the coordinates of the center of the sphere containing the parts to be plotted

C_A, C_B are the coordinates of the center of the plotting device in raster units

R is the radius of the sphere which contains the parts to be plotted

Ru is the number of raster units that can be plotted on the particular device.

Projections are performed by subroutine MAP and the final scaling is performed in subroutine PLOT.

Plotting

Reading and processing of plot commands is performed by subroutine INPLOT. When a command is read, it is first compared with the parts list to determine if it should be plotted. Plot commands are then checked to determine if they are lines or curves. If the midside node number is zero then the command is a line. Subroutine FN2 determines if curves are approximately staight and changes the midside node number to zero if the three nodes are colinear.

Curves are divided into eight line segments. Each line segment is plotted in sequence using the following equation to determine the coordinates of intermediate points along the curve:

$$\begin{Bmatrix} A \\ B \end{Bmatrix} = \begin{vmatrix} A_1 & A_2 & A_3 \\ B_1 & B_2 & B_3 \end{vmatrix} \begin{Bmatrix} (s^2 - s)/2 \\ 1 - s^2 \\ (s + s)/2 \end{Bmatrix}$$

where

A, B is the intermediate node

A_1, B_1 is the starting node

A_2, B_2 is the midside node

A_3, B_3 is the end node

S is a parameter which varies from -1.0 to 1.0.

Subroutine FN1 determines the locations of the intermediate nodes while subroutine PLOT performs the final scaling and vector plotting.

Before a line is plotted, it is compared with the next plot command to determine if the two lines are colinear. If they are co-linear, the two commands are combined and the process is repeated. Otherwise, the old command is plotted and the new command is stores. The subroutines which perform the manipulation of line plotting commands are described in Table IV.

TABLE IV

PLOTTING SUBROUTINES

Subroutine	Description
PLTADD	Adds a plot command to the plot buffer
PLTAD2	Combines two plot commands in the plot buffer
PLTCHK	Checks if the next plot command and the command in the plot buffer are colinear
PLTDMP	Dumps the plot buffer

After all plot commands are processed, the appropriate files are initialized and the user is prompted for more commands.

VI. OUTPUT

The final processing of data is performed by subroutine OUTPUT. This includes removal of duplicate nodes, determining the necessary boundary conditions, flattening plates, and renumbering solid elements. Following is a description of the algorithms used for these problems.

Duplicate Node Removal

Removal of duplicate nodes involves comparing coordinates of each node with all other nodes to determine if any lie within a certain distance of each other. If the nodes are within a tolerance value, they are combined to form a single node. Since this process can be expensive, a special algorithm is used to minimize the cost.

Using the structure in Figure 12 as an example, the number of comparisons necessary to remove duplicate nodes is $NODES^2/2$ where NODES is the number of node points in the structure. If the structure has 2,000 nodes, then 2,000,000 comparisons must be performed. The number of comparisons can be reduced by dividing the structure as shown in Figure 12. A code which lies in the ith region needs to be compared only to the nodes in the (i-1) region, the (i) region, and the (i+1) region. If the nodes are distributed evenly throughout the structure and there are 99 divisions, then the total number of comparisons is approximately $NODES^2/66$. Again using a structure with 2,000 nodes, this would only require 61,000 comparisons.

To efficiently implement this algorithm required the use of additional arrays. Figure 13 shows a diagram of the required storage. IARRAY contains a list of node numbers which are ordered by the

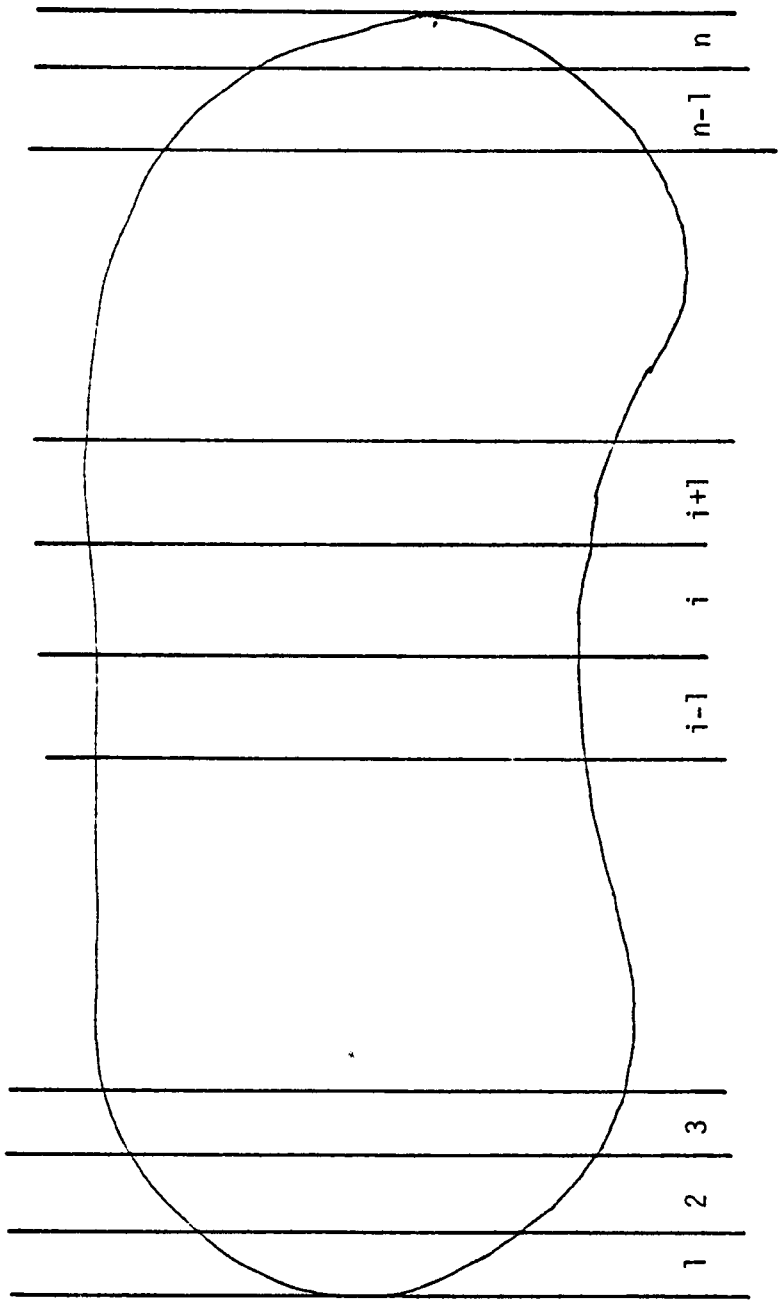


Figure 12. Division of an Object for Tolerancing

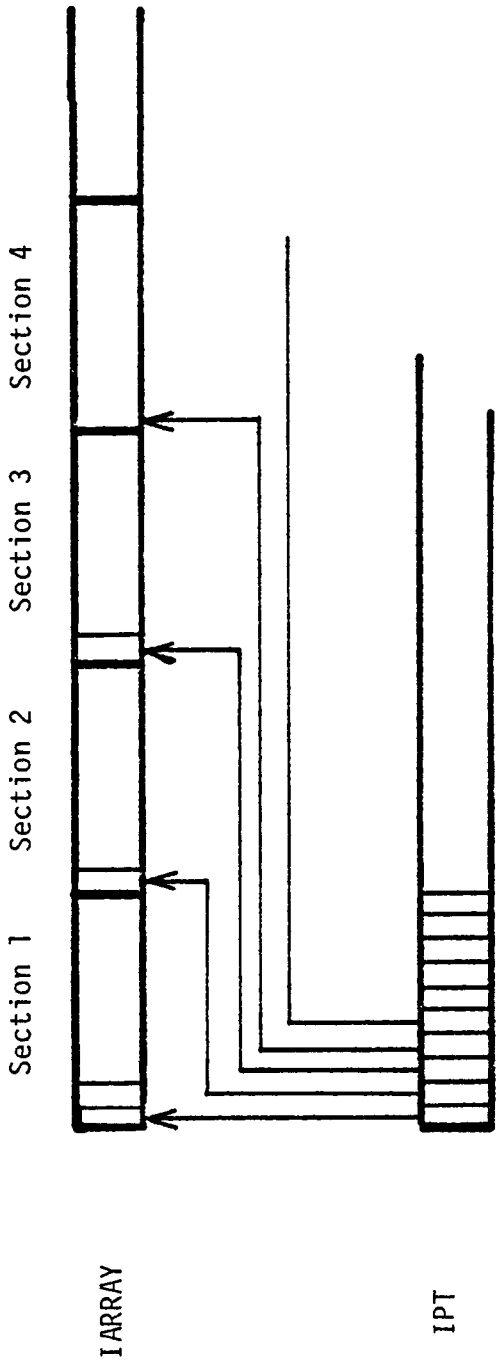


Figure 13. Storage Diagram for Tolerancing Routines

sections they occur in. All of the node numbers which occur in the first division of the structure are located at the beginning of IARRAY followed by the other sections. In order to determine where a section starts and stops, a second array, IPT, is also included which has pointers to the beginning of each section. Thus, to find the nodes adjacent to another node in the Ith region requires comparisons to be made on every node from location [(IPT(I -1)] to (IPT(I +2)-1) in IARRAY.

Plate Flattening

For quadratic shell elements generated over a curved surface the midside nodes lie out of the plane formed by the element vertices; however, it is frequently necessary that all of the nodes lie in the same plane. For a typical shell element such as in Figure 14 the midside nodes are moved normal to the element until they lie in the plane defined by nodes 1, 2, and 4. The unit normal is determined from the cross produce of $\bar{V}_1$ and $\bar{V}_2$:

$$\hat{U}_{12} = (\bar{V}_1 \times \bar{V}_2)/V_1V_2$$

The distance from any midside node to the plane of the plate is:

$$D = \bar{Q} \cdot \hat{U}_{12}$$

The final location of a midside node is:

$$\bar{X} = \bar{X} - D\ \hat{U}_{12}$$

Renumbering Solid Elements

Numbering solid elements properly is important since an improperly numbered element will result in a negative jacobian in a finite element program. Improper numbering of elements can be detected by

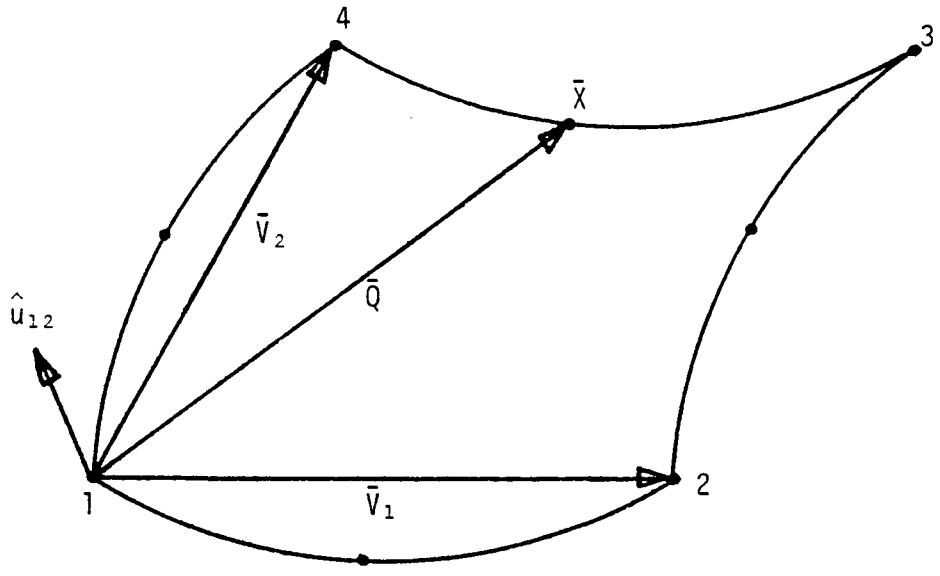


Figure 14. Flattening of Plate Elements

calculating the volume of the element. If the volume is negative, then the nodes are reordered to give a position volume. For the element in Figure 15, the volume is approximately determined by the scalar triple product of three vectors:

$$\text{Volume} = (\bar{V}_1 \times \bar{V}_2) \cdot \bar{V}_3$$

For the SAP 5 program all elements must have a positive volume; however, SAP6 and PAFEC both require elements to have a negative volume when using INGRID's numbering convention. Depending on the output type, all elements are numbered to have either a positive or a negative volume and then output in the specified format.

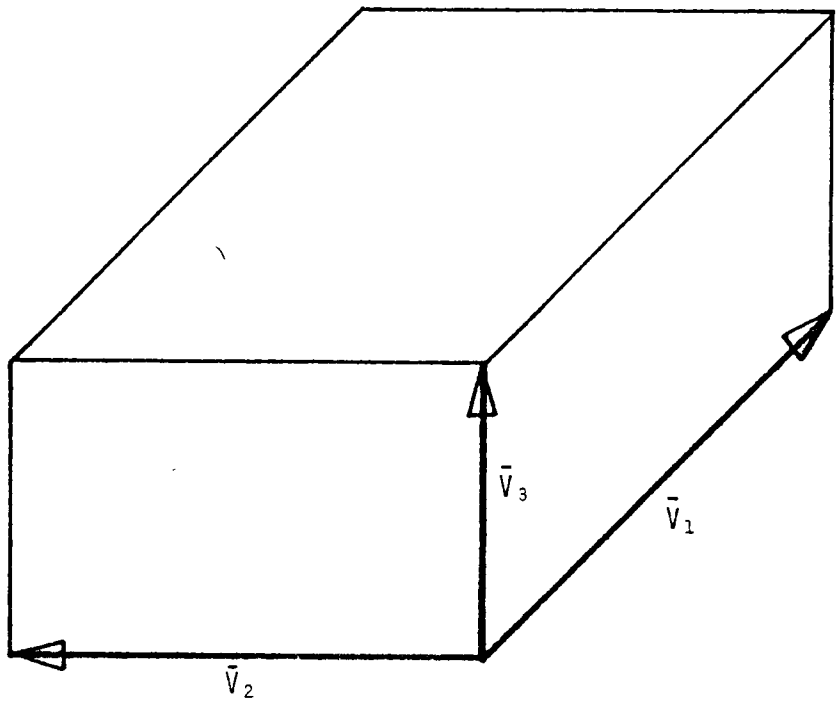


Figure 15. Calculating the Volume of a Solid Element

VII. CONCLUSION

INGRID has adequately demonstrated its ability to solve a wide variety of mesh generation problems; however, there are still many improvements which can be made in all aspects of the program. The improvements can be made in the areas of data input, index notation, mesh generation, functions, plotting, and post-processing of data. These changes are directed towards increasing both the generality and efficiency of the program in order to make it as widely applicable as possible. Following is a description of many of the possible improvements.

Index Notation

The index notation presented suffers from two limitations: triangular and pentahedral elements cannot be defined using the notation, and it is difficult to perform mesh grading. These problems can be overcome by some minor modifications to the index notation.

Index notation is most convenient for definition of regions such as shown in Figure 16(a). Figure 16(b) shows a region which would best be modeled using triangles. If nodes A and B of Figure 16(a) are assigned the same coordinates, the mesh becomes that of Figure 16(b) which is not only characterized by poor aspect ratios, but also has zero jacobians for elements adjacent to AB. Such conditions can be recognized by INGRID and appropriate routines called to produce triangular meshes such as in Figures 16(c) and 16(d).

Thus, the further convention can be added that if two vertices of a region which lie on the same edge have the same coordinates, then

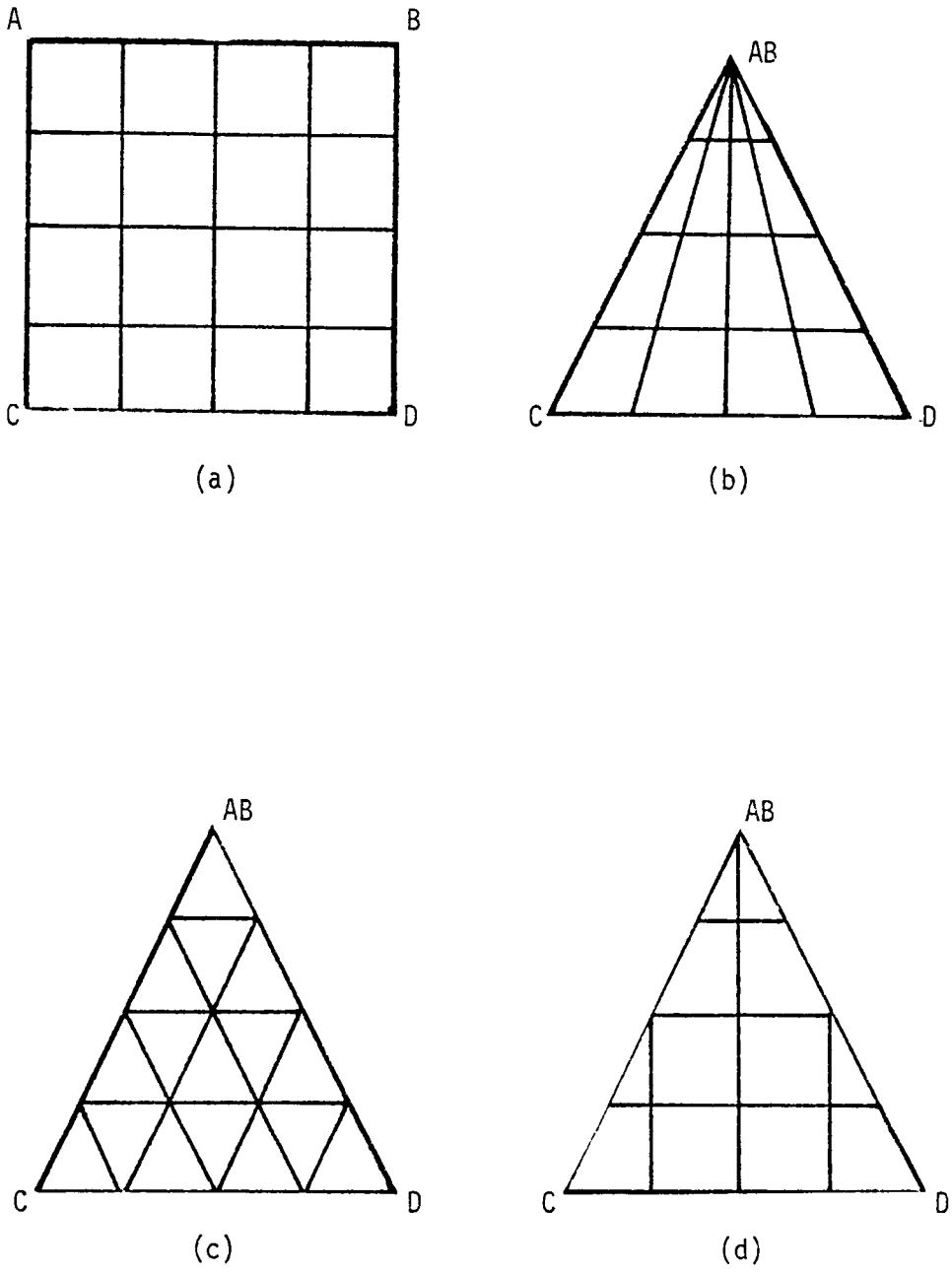


Figure 16. Triangular Element Generation

INGRID can be programmed to automatically model the region with triangular elements. Since assigning two vertices of a region the same coordinate is illegal using the present notation, this change would make the notation more general. The triangular region along with its edges and vertices can still be referred to by the index notation which insures the usefulness of this addition.

Similar conventions can be used for the representation of pentahedral wedge elements or pyramid elements. Illegal regions, such as occur if nodes A and D of Figure 16(a) were coincident, can also be deleted. An example of this is in Figure 17. Region BCEF is illegal and can be automatically deleted.

One of the most difficult problems in mesh generation is the problem of mesh grading. Imafuku, et al. suggest an algorithm for mesh grading which can be useful in connection with the index notation.[11] By adding a function which changes the number of elements along the i, j, or k direction, INGRID can recognize regions which require either mesh refinement or mesh grading. The appropriate routines can be called to perform the necessary mesh refinement or mesh grading. Again, it is useful to notice that this in no way restricts the present notation and only represents a generalization of the notation.

Functions

Much of the effectiveness of INGRID depends on choosing a complete set of functions which are both general enough to define all the necessary information for a finite element model and simple enough to be comprehended by the user. Further generalization of the functions in INGRID can meet this goal. Although the functions now used are

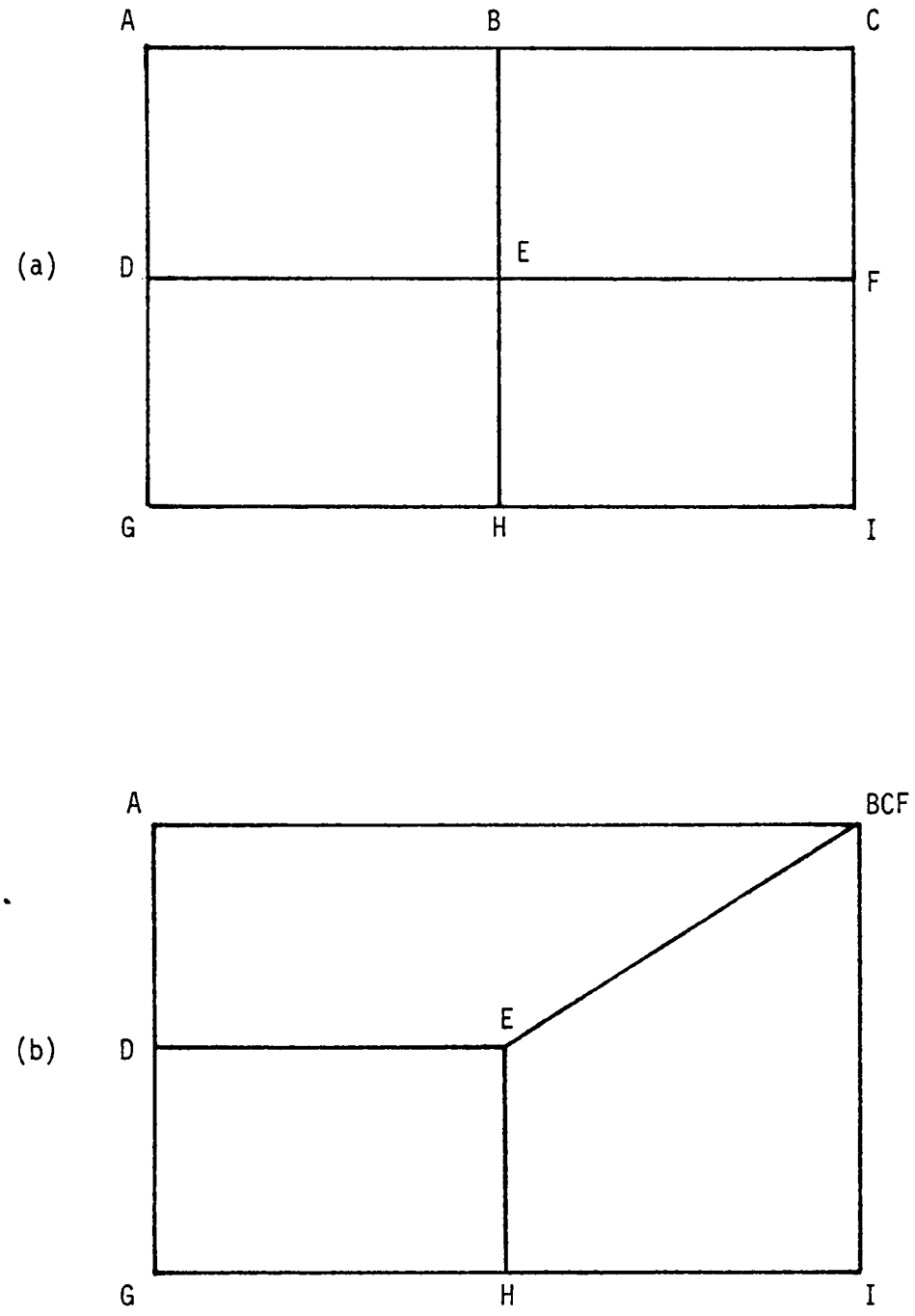


Figure 17. Automatic Region Deletion

completely separate, they have many features in common which can be used to combine the functions into a very simple system. One possible function which requires a special free format is of the form:

(index specification)/command = parameter, parameter,

. . . /command = . . .

examples:

(2, 3, 1, 4, 5, 2)/MATERIAL = STEEL
or (1, 0, 5)/X = 9.0/Y = 6.0
or (1, 3, 5, 7, 4, 6)/DELETE/CYLINDER = I/PRESSURE = 50.0
(put a circular hole in the structure and apply a 50 lb pressure)

As can be seen from the examples, the user only needs to have a knowledge of index notation and a list of keywords. The notation is both concise and readable.

Plotting

A very helpful feature of INGRID is the ability to quickly plot a grid for verification. When a user is viewing a complex grid at a slow baud rate, much time is wasted. If iterations need to be performed on the plot, then this situation becomes even worse.

For this reason, every effort has been made to decrease the number of vectors plotted; however, substantial improvements still can be made to decrease the plotting costs even further. INGRID decreases the plotting time for some problems by determining which vectors within a region of the index space lie on a straight line. Straight lines frequently cross several regions. By determinining which regions are adjacent, the plot file can be arranged to find straight lines which cross several regions. This can provide savings of more than 50% for plotting many meshes.

Since the speed of pen plotters is generally a function of the total distance travelled by the pen, substantial savings can be achieved by adjusting the sequence of pen commands. Figure 18(a) shows the normal sequence of pen commands. Moving the pen from the end of one line to the beginning of the next requires almost as much time as drawing a line; however, in Figure 18(b) a different sequence is presented which can decrease the plotting time. This sequence can be programmed into the plot file generator with no extra CPU time required while decreasing the total plotting time from 30% to 45% for the average problem.

Other useful features which can be included are node and element numbering and a hidden line algorithm for plotting curved surfaces. A hidden surface algorithm can also be made extremely efficient by taking advantage of adjacencies in the index space.

Data Base

A permanent data base is useful in many aspects of computer aided design. Figure 19 shows a typical problem which might require several design iterations and can take advantage of a data base. Since INGRID is capable of producing complex primitives such as are required to model the pipe joint accurately, the user should be able to store these in a permanent data base which can be accessed with only the necessary parameters. Modifications to INGRID which allow a permanent data base can be made to increase the applications to computer aided design.

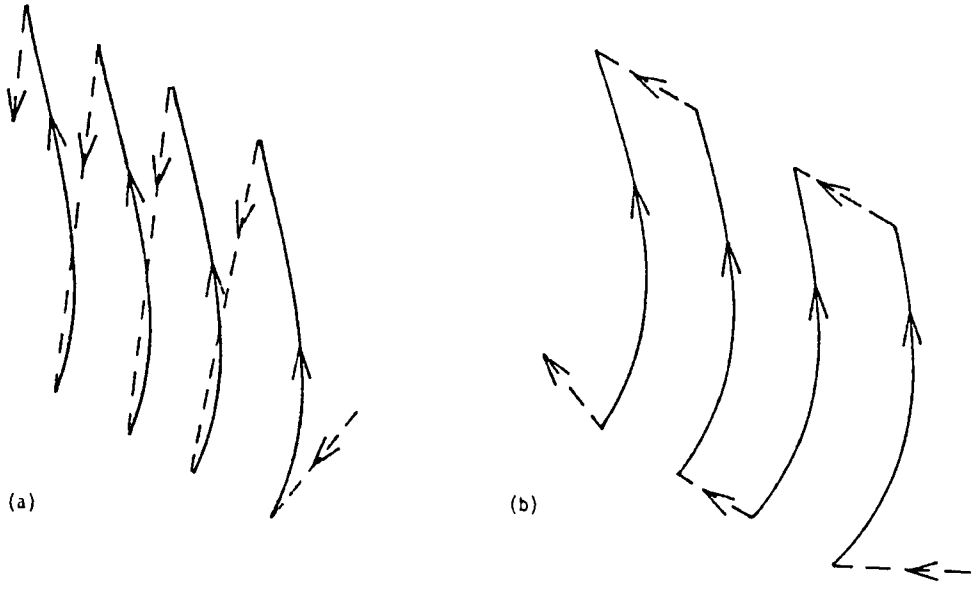


Figure 18. Plotting Sequences

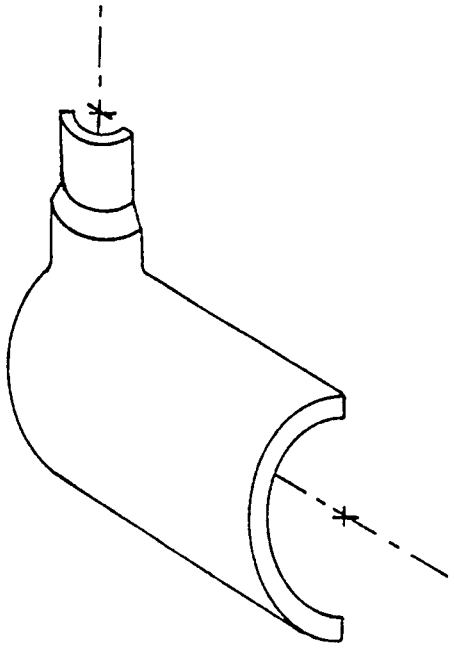


Figure 19. Pipe Joint

LIST OF REFERENCES

LIST OF REFERENCES

1. Thacker, W. C. "A Brief Review of Techniques for Generating Irregular Computational Grids." International Journal for Numerical Methods in Engineering, Vol. 15, pp. 1335-1341, 1980.

2. Cook, William A. "INGEN: A General Purpose Mesh Generator for Finite Element Codes." Los Alamos Scientific Laboratory Report LA-7135-MS, 1978.

3. "SapV2: A Structural Analysis Program for Static and Dynamic Response of Linear Systems." SAP User's Group, University of Southern California, 1977.

4. "SAP6-2: A Structural Analysis Program for Static and Dynamic Analysis." SAP User's Group, University of Southern California, 1979.

5. "PAFEC 75: Data Preparation," Pafec Ltd., 1978.

6. Christiansen, Hank and Stephenson, Mike. "MOVIE.BYU: A General Purpose Computer Graphics Display System." Brigham Young University, 1976.

7. Becker, Eric and Brisbane, John. "Application of the Finite Element Method to Stress Analysis of Solid Propellane Rocket Grains." Rohm and Haas, Redstone Arsenal Research Division Report S-76, Vol. 1, 1965.

8. Cook, William A. "Body Oriented (Natural) Coordinates for Generating Three-Dimensional Meshes." International Journal for Numerical Methods in Engineering, Vol. 8, pp. 27-43, 1974.

9. Gerhard, Michael. "SLIC: The Interactive, Graphic Mesh Generator." Lawrence Livermore National Laboratory, 1980.

10. Park, S. and Washam, C. J. "Drag Method as a Finite Element Mesh Generation Scheme." Computers and Structures, Vol. 10, pp. 343-346, 1979.

11. Imafuku, I., Kodera, Y., and Sayawaki, M. "A Generalized Automatic Mesh Generation Scheme for Finite Element Method." International Journal for Numerical Methods in Engineering, Vol. 15, pp. 713-731, 1980.

APPENDICES

APPENDIX A

INDEX NOTATION

INGRID is based on the index space notation which is used in INGEN. Since the index space notation is not completely adequate for manipulating geometries, another notation, the "Index Progression," is derived from the index space. Index progressions provide a concise and simple method for describing complex structures, and are used to input data to INGRID. Following is a detailed description of the index space notation and the index progression. This information provides the user with the concepts necessary to use INGRID effectively.

Index Space

Node generation in INGEN is done by a mapping from Index space onto the object of interest as is shown in Figure A.1. Each region of the object is referenced by a set of six indices. (IMIN, JMIN, KMIN) specify the minimum indices for a region in the index space and (IMAX, JMAX, KMAX) specify the maximum indices. For a solid region, all eight corner nodes must be defined by the combinations of minimum and maximum indices. Table A.1 lists the indices of the vertices in the example of Figure A.1. We assume that any set of three indices, (I, J, L), defines a point; and any set of six indices, (IMIN, JMIN, KMIN, IMAX, JMAX, KMAX), defines a region in space.

If KMIN is set equal to KMAX, the resulting region is a plane of constant K as shown in Figure A.2(a). Similarly, a plane of constant I is defined when IMIN is set equal to IMAX and a plane of constant J for

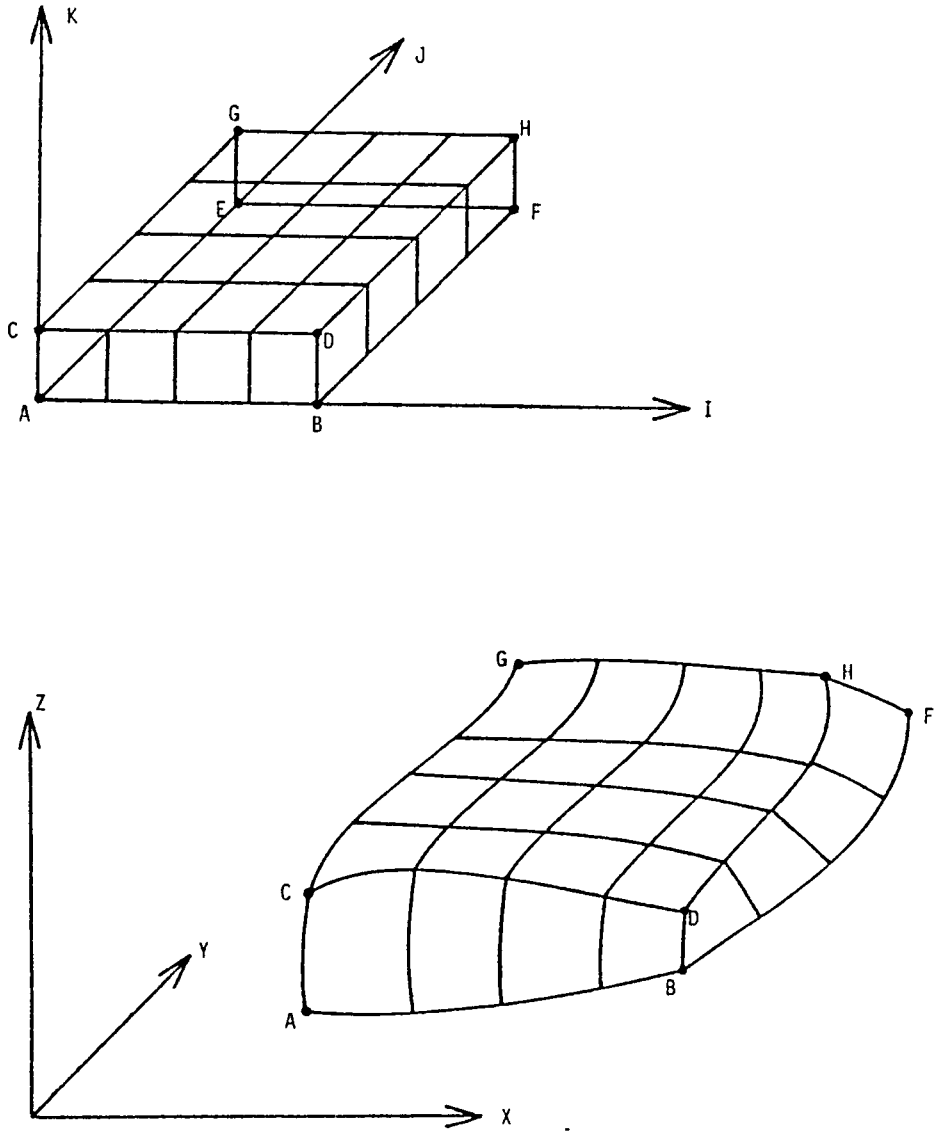


Figure A.1. Mapping from Index Space to Object Space

JMIN equal to JMAX. A line in the index space is defined by holding two indices constant while the third index varies as shown in Figure A.2(b).

TABLE A.1

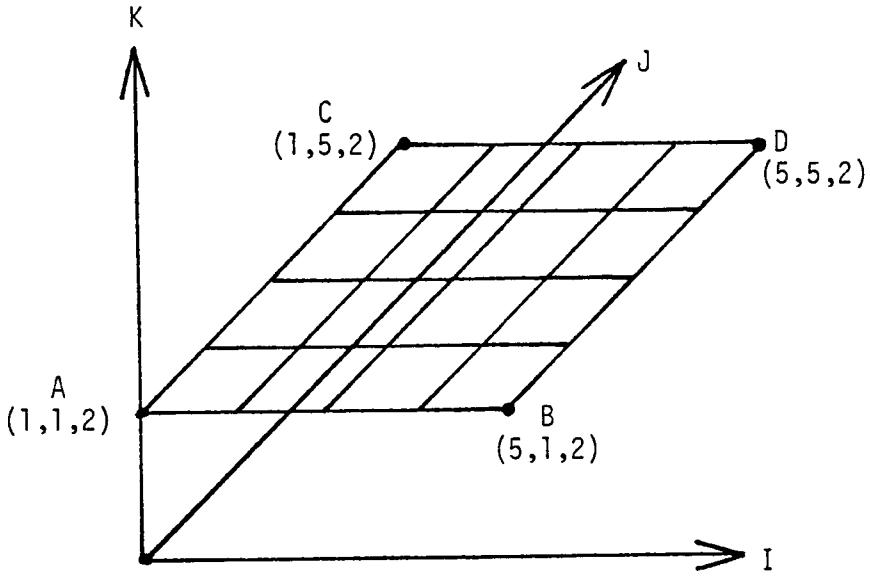
INDICES ASSOCIATED WITH THE VERTICES OF A REGION

Node	Indices	Position
A	(1, 1, 1)	(IMIN, JMIN, KMIN)
B	(5, 1, 1)	(IMAX, JMIN, KMIN)
C	(1, 1, 2)	(IMIN, JMIN, KMAX)
D	(5, 1, 2)	(IMAX, JMIN, KMAX)
E	(1, 5, 1)	(IMIN, JMAX, KMIN)
F	(5, 5, 1)	(IMAX, JMAX, KMIN)
G	(1, 5, 2)	(IMIN, JMAX, KMAX)
H	(5, 5, 2)	(IMAX, JMAX, KMAX)

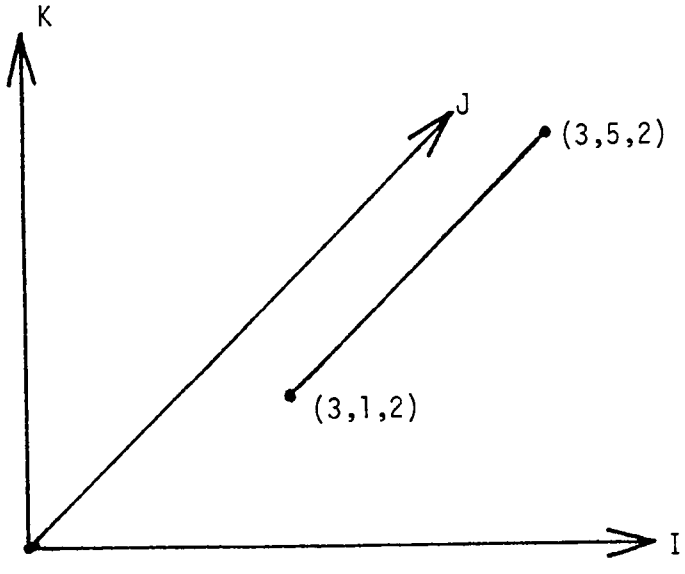
An index space is defined as the set of all possible indices, $1 \leq I \leq \infty$, $1 \leq J \leq \infty$, $1 \leq K \leq \infty$. If an index is zero, then it varies over all possible indices. Thus, the indices (3, 0, 2) defines a line which extends across the index space, and (0, 0, 2) defines a plane which divides the index space into two regions. (0, 0, 0) defines the entire index space.

Index Progressions

Index progressions were developed to facilitate the defining of multiple regions in index space. Rather than specifying the minimum and



(a)



(b)

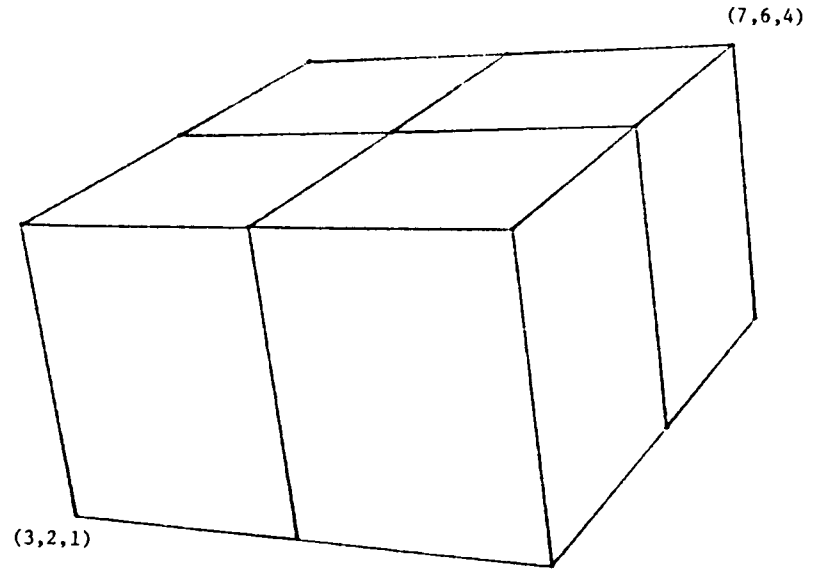
Figure A.2. Planes and Lines in Index Space

maximum indices for a region, one simply specifies the progression of indices along the I, J, and K directions respectively. For example, the region (2, 7, 6, 8, 9, 10) is represented as the progression {(2, 8); (7, 9); (6, 10)}. If there is a region adjacent to (2, 7, 6, 8, 9, 10) such as (2, 7, 2, 2, 7, 6) the two regions are defined together by a new progression {(2, 8); (7, 9); (2, 6, 10)}. To define the four solid regions shown in Figure A.3(a) requires the progression {(3, 5, 7); (2, 4, 6); (1, 4)}.

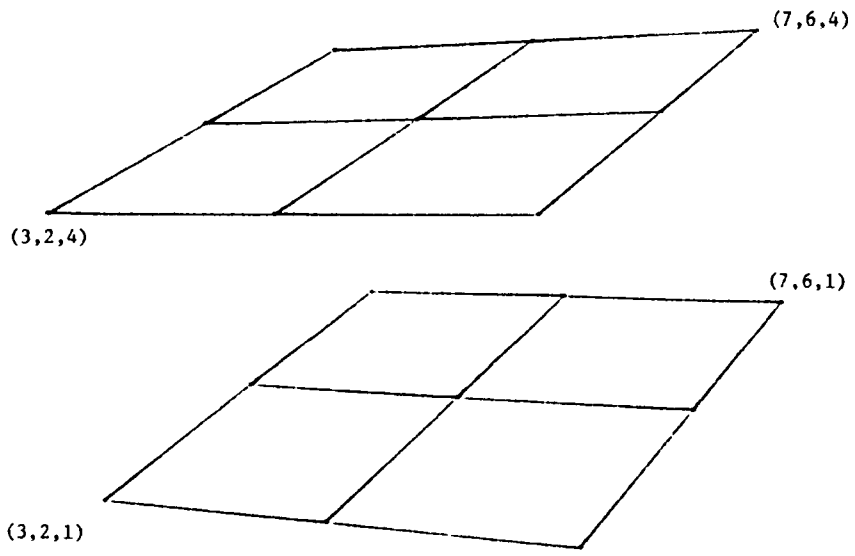
Index progressions for planes are defined in a similar manner. The index which remains constant throughout a plane is indicated by a negative sign so the plane (2, 5, 5, 2, 7, 8) is represented as {(-2); (5, 7); (5, 8)}. A parallel plane such as (4, 5, 5, 4, 7, 8) can be added to this progression by adding another i index to give {(-2, -4); (5, 7); (5, 8)}. In Figure A.3(b) there are eight planes which can be represented by the progression {(3, 5, 7); (2, 4, 6), (-1, -4)}. The savings by this notation is apparent since specifying separately the eight regions in Figure A.3(b) requires 48 numbers whereas the index progression requires only 8 numbers.

Another addition to the index progression notation is the zero index. The two solid regions shown in Figure A.4 could be represented as an index progression except that they are not connected. In this case a zero index is used along the i progression to indicate that the structure is discontinuous. This gives the progression {(2, 4, 0, 6, 8); (3, 7); (4, 5)}. Plane regions can be separated by the zero index in a manner similar to solid regions.

More complicated regions can now be represented by combining index progressions. An example of this is in Figure A.5. The open box



(a) Index Space



(b) Object Space

Figure A.3. Index Progressions for Planes and Solids

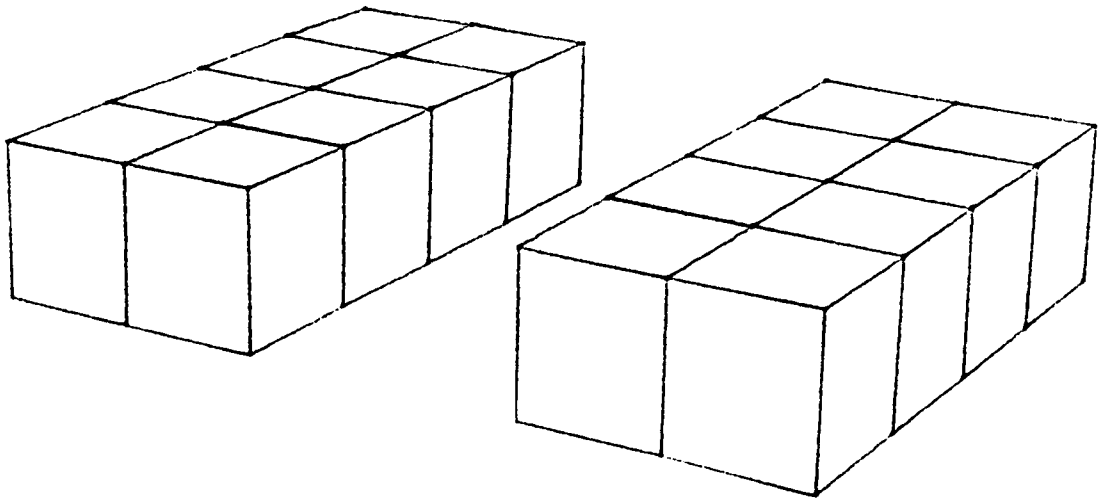


Figure A.4. Separated Solid Regions

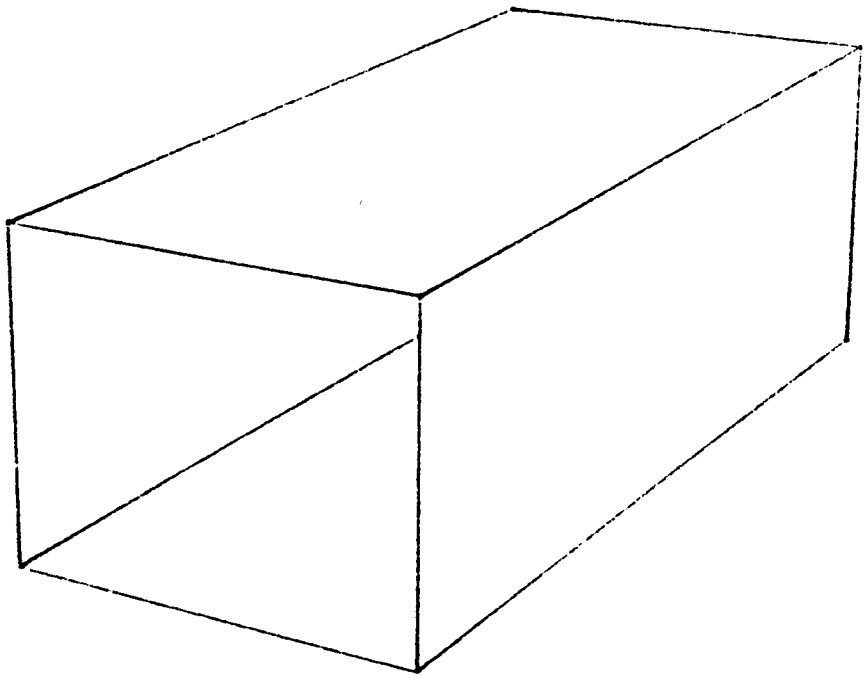
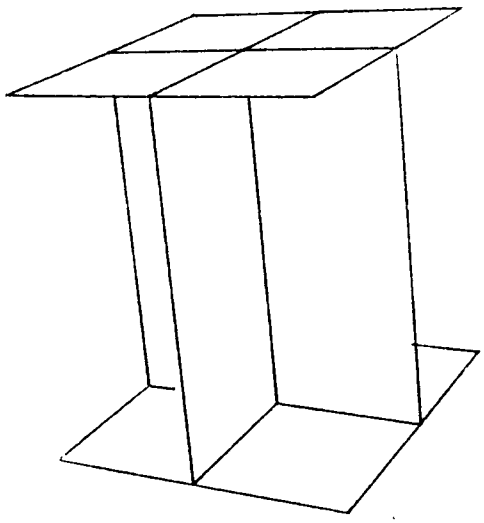


Figure A.5. Open Box

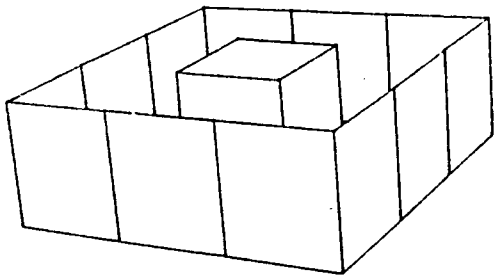
could be represented by two index progressions {(-2, -5); (1, 7); (3, 5)} and {(2, 5); (1, 7); (-3, -5)}, but they can also be combined to give {(-2, -5); (1, 7); (-3, -5)}. Figure A.6 shows several more structures and their index progression representation.

INGRID uses the index progression notation to set up regions in the index space which are to be mapped on to the object of interest. This notation has the advantage that it requires very little input data and with less than 20 indices can represent thousands of configurations in index space. In practice, not all configurations in index space can be defined by an index progression so a command is added to INGEN to allow deletion of regions in the index space. The delete command along with the index progression is enough to produce almost any conceivable region in the index space and is used as the central part of INGEN's mesh generation.



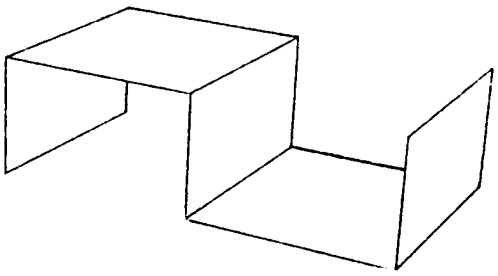
(a) Intersecting Plates

{(2, -4, 6); (2, -4, 6); (-3, -7)}



(b) Cube in a Box

{(-2, 4, 6, -8); (-2, 4, 6, -8), (3, 5)}

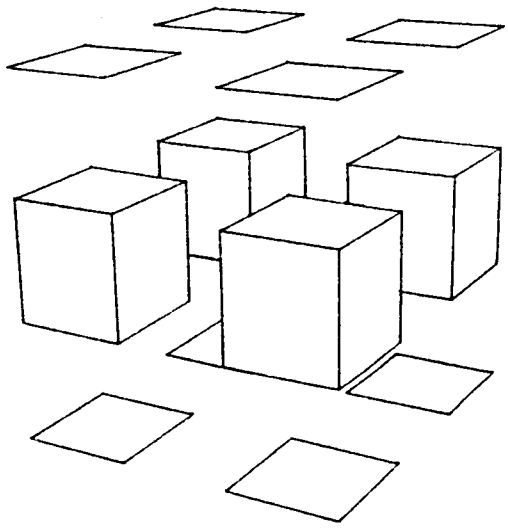


(c) Examples of
Region Deletion

{(-2, -6, -10); (3, 7); (-2, -4)}

Deleted Regions:

(2, 3, 2, 6, 7, 2) and (6, 3, 4, 10, 7, 4)



(d) Planes and Solids
with Gaps

{(2, 4, 0, 6, 8); (2, 4, 0, 6, 8); (-2, 4, 6, -8)}

Figure A.6. Examples of Index Progressions

APPENDIX B

DATA PREPARATION

This section describes the data required as input to INGRID. All data is input using standard FORTRAN format specifications. The data file consists of a title card, a series of control cards, and a series of "mesh generation modules."

1. Title Card (20A4)

   This card is required at the beginning of every data file.

2. Control Cards (A4, 6X)

   The following control cards are allowed.

AXISymmetric	MOVIE
BEAMs	PAFEC
COORdinates	PLOTs
C (Comment)	SAP5
EXCHange	SAP6
FAST	TOLErance
LOOF	

Keywords must be between four and ten characters except for comments which only require a 'C' in column 1. Control cards are optional and can be input in any order.

(a) AXIS (A4, 6X)

Output specifications are for axisymmetric elements.

(b) BEAM (A4, 6X)

This command is followed by a series of node cards and a series of element generation cards.

i. Node cards

columns 1 Constraint flag
EQ. ; (blank) node is free
EQ.F; node is constrained

2 - 5 Node number

6 - 15 X-coordinate

16 - 25 Y-coordinate

26 - 35 Z-coordinate

As many node cards are input as necessary.

A BLANK CARD TERMINATES NODAL POINT INPUT.

ii. Element Cards

columns 1 - 5 Beginning node number

6 - 10 Ending node number

11 - 15 Number of elements to be generated (Nodes are also generated)

16 - 20 Material property number

21 - 25 Node which defines direction of the principal bending axes.

As many element cards are input as necessary.

THIS SECTION ENDS WITH A BLANK CARD.

(c) COOR

columns 11 - 15 numer of local coordinate systems to be input.

This command generates local coordinate systems. Each local coordinate system is defined by three nodes, as shown in Figure B.1. The first point locates the origin of the local coordinate system, the second point lies along the local x-axis, and the third point lies anywhere on the positive x-y plane.

For each local coordinate system, a set of three cards must be input.

Card 1 (location of point 1) (A1, 4X, 3F10.0)

columns	1	Flag describing coordinate system EQ. ; (blank) cartesian (X, Y, Z) EQ.C; cylindrical coordinates (R, Θ, Z) EQ.S; spherical coordinates (R, Θ, ϕ)
	6 - 15	Global X(R) coordinate of point 1
	16 - 25	Global Y(Θ) coordinate of point 1
	26 - 35	Global Z(Θ, ϕ) coordinate of point 1

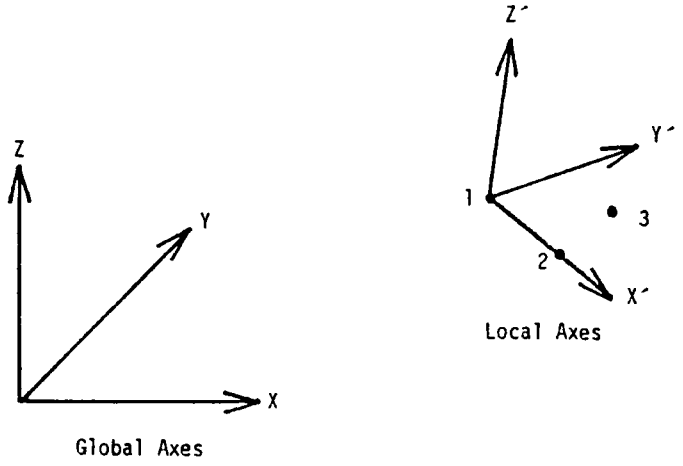


Figure B.1. Coordinate Transformation

Card 2, 3 Similar to card 1 for points 2 and 3.

NO FURTHER INPUT IS REQUIRED FOR THIS SECTION.

(d) Comments (A1)

Comments may be used to document the data file; however, there are certain cases where they can not be used. They can not be used after a BEAM or COOR and before the next

command. There are no restrictions when comments are used with mesh modules.

(e) EXCH (A4, 8X, 311)

This command performs a permutation of the coordinate axes and is performed before any other coordinate transformations.

columns	description
13	Flag for changing the x coordinate
	EQ.1; This coordinate becomes the x coordinate
	EQ.2; This coordinate becomes the y coordinate
	EQ.3; This coordinate becomes the z coordinate
14	Flag for changing the y coordinate
15	Flag for changing the z coordinate

(f) FAST (A4)

Only data necessary for plotting is generated. No output is generated.

(g) LOOF (A4)

Plate elements do not lie in a plane. If this command is not used, midside nodes will be moved to the plane of the plate or shell element.

(h) MOVIE (A4)

Output for the MOVIE.BYU program is generated. This output must be further processed by the utility program, MOVE.

(i) PAFEC (A4)

Output is generated which is compatible with the PAFEC finite element computer program.

(j) PLOT (A4)

This instructs INGRID to enter the interactive plotting package.

(k) SAP5 (A4)

Output is generated which is compatible with the SAP5 structural analysis computer program.

(l) SAP6 (A4)

Output is generated which is compatible with the SAP6 structural analysis computer program.

(m) TOLE (A4, 6X, F10.0)

columns	description
11 - 20	tolerance value for combining duplicate nodes

3. Mesh Generation Modules

Mesh generation modules instruct INGRID to perform the actual mesh generation. Each mesh generation module consists of the following data:

"START"	3.
Module Control Cards	3.1
Index Progression	3.2
Function Cards	3.3
"END"	

"START" signifies the beginning of a mesh generation module and is required as the first card in each module. Module control cards affect properties of the mesh defined in the module. Following is a list of the default properties for a mesh generation module:

4 node plate elements
8 node solid elements
cartesian coordinates
material property = 1
plate thickness = 0.0

The index progression is defined in appendix A. The dimensions of the index space along with all plane and solid regions are defined by the index progression. Function cards manipulate the mesh defined by the index progression and a "END" signifies the end of a module.

Following are some important definitions in addition to those in Appendix A.

Index space. The set of all indices defined by an index progression. For example, the progression {(2, 3, -5, 10) (4, (4, 5), (2, 6)} defines the index space 2<I<10, 4<J<5, 2<K<6.

Reduced index space. The reduced index space references positions in an index progression. The point I, J, K in the reduced index space refers to the point in the index progression defined by the Ith integer in the I-progression, the Jth integer in the J-progression and the Kth integer in the K-progression. For the progression {(2, 3, -5, 10), (4, 5), (2, 6)} the relationship between the reduced index space and the index space is shown in Table B.1.

Points and regions are defined in Appendix A. Unless otherwise noted, all points and regions are defined in the reduced index space. Since the reduced index space is independent of the actual values of the index progression, the mesh can be refined

or contracted by only changing the index progression. Points are always input with (2X, I3, 2I5) format and regions are input using (2X, 6I2) format.

TABLE B.1

COMPARISON OF THE REDUCED INDEX SPACE AND THE INDEX SPACE FOR THE INDEX PROGRESSION {(2, 3, -5, 10), (4, 5), (2, 6)}

Reduced Index Space	Index Space
1, 1, 1	2, 4, 2
1, 1, 2	2, 4, 6
1, 2, 1	2, 5, 2
2, 2, 2	3, 5, 6
3, 1, 2	5, 4, 6
4, 1, 1	10, 1, 1

Module Cards. The following module controls cards are allowed.

8NODe	REPEat
20NOde	SPHErical
CYLIndrical	THICkness
MATErial	

Keywords must be between four and ten characters. All control cards are optional and can be placed in any order after START and before the index progression.

(a) 8NODE (A4)

Eight node plate elements are generated. All indices are doubled to provide the extra indices necessary for the eight node elements.

(b) 20NODE (A4)

Twenty node solid elements are generated. All indices are doubled.

(c) CYLI (A4, 6X, 15)

Nodes are converted from cylindrical to cartesian coordinates.

columns	description
15	Flag for specifying type of conversion EQ. ; (blank) Conversion is performed after generation is completed. EQ.1; Conversion occurs before mesh generation is performed.

The equations for this transformation are:

$$X = R \cos \Theta$$
$$Y = R \sin \Theta$$

(d) MATE (A4, 6X, I5)

columns	description
11 - 15	Material property number

Material property numbers are output with the element data in the specified format.

(e) REPEAT (A4, 6X, 15)

columns	description
11 - 15	Local coordinate system number

The REPEAT command makes copies of a mesh module using different coordinate systems. If the local coordinate system number is zero, then the global axes are used. Otherwise the number refers to the coordinate system defined by the "COORDINATES" command. The first REPEAT command only changes the default coordinate system to a local coordinate system, thus it does not produce two copies of the module. For every additional REPEAT command one more copy of the module is generated.

(f) THIC (A4, 6X, F10.0, I5)

columns	description
11 - 20	Plate thickness (for SAP5)
21 - 25	Plate thickness number (for SAP6)

The thickness of plate elements within a module are specified using this command and are output with the element data for the specified format.

Index Progression. The mesh within each mesh generation module is defined by an index progression which is described in Appendix A.

CARD 1: I-progression (16I5)

columns	description
1 - 5	1st index of I-progression
6 - 10	2nd index of I-progression

As many as 16 indices may be placed on this card.

CARD 2: J-progression (16I5)

Similar to CARD 1.

CARD 3: K-progression (16I5)

Similar to CARD 1.

Cards 1, 2, and 3 are required for every module. Only the integers which are necessary for the progression should be input and the remainder of the entries are blank.

Function Cards. For each module a function is required to define the initial location of vertices. This function assigns an X-coordinate to each integer of the I-progression, a Y-coordinate to each integer of the J-progression, and a Z-coordinate to each integer of the K-progression.

CARD 1: X-coordinates (8F10.0)

columns	description
1 - 10	X-coordinate associated with first I-index
11 - 20	X-coordinate associated with first I-index

Eight X-coordinates may be placed on this card.

CARD 2: X-coordinates (8F10.0)

Note: This card is used only if the I-progression has more than 8 integers.

CARD 3 and 4: Y-coordinates (8F10.0)

Similar to CARD 1 and 2.

CARDS 5 and 6: Z-coordinates (8F10.0)

Similar to CARD 1 and 2.

Cards 1, 3, and 5 are always required.

Optional Functions. All optional functions are input after section 3.3.a and have the following form:

Keyword - index specification - parameters

INGRID has the following keywords for function cards:

keyword	description
b. blank	point function
c. I, J, K	A coordinate is a function of the I, J, or K index
d. A	arc card
e. B	boundary condition card
f. D	delete
g. M	midside node deletion

(b) Point Function (2X, I3, 2I5, I5, 3F10.0)

columns	description
1	blank
3 - 15	Indices of point
16 - 20	Flag for coordinates of point EQ. ; (blank) X, Y, and Z coordinates are modified EQ.1; X coordinate is modified EQ.2; Y coordinate is modified EQ.3; Z coordinate is modified EQ.12; X and Y coordinates are modified EQ.13; X and Z coordinates are modified EQ.23; Y and Z coordinates are modified EQ.123; X, Y, and Z coordinates are modified
21 - 30	X-coordinate if required
31 - 40	Y-coordinate if required
41 - 50	Z-coordinate if required

(c) Coordinate Modifying Function (A1, 1X, I3, 3I5, 6F10.0)

columns	description
1	Flag specifies independent variable for the function EQ.I; Coordinates vary as a function of the I-index EQ.J; Coordinates vary as a function of the J-index EQ.K; Coordinates vary as a function of the K-index
3 - 15	Indices of point (The index of the independent variable must be zero)
16 - 20	Flag specifies which coordinate is modified EQ.1; X-coordinate is modified EQ.2; Y-coordinate is modified EQ.3; Z-coordinate is modified
21 - 30	New coordinate for first index
31 - 40	Coordinate for second index

Six coordinates may be placed on this card.

CARD 2: New coordinates (8F10.0)

This card is required only if the independent variable has more than six values in the index progression. Eight coordinates may be placed on this card.

CARD 3: New coordinates (2F10.0)

This card is required only if the independent variable has more than 14 values in the index progression. Two coordinates may be placed on this card.

(d) Arc card (A1, 1X, 6I2, I2, 4X, 4F10.0)

Type 1: Curved Boundaries

columns	description
3 - 14	Indices of region (the region must be a line of length 1 in the reduced index space)
15 - 16	Flag specifying type of curve EQ.1; A parabola through point P1 (see Figure B.2) EQ.2; A circular arc through point P1 EQ.3; A circular arc with center P2

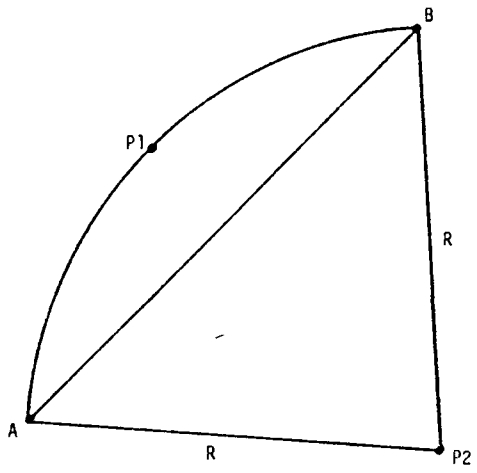


Figure B.2. Curved Boundaries

columns	description
21 - 30	X-coordinate of point P1 or P2
31 - 40	Y-coordinate of point P1 or P2
41 - 50	Z-coordinate of point P1 or P2
51 - 60	Radius (optional)

If the radius is input for a circular arc with center P2, then nodes A and B are moved radially from P2 until they are a distance equal to the radius from P2. An arc is formed through the nodes at their final location.

Type 2: Cylindrical Regions

columns	description
3 - 14	Indices of region (see Figure B.3. The region is a plane or a solid in the reduced index space with a width and height of 1 and an arbitrary length).
15 - 16	Flag specifying axis of rotation in the index space. EQ.1; I-axis is axis of rotation EQ.2; J-axis is axis of rotation EQ.3; K-axis is axis of rotation
51 - 60	Radius (optional)

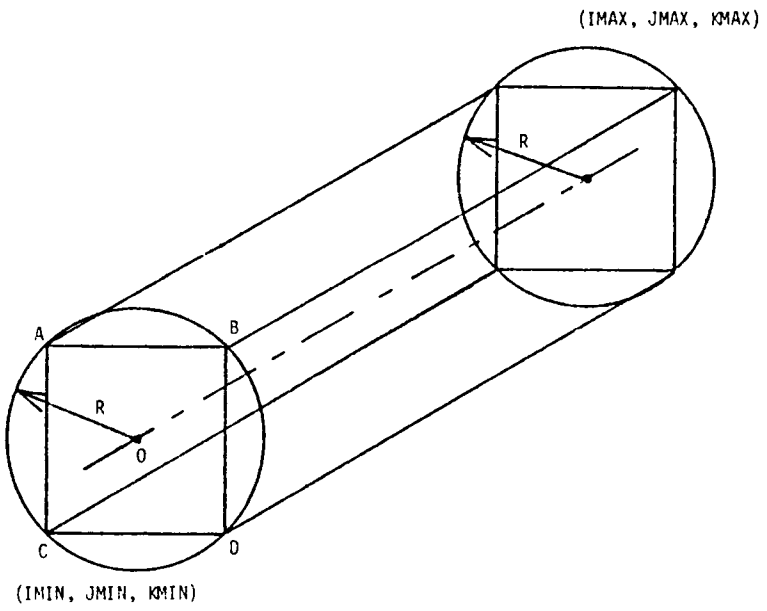


Figure B.3. Cylindrical Region

For any plane normal to the axis of rotation such as ABCD in Figure B.3, a point O on the axis of rotation is located in the center of the plane. If the radius of the cylinder is input, then the points A, B, C, and D are moved radially from O until they are a distance, R, from point O. Curved boundaries are then formed for the segments AB, BD, AC, and CD using center O. This is done for each plane normal to the axis of rotation in the reduced index space.

(e) Boundary Condition Cards (A1, IX, 6I2, 6I2)

columns	description
3 - 14	Indices of the region
15 - 16	Flag for constraining X displacement EQ.0; free EQ.1; fixed
17 - 18	Flag for constraining Y displacement
19 - 20	Flag for constraining Z displacement
21 - 22	Flag for constraining X rotation
23 - 24	Flag for constraining Y rotation
25 - 26	Flag for constraining Z rotation

(f) Region Deletion (A1, 1X, 6I2)

columns	description
3 - 14	Indices of region to be deleted

(g) Midside Node Elimination (A1, 1X, 6I2, 3I2)

columns	description
3 - 14	Indices of region
15 - 16	Flag for deletion of midside nodes along the I-direction EQ.0; Midside nodes are retained EQ.1; Midside nodes are deleted
17 - 18	Flag for deletion of midside nodes along the J-direction
19 - 20	Flag for deletion of midside nodes along the K-direction

APPENDIX C

EXECUTION

Input to INGRID consists of a disk data file and a series of interactive commands. The input data file is described in detail in Appendix B. INGRID also creates several scratch data files and output files which are listed in Table C.1.

TABLE C.1

DATA FILES USED BY INGRID

FORTRAN Unit Number	Description
5	Terminal
6	Job Statistics
7	Scratch (Nodes)
8	Scratch (Plate Elements)
9	Scratch (Solid Elements)
10	Scratch (Beam Elements)
11	Output (Nodal Points)
12	Input Data File
13	Scratch
14	Output (Elements)
15	Output (Boundary Conditions)
20	Scratch (Plotting)
21	Output (MOVIE.BYU)

When INGRID is executed it first reads the input data file and generates the mesh. If the PLOTS command is in the input file, INGRID will enter the interactive plotting package as soon as mesh generation is completed. The plotting package accepts commands to produce plots and commands to modify the output format. Following is a list of the allowable commands.

Plotting Commands	Output Modification Commands
CONTINUE	FLAT
END	LOOF
HELP	MOVIE
INFORMATION	NOMOVIE
PARTS	NO OUTPUT
PLOT	PAFEC
ROTATE	SAP5
SCALE	SAP6

Description of plotting commands

CONTINUE - Plotting is terminated and the final processing of the model will begin.

END - Stop execution of INGRID.

HELP - Print out the available commands.

INFORMATION - Some job statistics are printed.

PARTS - This command is used to specify whether all or some of the parts will be plotted. INGEN automatically assigns part numbers as the model is being generated. The portion of the model which is generated after the first 'BEAM' card or 'START' is assigned part number 1. After this, new part numbers are assigned for each

'START' that is encountered in the input data. When the 'PARTS' command is entered, INGRID responds with "< # of PARTS >." If the whole structure is to be plotted enter a 0. Otherwise enter the number of parts which are to be plotted and then the part numbers in free format.

PLOT - Plot the mesh. Two carriage returns are required.

ROTATE - This command is used to rotate the structure so that it can be viewed from any direction as shown in Figure 11, page 30. After typing this command INGRID requests the number of the vertical axis, angle α, and angle β which are input in free format.

The scale command allows the user to input a scale factor to expand or decrease the size of the plot on the screen. INGRID responds with "< SCALE FACTOR? >." The new scale factor is then input in free format.

scale factor	effect
= 1.0	No effect (default value)
> 1.0	The mesh is expanded
< 1.0	The mesh is decreased

Description of Output Modification Commands

FLAT - Nodes for plate elements must all lie in the same plane. INGRID attempts to put midside nodes in their proper position.

LOOF - Nodes for plate elements do not have to lie in the same plane. This is the opposite of FLAT.

MOVIE - Output for the MOVIE.BYU program is generated.

NOMOVIE - Output for the MOVIE.BYU program is not generated.

NOOUTPUT - No output for SAP5, SAP6, or PAFEC is generated

MOVIE output is generated if requested.

PAFEC - output compatible with PAFEC is generated.

SAP5 - Output compatible with SAP5 is generated.

SAP6 - Output compatible with SAP6 is generated.

Utility Program MOVE

Program MOVE reads the output from unit 21 of INGRID AND converts the data to a format compatible with MOVIE.BYU. The data files required by this program are listed in Table C.2. Besides reformatting data, MOVE also removes any faces which are adjacent to two solid elements and could not possibly be seen by a hidden line algorithm. This extra processing can save much computation when running MOVIE.BYU.

TABLE C.2

DATA FILES USED BY MOVE

FORTRAN Unit Number	Description
6	INGRID output
	MOVE input
10	MOVE output (MOVIE input)
21	MOVE input (INGRID output)

APPENDIX D

EXAMPLES

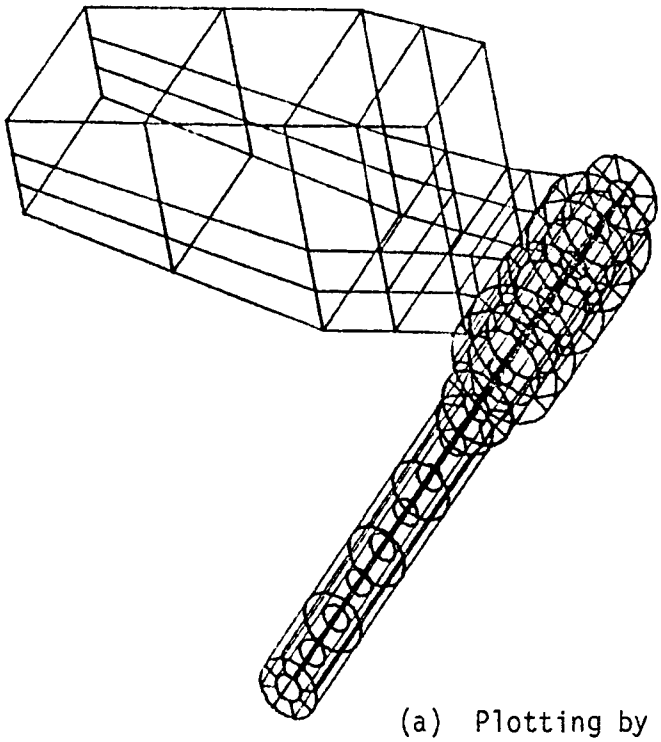
Demonstration Model

The demonstration model is shown in Figure D.1. It is chosen to demonstrate most of the mesh generation and plotting capabilities of INGRID. This model is generated from six separate mesh generation modules as is shown in Figure D.2. These modules are copied together to produce the final structure.

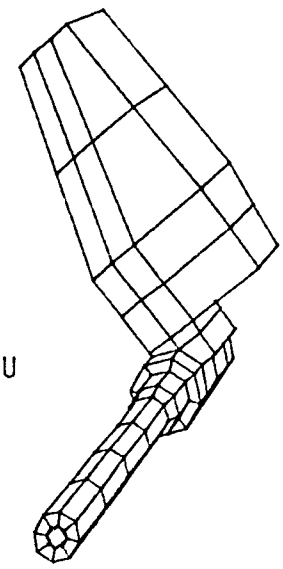
Part 1 is formed by generating a plane of solid elements and performing a cylindrical coordinate transformation to make a cylinder. In contrast to this is part 6 which is also a cylinder made of solid elements, however it is formed by moving the nodes of a unit square bar to their proper locations. Parts 2 and 3 demonstrate how complicated intersections of plates can be made with minimal effort. Parts 5 and 6 are formed by a coordinate transformation similar to the one used for part 1.

Mirror Fixture

Figure D.3 shows the mirror fixture. The key shaped regions are the mirrors which lie on a circular base. Each mirror is connected to the base by three pin connections. This model is chosen to demonstrate the ability of INGRID to make multiple copies of an object and to place each one in a different coordinate system.

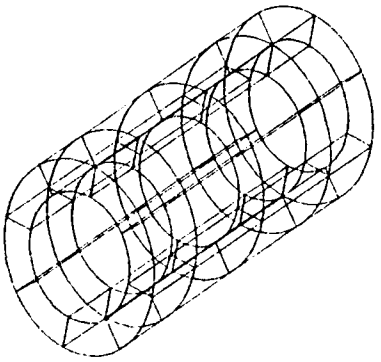


(a) Plotting by INGRID

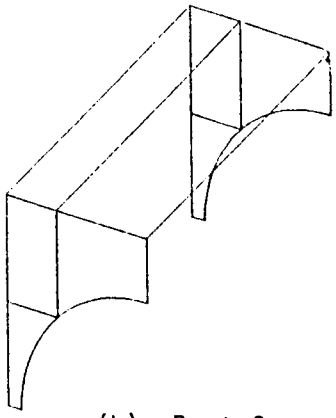


(b) Plotting by MOVIE.BYU

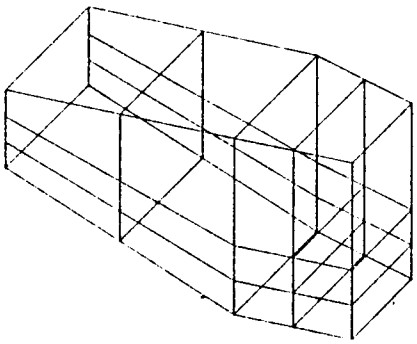
Figure D.1. Examples of Plotting by INGRID and MOVIE.BYU



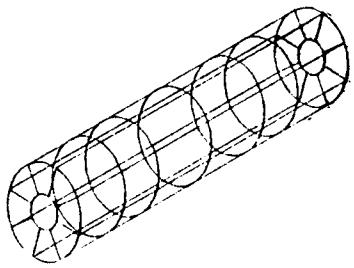
(a) Part 1



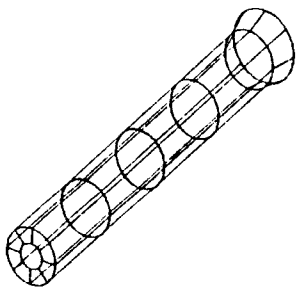
(b) Part 2



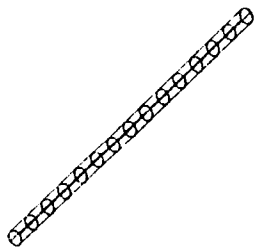
(c) Part 3



(d) Part 4



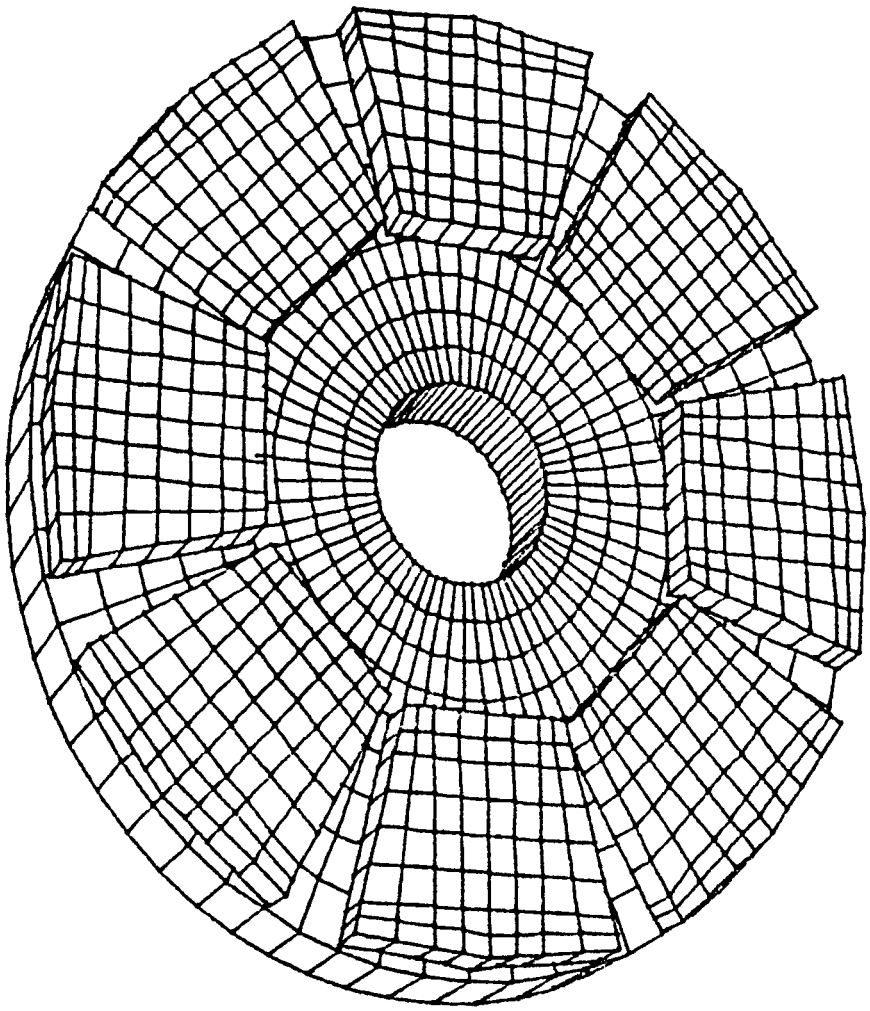
(e) Part 5



(f) Part 6

Figure D.2. Demonstration Model

The demonstration model is broken down to individual parts by INGRID. For each part the corresponding index progression is listed minus any deleted regions.

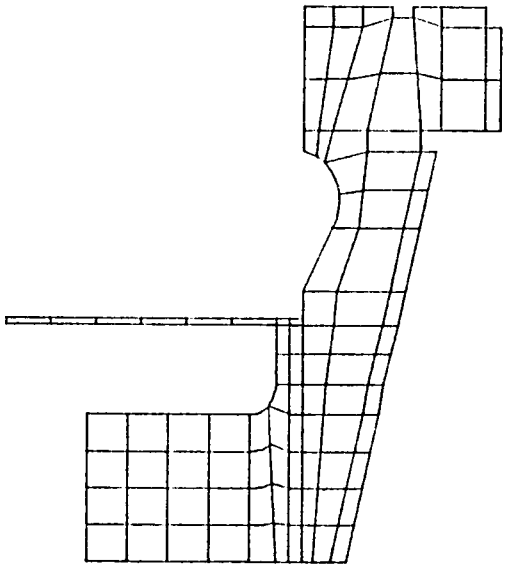


D.3. Mirror Fixture

Axisymmetric Model

The axisymmetric model is shown in Figure D.4 along with a 3-D body of revolution generated from the axisymmetric mode. Since both plane regions and solid regions can be generated easily by an index progression, sometimes conversions from an axisymmetric model to a 3-D solid require only two indices of an index progression to be changed. For this particular example, more changes are required; however, it still demonstrates how major changes to a model can be made with little effort.

(a)



(b)

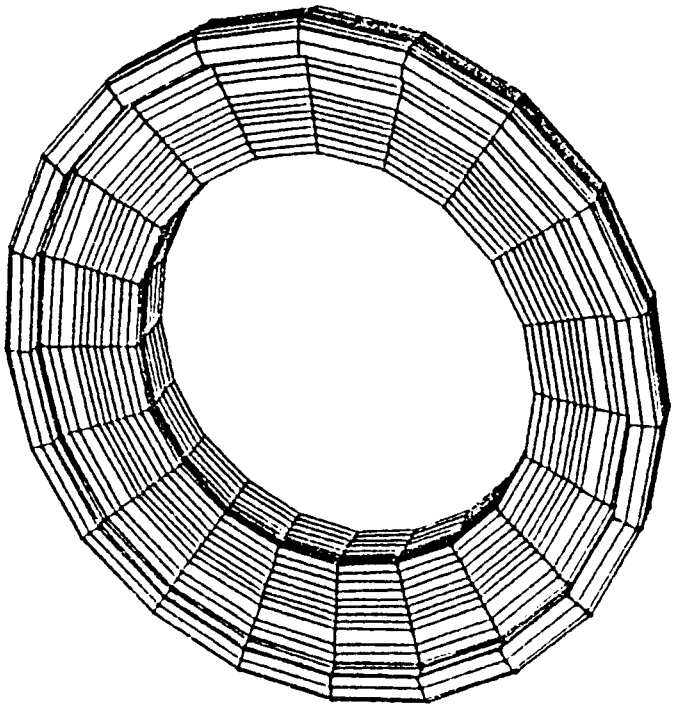


Figure D.4. Axisymmetric Model

APPENDIX E

SAMPLE DATA

```
    DRAG MESH EXAMPLE
```

```
DRAG MESH EXAMPLE
PLOTS
START
CYLI
     1     3     7     9    11
     1    18
     1     3     5     7
       0.25        0.5       1.25        1.5       1.75
        0.0      180.0
        0.0       0.25        0.5       0.75
D  1 1 1 4 2 2
D  1 1 3 4 2 4
END
```

```
   DUMORE MACHINE MODEL
```

```
  DUMORE MACHINE MODEL
PLOTS
PAPEC
LOOP
MOVIE
START
C    PART1
MATERIAL        1
ZONODE
CYLINDRICA
     2     3
     2    10
     2     3     5     6
       1.25       1.875
        0.0       360.0
        0.0         1.5       6.25       7.75
END
START
C    PART2
MATERIAL        2
ANODE
THICKNESS       0.25
     2     3     4
     2     3    -4
    -2    -3
        0.0     -1.326    -2.0625
        0.0      1.326     2.6875
        1.5       6.25
A  2 1 1 2 2 1 3              0.0         0.0         1.5       1.875
A  1 2 1 2 2 1 3              0.0         0.0         1.5       1.875
A  2 1 2 2 2 2 3              0.0         0.0        6.25       1.875
A  1 2 2 2 2 2 3              0.0         0.0        6.25       1.875
D  1 1 1 2 2 2
END
```

```
START
C   PART3
8NODE
MATERIAL        2
THICKNESS       0.25
     2     4    -5    -6
    -2     3     4    -5
    -2    -3
    -17.26        -6.9   -4.48125    -2.0625
       0.0        1.326     2.6875      6.875
     0.625        7.125
0   2 1 1 3 1 2
8   3 1 1 4 1 2 1 1 1 1 1 1
J    1     0     0     0        2.7        3.7        4.7        5.75
K    3     0     0     0     1.0625     6.6875
K    4     0     0     0        1.5       6.25
END
START
C   PART4
8NODE
MATERIAL        3
THICKNESS       0.5
CYLI
     1    -2
     2    10
    -1     2     3     5     6    -7
       0.45       1.25
        0.0      360.0
       -2.0        0.0        1.5       6.25       7.75       10.0
END
START
C   PART5
8NODE
MATERIAL        4
THICKNESS       0.3
CYLI
     1    -2
     2    10
     1     2    -4
       0.45        1.0
        0.0      360.0
       10.0       11.0       25.7
     2     0     1     1       1.25
END
START
C   PART6
MATERIAL        1
2CNODE
     2     3
     2     3
     1     7    15
       -1.0        1.0
       -1.0        1.0
       -2.0       10.0       25.7
A   1 1 1 2 2 3 3             0.0        0.0        0.0        0.45
END
```

```
   MIRRCR FIXTURE

  MIRRCR  FIXTURE
PAFEC
PLOTS
MOVIE
TOLER            0.003
CCORD     8
1
C              1.0       -90.0
C              1.0         0.0
2
C              1.0       -45.0
C              1.0        45.0

C              1.0         0.0
C              1.0        90.0
4
C              1.0        45.0
C              1.0       135.0
5
C              1.0        90.0
C              1.0       180.0
6
C              1.0       135.0
C              1.0       225.0
7
C              1.0       180.0
C              1.0       270.0
8
C              1.0       225.0
C              1.0       315.0
START
REPEAT           1
REPEAT           2
REPEAT           3
REPEAT           4
REPEAT           5
REPEAT           6
REPEAT           7
REPEAT           8
MATERIAL         1
CYLIN
     2     7    10    11
     2     4     6     8    10
     2     3
        6.0         17.0         25.0     28.125
       67.5         78.1         90.0      101.9      112.5
        0.0          6.0
8  1 3 1 1 2 1 0 0 1 0 0 0
8  1 3 1 4 3 2 1 0 0 0 0 0
8  1 1 1 1 3 2 1 1 0 0 0 0
END-
START
REPEAT           1
REPEAT           2
REPEAT           3
REPEAT           4
REPEAT           5
REPEAT           6
REPEAT           7
REPEAT           8
```

```
MATERIAL        2
     2     4     6     8    10
     2     4     8    10
     2     3
      -5.3        -3.5         0.0         3.5         5.3
     14.28        17.0         25.0        27.185
       6.5         9.4
     1     4     1     0       -8.54       26.66        6.5
     1     4     2     0       -8.54       26.66        9.4
     3     4     1     0         0.0       27.55        6.5
     3     4     2     0         0.0       27.55        9.4
     5     4     1     0        8.54       26.66        6.5
     5     4     2     0        8.54       26.66        9.4
     1     3     1     0      -8.106        25.0        6.5
     1     3     2     0      -8.106        25.0        9.4
     5     3     1     0       8.106        25.0        6.5
     5     3     2     0       8.106        25.0        9.4
     1     2     1     0      -6.012         17.        6.5
     1     2     2     0      -6.012        17.0        9.4
     5     2     1     0       6.012        17.0        6.5
     5     2     2     0       6.012        17.0        9.4
B  3 1 1 3 4 2 1 0 0 0 0 0
END-
```

AXISYMMETRIC MODEL

```
  AXISYMMETRIC MIRROR PROBLEM
PAFEC
AXISYMMETRIC
TOLER              0.01
PLOTS
START
BNODE
     2     6     7     8     9    10    11    12    13    14
     2     6     7     9    10    11    13    14    16    17
    -1
     -4.872      -2.947      -2.637       -2.48       -2.32        -2.2        -2.0        -1.8
      -.183         0.0
      9.125      10.625       10.935      11.535      11.875      12.514      13.285      13.500
     14.550      14.750
         0.0
D  1 2 1 310 1
D  3 4 1 5 7 1
D  5 9 1 710 1
D  9 9 11010 1
D  7 7 1 8 8 1
D  8 1 110 8 1
J    4     0     0     1    -2.480      -2.480      -2.480      -2.480         0.0         0.0
      -2.137      -2.137      -1.975      -1.975
J    3     0     3     1    -2.637      -2.637      -2.637      -2.637         0.0         0.0
      -2.320      -2.320      -2.320      -2.320
A  2 2 1 3 2 1 3          -2.947       10.935         0.0        0.31
A  3 2 1 3 3 1 3          -2.947       10.935         0.0        0.31
J    8     0     0     1      -1.8      -1.425      -1.368      -1.197      -1.112      -0.952
      -0.755        -0.7        -0.7        -0.7
J    7     0     0     1    -1.983      -1.608      -1.550      -1.380      -1.295      -1.135
      -0.938      -0.938      -1.030      -1.030
J    6     0     0     1      -2.2      -2.073      -2.047      -1.960      -1.916      -1.664
      -1.565      -1.565      -1.285      -1.285
J    5     0     0     1    -2.320      -2.320      -2.320      -2.320      -2.320      -1.975
      -1.975      -1.975      -1.630      -1.630
A  4 7 1 5 7 1 3          -2.395       12.785         0.0         0.5
A  5 6 1 5 7 1 3          -2.395       12.785         0.0         0.5
     6     9     1     0    -1.285       14.650
     7     9     1     0    -1.030       14.650
END-
```

```
  START
  8NCDE
       2     8     9    10
       2     3
      -1
      -5.832      -2.637      -2.480      -2.320
      11.535      11.600
         0.0
B  1 1 1 1 2 1 1 0 0 0 0 0
END-
```

3-D 8CDY OF REVOLUTICN

AXISYMMETRIC MIRROR PRCBLEM

```
PAFEC
TCLER              0.01
EXCH          312
PLCTS
START
CYLI
      2     5     7     8     9    10    11    12    13    14
      2     6     7     9    10    11    13    14    16    17
      1    21
     -4.872      -2.947      -2.637       -2.48       -2.32        -2.2        -2.0        -1.8
      -.183         0.0
      9.125      10.625      10.935      11.535      11.875      12.514      13.285      13.500
     14.550      14.750
        0.0       360.0
D  1 2 1 310 2
D  3 4 1 5 7 2
D  6 9 1 710 2
D  9 9 11010 2
D  7 7 1 8 8 2
D  8 1 110 8 2
J     4     0     0     1    -2.480      -2.480      -2.480      -2.480         0.0         0.0
      -2.137      -2.137      -1.975      -1.975
J     3     0     0     1    -2.637      -2.637      -2.637      -2.637         0.0         0.0
      -2.320      -2.320      -2.320      -2.320
A  2 2 1 3 2 1 3          -2.947      10.935         0.0         0.31
A  2 2 2 3 2 2 3          -2.947      10.935       360.0         0.31
A  3 2 1 3 3 1 3          -2.947      10.935         0.0         0.31
A  3 2 2 3 3 2 3          -2.947      10.935       360.0         0.31
J     8     0     0     1      -1.8      -1.425      -1.368      -1.197      -1.112      -0.952
      -0.755        -0.7        -0.7        -0.7
J     7     0     0     1    -1.983      -1.608      -1.550      -1.380      -1.295      -1.135
      -0.938      -0.938      -1.030      -1.030
J     6     0     0     1      -2.2      -2.073      -2.047      -1.960      -1.916      -1.664
      -1.565      -1.565      -1.285      -1.285
J     5     0     0     1    -2.320      -2.320      -2.320      -2.320      -2.320      -1.975
      -1.975      -1.975      -1.630      -1.630
A  4 7 1 5 7 1 3          -2.395      12.785         0.0         0.5
A  4 7 2 5 7 2 3          -2.395      12.785       360.0         0.5
A  5 6 1 5 7 1 3          -2.395      12.785         0.0         0.5
A  5 6 2 5 7 2 3          -2.395      12.785       360.0         0.5
      6     9     0    12    -1.285      14.650
      7     9     0    12    -1.030      14.650
END-
START
CYLI
      2     8     9    10
      2     3
      1    21
     -5.832      -2.637      -2.480      -2.320
     11.535      11.600
        0.0       360.0
B  1 1 1 1 2 1 1 0 0 0 0 0
END-
```

APPENDIX F

PROGRAM LISTINGS

1. Fortran Listing of INGRID

```
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C   PROGRAM INGRID                          BY: DOUGLAS STILLMAN
C
C   INGRID IS A GENERAL PURPOSE FINITE ELEMENT MESH GENERATOR. FOR MORE
C   INFORMATION CONSULT THE USER'S GUIDE OR CONTACT:
C            J. E. STONEKING
C            DEPARTMENT OF ENGINEERING SCIENCE AND MECHANICS
C            UNIVERSITY OF TENNESSEE
C            KNOXVILLE, TENNESSEE 37916
C            TELEPHONE: 615-974,8184
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C   LIST OF COMMON BLOCKS AND ROUTINES IN WHICH THEY ARE USED
C
C   BOUND  -  BCS     BCS2    DLEE
C
C   DEL    -  DELETE  DELET2  DLEE
C
C   DUMPAR-   ION1    ION2    ION3    ION4
C
C   DUMPA2-   IOP1    IOP2    IOP3    IOP4
C
C   JUNK   -  EL1D    EL2D    EL3D    EXCHNG  INFO    INIT    OUTPUT
C             PLOTMK
C
C   JUNK2 -   ELTHRE  ELTWO   INFO    INIT    INPLOT
C
C   KNTS   -  MAIN    DLEE    ELTHRE  ELTWO   INFO    INIT    INPLOT
C             IOP1    LOADS2  OUTPUT
C
C   LOAD   -  DLEE    LOADS   LOADS2
C
C   MID    -  DLEE    MIDSID  MIDSI2
C
C   PARTS -   DLEE    INFO    INIT    IOP1    MAXMIN  OUTPUT  SCALE
C
C   PARTS2-   INPLOT  PART    SCALE
C
C   PROP   -  DOUBLE  ELTHRE  ELTWO   MAX     SEQ2    TRANS
C
C   OTSARG-   CTRAN   CTRAN2
C
C   REPEAT-   CTRAN2  DLEE    TRANS
C
C   SCALES-   MAP     PLOT    SCALE   SCALE2
C
C   SEQD1 -   FN3     FN5     SEQ2
C
C   SYSAPY-   MAIN    BEAM    BCS3    BOUT1   BOUT2   EL1D    EL2D
C             EL3D    FLAT    FN1     FN2     INPLOT  ITST    LOADS3
C             MAP     NORM    OUTPUT  VOLUME
C
C   VECFNC-   CIRCLE  DLEE    MOVE    PARABO
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C   MAIN PROGRAM
C
C   SUBROUTINES CALLED:
```

```
C                                                                       CMAIN0061
C  DLEE  INIT  ICN2  IOP2  INPLOT  OUTPUT                               CMAIN0062
C                                                                       CMAIN0063
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCMAIN0064
      COMMON/SYSARY/D1(54004), N(18002)                                 MAIN0065
      NODES=18002                                                       MAIN0066
      LIM=54004                                                         MAIN0067
C                                                                       MAIN0068
C  CALL INITIALIZING ROUTINES                                           MAIN0069
C                                                                       MAIN0070
      CALL INIT(IOUT,IPLOT,TOLER,IMESH)                                 MAIN0071
      IF(IOUT.EQ.1)WRITE(6,30)                                          MAIN0072
      IF(IOUT.EQ.2)WRITE(6,40)                                          MAIN0073
      IF(IOUT.EQ.3)WRITE(6,50)                                          MAIN0074
C                                                                       MAIN0075
C  CALL MESH GENERATION ROUTINES                                        MAIN0076
C                                                                       MAIN0077
      CALL DLEE (D1,LIM,N,NODES)                                        MAIN0078
C                                                                       MAIN0079
C  DUMP OUTPUT BUFFERS                                                  MAIN0080
C                                                                       MAIN0081
      CALL ICN2                                                         MAIN0082
      CALL IOP2                                                         MAIN0083
      IF(IPLOT.EQ.0)GO TO 10                                            MAIN0084
      REWIND 7                                                          MAIN0085
      REWIND 8                                                          MAIN0086
      REWIND 9                                                          MAIN0087
      REWIND 10                                                         MAIN0088
C                                                                       MAIN0089
C  CALL PLOTTING ROUTINES                                               MAIN0090
C                                                                       MAIN0091
      CALL INPLOT                                                       MAIN0092
   10 CONTINUE                                                          MAIN0093
      IF(IOUT.EQ.0)GO TO 20                                             MAIN0094
      REWIND 7                                                          MAIN0095
      REWIND 8                                                          MAIN0096
      REWIND 9                                                          MAIN0097
      REWIND 10                                                         MAIN0098
      REWIND 13                                                         MAIN0099
C                                                                       MAIN0100
C  CALL ROUTINES TO COMPLETE PROCESSING OF MESH                         MAIN0101
C                                                                       MAIN0102
      CALL OUTPUT(IOUT,TOLER)                                           MAIN0103
   20 CONTINUE                                                          MAIN0104
      STOP                                                              MAIN0105
   30 FORMAT(12H SAP5 FORMAT)                                           MAIN0106
   40 FORMAT(12H SAP6 FORMAT)                                           MAIN0107
   50 FORMAT(13H PAFEC FORMAT)                                          MAIN0108
      END                                                               MAIN0109
      SUBROUTINE BEAM                                                   BEAM0001
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCBEAM0002
C                                                                       CBEAM0003
C  SUBROUTINE BEAM PERFORMS NODE AND ELEMENT GENERATION OF BEAMS        CBEAM0004
C                                                                       CBEAM0005
C  SUBROUTINE BEAM IS CALLED BY INIT                                    CBEAM0006
C                                                                       CBEAM0007
C  SUBROUTINES CALLED:  ION1    ICF1    MAXMIN                         CBEAM0008
C                                                                       CBEAM0009
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCBEAM0010
      COMMON/SYSARY/X(1000),Y(1000),Z(1000)                             BEAM0011
      COMMON/KNTS/NODE,IBEAM,ITWO,ITHRE,LOAD,LINES,MIN                  BEAM0012
```

```
      DATA FIXC/1HF/                                                    BEAM0013
      BACKSPACE 12                                                      BEAM0014
      READ(12,170)XD,YD,ZD                                              BEAM0015
      N=1                                                               BEAM0016
C                                                                       BEAM0017
C  INPUT NODES                                                          BEAM0018
C                                                                       BEAM0019
   10 READ(12,110)FIX,IN,X(N),Y(N),Z(N)                                 BEAM0020
      IF(IN.EQ.0)GO TO 20                                               BEAM0021
      XO=X(N)+XD                                                        BEAM0022
      YO=Y(N)+YD                                                        BEAM0023
      ZO=Z(N)+ZD                                                        BEAM0024
      IF(FIX.NE.FIXC)IBC=0                                              BEAM0025
      IF(FIX.EQ.FIXC)IBC=111111                                         BEAM0026
      CALL MAXMIN(XO,YO,ZO)                                             BEAM0027
      CALL ION1(XO,YO,ZO,IBC)                                           BEAM0028
      N=N+1                                                             BEAM0029
      GO TO 10                                                          BEAM0030
   20 CONTINUE                                                          BEAM0031
C                                                                       BEAM0032
C  GENERATE NEW NODES AND ELEMENTS                                      BEAM0033
C                                                                       BEAM0034
      NEL=0                                                             BEAM0035
      N=N-1                                                             BEAM0036
      NODE=N                                                            BEAM0037
      N=N+1                                                             BEAM0038
   30 READ(12,120,END=100)IS,IF,INEL,NORMAL,MAT,IBTYPE                  BEAM0039
      IF(MAT.EQ.0)MAT=1                                                 BEAM0040
      IF(IS.EQ.0)GO TO 100                                              BEAM0041
      IF(INEL.EQ.1)GO TO 50                                             BEAM0042
      VX=(X(IF)-X(IS))/INEL                                             BEAM0043
      VY=(Y(IF)-Y(IS))/INEL                                             BEAM0044
      VZ=(Z(IF)-Z(IS))/INEL                                             BEAM0045
      INELM1=INEL-1                                                     BEAM0046
      DO 40 I=1,INELM1                                                  BEAM0047
      XO=X(IS)+I*VX+XD                                                  BEAM0048
      YO=Y(IS)+I*VY+YD                                                  BEAM0049
      ZO=Z(IS)+I*VZ+ZD                                                  BEAM0050
      NODE=NODE+1                                                       BEAM0051
      IBC=0                                                             BEAM0052
      CALL ION1(XO,YO,ZO,IBC)                                           BEAM0053
   40 CONTINUE                                                          BEAM0054
   50 CONTINUE                                                          BEAM0055
      IF(INEL.GT.1)GO TO 60                                             BEAM0056
      NEL=NEL+1                                                         BEAM0057
      WRITE(10)IS,IF,NORMAL,MAT,IBTYPE                                  BEAM0058
      CALL IOP1(IS,0,IF)                                                BEAM0059
      GO TO 90                                                          BEAM0060
   60 CONTINUE                                                          BEAM0061
      IF(INEL.GT.2)GO TO 70                                             BEAM0062
      NEL=NEL+1                                                         BEAM0063
      WRITE(10)IS,NODE,NORMAL,MAT,IBTYPE                                BEAM0064
      CALL IOP1(IS,0,NODE)                                              BEAM0065
      NEL=NEL+1                                                         BEAM0066
      WRITE(10)NODE,IF,NORMAL,MAT,IBTYPE                                BEAM0067
      CALL IOP1(NODE,0,IF)                                              BEAM0068
      GO TO 90                                                          BEAM0069
   70 CONTINUE                                                          BEAM0070
      NEL=NEL+1                                                         BEAM0071
      WRITE(10)IS,N,NORMAL,MAT,IBTYPE                                   BEAM0072
      CALL IOP1(IS,0,N)                                                 BEAM0073
```

```
      NODEM1=NODE-1                                                    BEAM0074
      DO 80 I=N,NODEM1                                                 BEAM0075
      NEL=NEL+1                                                        BEAM0076
      II=I+1                                                           BEAM0077
      WRITE(10)I,II,NORMAL,MAT,IBTYPE                                  BEAM0078
      ISAVE=I                                                          BEAM0079
      CALL IOP1(ISAVE,0,II)                                            BEAM0080
   80 CONTINUE                                                         BEAM0081
      NEL=NEL+1                                                        BEAM0082
      WRITE(10)NODE,IF,NORMAL,MAT,IBTYPE                               BEAM0083
      CALL IOP1(NODE,0,IF)                                             BEAM0084
   90 CONTINUE                                                         BEAM0085
      N=NODE+1                                                         BEAM0086
      GO TO 30                                                         BEAM0087
  100 CONTINUE                                                         BEAM0088
      IBEAM=NEL+1                                                      BEAM0089
      NODE=NODE+1                                                      BEAM0090
      RETURN                                                           BEAM0091
  110 FORMAT(A1,I4,3F10.0)                                             BEAM0092
  120 FORMAT(6I5)                                                      BEAM0093
  130 FORMAT(I5,6(4X,1H0),3F10.4)                                      BEAM0094
  140 FORMAT(I5,6(4X,1H1),3F10.4)                                      BEAM0095
  150 FORMAT(6I5)                                                      BEAM0096
  160 FORMAT(35X,3E10.3)                                               BEAM0097
  170 FORMAT(5X,3F10.0)                                                BEAM0098
      END                                                              BEAM0099
      SUBROUTINE BCS(IB,IBC)                                           BCS0001
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCBCS0002
C                                                                      CBCS0003
C  THIS SUBROUTINE STORES COMMANDS FOR RESTRAINING BOUNDARY CONDITIONS CBCS0004
C                                                                      CBCS0005
C  CALLED BY:  SEQ2                                                    CBCS0006
C                                                                      CBCS0007
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCBCS0008
      DIMENSION IB(6),IBC(6)                                           BCS0009
      COMMON/BCUND/NBC,IBCAPL(100,6),IBCS(100,6)                       BCS0010
      NBC=NBC+1                                                        BCS0011
      DO 10 I=1,6                                                      BCS0012
      IBCAPL(NBC,I)=IB(I)                                              BCS0013
      IBCS(NBC,I)=IBC(I)                                               BCS0014
   10 CONTINUE                                                         BCS0015
      RETURN                                                           BCS0016
      END                                                              BCS0017
      SUBROUTINE BCS2(I,J,K,N)                                         BCS20001
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCBCS20002
C                                                                      CBCS20003
C  SUBROUTINE BCS2 RESTRAINS THE BOUNDARIES OF NODES BEING OUTPUT      CBCS20004
C                                                                      CBCS20005
C  CALLED BY:  ELTHRE  ELTWO                                           CBCS20006
C                                                                      CBCS20007
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCBCS20008
      DIMENSION N(6)                                                   BCS20009
      COMMON/BCUND/NBC,IBCAPL(100,6),IBCS(100,6)                       BCS20010
      IF(NBC.EQ.0)RETURN                                               BCS20011
      DO 20 L=1,NBC                                                    BCS20012
      IF(I.LT.IBCAPL(L,1).OR.I.GT.IBCAPL(L,4))GO TO 20                 BCS20013
      IF(J.LT.IBCAPL(L,2).OR.J.GT.IBCAPL(L,5))GO TO 20                 BCS20014
      IF(K.LT.IBCAPL(L,3).OR.K.GT.IBCAPL(L,6))GO TO 20                 BCS20015
      DO 10 M=1,6                                                      BCS20016
      IF(N(M).EQ.0)N(M)=IBCS(L,M)                                      BCS20017
   10 CONTINUE                                                         BCS20018
```

```
   20 CONTINUE                                                          BCS20019
      RETURN                                                            BCS20020
      END                                                               BCS20021
      SUBROUTINE BCS3(IB,NODE,ICMD)                                     BCS30001
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCBCS30002
C                                                                       CBCS30003
C  SUBROUTINE BCS3 PACKS OR UNPACKS THE BOUNDARY CONDITION ARRAY        CBCS30004
C                                                                       CBCS30005
C  CALLED BY:  BOUT1   EL1D    EL2D    OUTPUT                           CBCS30006
C                                                                       CBCS30007
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCBCS30008
      DIMENSION IB(6)                                                   BCS30009
      COMMON/SYSARY/X(8000),Y(8000),Z(8000),N(8000)                     BCS30010
     1,IBB(8000),IFLAG(8000),V(8000,3)                                  BCS30011
      IF(ICMD.EQ.1)GO TO 20                                             BCS30012
C                                                                       BCS30013
C  PACK ARRAY                                                           BCS30014
C                                                                       BCS30015
      IBB(NODE)=0                                                       BCS30016
      DO 10 I=1,6                                                       BCS30017
      IBB(NODE)=IBB(NODE)+IB(I)*10**(I-1)                               BCS30018
   10 CONTINUE                                                          BCS30019
      RETURN                                                            BCS30020
   20 CONTINUE                                                          BCS30021
C                                                                       BCS30022
C  UNPACK ARRAY                                                         BCS30023
C                                                                       BCS30024
      NN=IBB(NODE)                                                      BCS30025
      DO 30 I=1,6                                                       BCS30026
      IB(I)=MOD(NN,10)                                                  BCS30027
      A=(NN-IB(I))/10                                                   BCS30028
      NN=INT(A)                                                         BCS30029
   30 CONTINUE                                                          BCS30030
      RETURN                                                            BCS30031
      END                                                               BCS30032
      SUBROUTINE BOUT1(NODES)                                           BUT10001
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCBUT10002
C                                                                       CBUT10003
C  SUBROUTINE BOUT1 CHECKS FOR PLATES WHICH HAVE ROTATIONS TO BE        CBUT10004
C  RESTRAINED ALONG A COORDINATE DIRECTION.                             CBUT10005
C                                                                       CBUT10006
C  CALLED BY:  OUTPUT                                                   CBUT10007
C                                                                       CBUT10008
C  CALLS SUBROUTINES:  BCS3                                             CBUT10009
C                                                                       CBUT10010
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCBUT10011
      DIMENSION IB(6)                                                   BUT10012
      COMMON/SYSARY/X(8000),Y(8000),Z(8000),N(8000)                     BUT10013
     1,IBB(8000),IFLAG(8000),V(8000,3)                                  BUT10014
      DO 10 I=1,NODES                                                   BUT10015
      IF(IFLAG(I).EQ.0.OR.IFLAG(I).EQ.2)GO TO 10                        BUT10016
      I1=I                                                              BUT10017
      CALL BCS3(IB,I1,1)                                                BUT10018
      IF(ABS(V(I,1)).GT.0.99)IB(4)=3                                    BUT10019
      IF(ABS(V(I,2)).GT.0.99)IB(5)=3                                    BUT10020
      IF(ABS(V(I,3)).GT.0.99)IB(6)=3                                    BUT10021
      I1=I                                                              BUT10022
      CALL BCS3(IB,I1,0)                                                BUT10023
   10 CONTINUE                                                          BUT10024
      RETURN                                                            BUT10025
      END                                                               BUT10026
```

```
      SUBROUTINE BCUT2(NODES,ICUT)                                      BUT20001
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCBUT20002
C                                                                       CBUT20003
C  SUBROUTINE BCUT2 OUTPUTS BOUNDARY CONDITIONS AND BOUNDARY ELEMENTS   CBUT20004
C                                                                       CBUT20005
C  CALLED BY:  OUTPUT                                                   CBUT20006
C                                                                       CBUT20007
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCBUT20008
      COMMON/SYSARY/X(8000),Y(8000),Z(8000),N(8000)                     BUT20009
     1,IB3(8000),IFLAG(8000),V(8000,3)                                  BUT20010
      NEWND=NODES+1                                                     BUT20011
      DO 10 I=1,NODES                                                   BUT20012
      IF(IFLAG(I).EQ.0.OR.IFLAG(I).EQ.2)GO TO 10                        BUT20013
      VX=X(I)+V(I,1)                                                    BUT20014
      VY=Y(I)+V(I,2)                                                    BUT20015
      VZ=Z(I)+V(I,3)                                                    BUT20016
      IF(ABS(V(I,1)).GT.0.99.OR.V(I,2).GT.0.99.OR.V(I,3).GT.0.99)       BUT20017
     1GO TO 10                                                          BUT20018
      I1=N(I)                                                           BUT20019
      IF(ICUT.EQ.1)WRITE(11,30)NEWND,VX,VY,VZ                           BUT20020
      IF(ICUT.EQ.2)WRITE(11,50)NEWND,VX,VY,VZ                           BUT20021
      IF(ICUT.EQ.2)WRITE(15,60)NEWND                                    BUT20022
      IF(ICUT.EQ.1)WRITE(14,40)I1,NEWND                                 BUT20023
      IF(ICUT.EQ.2)WRITE(14,70)I1,NEWND                                 BUT20024
      NEL=NEL+1                                                         BUT20025
      NEWND=NEWND+1                                                     BUT20026
   10 CONTINUE                                                          BUT20027
      NEWND=NEWND-1                                                     BUT20028
      WRITE(6,20)NEWND,NEL                                              BUT20029
      RETURN                                                            BUT20030
   20 FORMAT(16H NUMBER OF NODES,/,21H AFTER TOLERENCING = ,I5,         BUT20031
     1/,31H NUMBER OF BOUNDARY ELEMENTS = ,I5)                          BUT20032
   30 FORMAT(I5,30H    1    1    1    1    1    1,3F10.4)               BUT20033
   40 FORMAT(2I5,20X,5H    1)                                          BUT20034
   50 FORMAT(I10,3F10.4)                                                BUT20035
   60 FORMAT(I5,10X,3HFIX,2X,2HNM)                                      BUT20036
   70 FORMAT(2I5,18X,2HNR)                                              BUT20037
      END                                                               BUT20038
      SUBROUTINE CIRCLE(X1,Y1,Z1,X3,Y3,Z3,X2,Y2,Z2,IB)                  CRCL0001
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCRCL0002
C                                                                       CCRCL0003
C  SUBROUTINE CIRCLE STORES COMMANDS FOR MOVING NODES TO FIT A CIRCLE   CCRCL0004
C                                                                       CCRCL0005
C  CALLED BY:  SEQ2                                                     CCRCL0006
C                                                                       CCRCL0007
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCRCL0008
      COMMON/VECFNC/NFN,IFN(100),IAPLY(100,6),COEF(100,5)               CRCL0009
      DIMENSION IB(6)                                                   CRCL0010
      NFN=NFN+1                                                         CRCL0011
      IFN(NFN)=2                                                        CRCL0012
      DO 10 I=1,6                                                       CRCL0013
      IAPLY(NFN,I)=IB(I)                                                CRCL0014
   10 CONTINUE                                                          CRCL0015
      A=SQRT((X3-X1)**2+(Y3-Y1)**2+(Z3-Z1)**2)                          CRCL0016
      B=SQRT((X3-X2)**2+(Y3-Y2)**2+(Z3-Z2)**2)                          CRCL0017
      C=SQRT((X2-X1)**2+(Y2-Y1)**2+(Z2-Z1)**2)                          CRCL0018
      S=(A+B+C)/2.0                                                     CRCL0019
      AREA=SQRT(S*(S-A)*(S-B)*(S-C))                                    CRCL0020
      R=A*B*C/(4.0*AREA)                                                CRCL0021
      QC=(A*A+C*C-B*B)/(2.0*C*C)                                        CRCL0022
      X4=(X2-X1)*QC+X1                                                  CRCL0023
```

```
      Y4=(Y2-Y1)*QC+Y1                                                  CRCL0024
      Z4=(Z2-Z1)*QC+Z1                                                  CRCL0025
      VX=X3-X4                                                          CRCL0026
      VY=Y3-Y4                                                          CRCL0027
      VZ=Z3-Z4                                                          CRCL0028
      ANORMV=SQRT(VX*VX+VY*VY+VZ*VZ)                                    CRCL0029
      VX=VX/ANORMV                                                      CRCL0030
      VY=VY/ANORMV                                                      CRCL0031
      VZ=VZ/ANORMV                                                      CRCL0032
      COEF(NFN,1)=VX                                                    CRCL0033
      COEF(NFN,2)=VY                                                    CRCL0034
      COEF(NFN,3)=VZ                                                    CRCL0035
      COEF(NFN,4)=R                                                     CRCL0036
      COEF(NFN,5)=C                                                     CRCL0037
      RETURN                                                            CRCL0038
      END                                                               CRCL0039
      SUBROUTINE CIRCL2(XP,YP,ZP,XC,YC,ZC,XQ,YQ,ZQ,IB,RADIUS)           CRCL0001
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCRCL0002
C                                                                       CCRCL0003
C  SUBROUTINE CIRCL2  DETERMINES THE POINT Z ON ARC PQ WHEN GIVEN THE   CCRCL0004
C  CENTER OF THE CIRCLE AT POINT C.                                     CCRCL0005
C                                                                       CCRCL0006
C  CALLED BY:  SEQ2                                                     CCRCL0007
C                                                                       CCRCL0008
C  CALLS SUBROUTINES:  PARABC                                           CCRCL0009
C                                                                       CCRCL0010
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCRCL0011
      DIMENSION IB(6)                                                   CRCL0012
      IF(RADIUS.EQ.0.0)GO TO 10                                         CRCL0013
      XCP=XP-XC                                                         CRCL0014
      YCP=YP-YC                                                         CRCL0015
      ZCP=ZP-ZC                                                         CRCL0016
      XCQ=XQ-XC                                                         CRCL0017
      YCQ=YQ-YC                                                         CRCL0018
      ZCQ=ZQ-ZC                                                         CRCL0019
      CP=SQRT(XCP*XCP+YCP*YCP+ZCP*ZCP)                                  CRCL0020
      CQ=SQRT(XCQ*XCQ+YCQ*YCQ+ZCQ*ZCQ)                                  CRCL0021
      XP=XCP*RADIUS/CP+XC                                               CRCL0022
      YP=YCP*RADIUS/CP+YC                                               CRCL0023
      ZP=ZCP*RADIUS/CP+ZC                                               CRCL0024
      XQ=XCQ*RADIUS/CQ+XC                                               CRCL0025
      YQ=YCQ*RADIUS/CQ+YC                                               CRCL0026
      ZQ=ZCQ*RADIUS/CQ+ZC                                               CRCL0027
   10 CONTINUE                                                          CRCL0028
      XR=XP-XC                                                          CRCL0029
      YR=YP-YC                                                          CRCL0030
      ZR=ZP-ZC                                                          CRCL0031
      R=SQRT(XR*XR+YR*YR+ZR*ZR)                                         CRCL0032
      XV=(XP+XQ)/2-XC                                                   CRCL0033
      YV=(YP+YQ)/2-YC                                                   CRCL0034
      ZV=(ZP+ZQ)/2-ZC                                                   CRCL0035
      V=SQRT(XV*XV+YV*YV+ZV*ZV)                                         CRCL0036
      XZ=XV*R/V+XC                                                      CRCL0037
      YZ=YV*R/V+YC                                                      CRCL0038
      ZZ=ZV*R/V+ZC                                                      CRCL0039
      CALL PARABC(XP,YP,ZP,XZ,YZ,ZZ,XQ,YQ,ZQ,IB)                        CRCL0040
      RETURN                                                            CRCL0041
      END                                                               CRCL0042
      SUBROUTINE CTRAN                                                  CTRN0001
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCTRN0002
C                                                                       CCTRN0003
```

```
C  SUBROUTINE CTRAN SAVES THE COORDINATE TRANSFORMATIONS                    CCTRN0004
C                                                                          CCTRN0005
C  CALLED BY:  INIT                                                        CCTRN0006
C                                                                          CCTRN0007
C  CALLS SUBROUTINES:  CYLIND  SPHERI                                      CCTRN0008
C                                                                          CCTRN0009
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCTRN0010
      COMMON/QTSARG/TI(3,3,50),X(3,50),Y(3,50),Z(3,50)                     CTRN0011
      DATA NCYL/1HC/,NSPH/1HS/                                             CTRN0012
      BACKSPACE 12                                                         CTRN0013
      READ(12,40)NC                                                        CTRN0014
      IF(NC.EQ.0)RETURN                                                    CTRN0015
      DO 20 J=1,NC                                                         CTRN0016
      DO 10 I=1,3                                                          CTRN0017
      READ(12,50)KT,X(I,J),Y(I,J),Z(I,J)                                   CTRN0018
      IF(KT.EQ.NCYL)CALL CYLIND(X(I,J),Y(I,J),Z(I,J))                      CTRN0019
      IF(KT.EQ.NSPH)CALL SPHERI(X(I,J),Y(I,J),Z(I,J))                      CTRN0020
   10 CONTINUE                                                             CTRN0021
   20 CONTINUE                                                             CTRN0022
      DO 30 I=1,NC                                                         CTRN0023
      X1=X(2,I)-X(1,I)                                                     CTRN0024
      Y1=Y(2,I)-Y(1,I)                                                     CTRN0025
      Z1=Z(2,I)-Z(1,I)                                                     CTRN0026
      X2=X(3,I)-X(1,I)                                                     CTRN0027
      Y2=Y(3,I)-Y(1,I)                                                     CTRN0028
      Z2=Z(3,I)-Z(1,I)                                                     CTRN0029
      S1=X1*X1+Y1*Y1+Z1*Z1                                                 CTRN0030
      S1=SQRT(S1)                                                          CTRN0031
      AX=Y1*Z2-Z1*Y2                                                       CTRN0032
      AY=Z1*X2-X1*Z2                                                       CTRN0033
      AZ=X1*Y2-Y1*X2                                                       CTRN0034
      S2=AX*AX+AY*AY+AZ*AZ                                                 CTRN0035
      S2=SQRT(S2)                                                          CTRN0036
      X1=X1/S1                                                             CTRN0037
      Y1=Y1/S1                                                             CTRN0038
      Z1=Z1/S1                                                             CTRN0039
      AX=AX/S2                                                             CTRN0040
      AY=AY/S2                                                             CTRN0041
      AZ=AZ/S2                                                             CTRN0042
      X2=AY*Z1-AZ*Y1                                                       CTRN0043
      Y2=AZ*X1-AX*Z1                                                       CTRN0044
      Z2=AX*Y1-AY*X1                                                       CTRN0045
      TI(1,1,I)=X1                                                         CTRN0046
      TI(1,2,I)=Y1                                                         CTRN0047
      TI(1,3,I)=Z1                                                         CTRN0048
      TI(2,1,I)=X2                                                         CTRN0049
      TI(2,2,I)=Y2                                                         CTRN0050
      TI(2,3,I)=Z2                                                         CTRN0051
      TI(3,1,I)=AX                                                         CTRN0052
      TI(3,2,I)=AY                                                         CTRN0053
   30 TI(3,3,I)=AZ                                                         CTRN0054
      RETURN                                                               CTRN0055
   40 FORMAT(5X,I5)                                                        CTRN0056
   50 FORMAT(A1,4X,3F10.0)                                                 CTRN0057
      END                                                                  CTRN0058
      SUBROUTINE CTRAN2(XO,YO,IC)                                          CTRN0001
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCTRN0002
C                                                                          CCTRN0003
C  SUBROUTINE CTRAN2 PERFORMS COORDINATE TRANSFORMATIONS ON NODE O         CCTRN0004
C                                                                          CCTRN0005
C  CALLED BY:  ELTHRE  ELTWO                                               CCTRN0006
```

```
C                                                                       CCTRN00C7
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCTRN00C8
      DIMENSION XC(3),XX(3)                                             CTRN00C9
      COMMON/QTSARG/TI(3,3,50),X(3,50),Y(3,50),Z(3,50)                  CTRNC01C
      COMMON/REPEAT/NR,NRT(50)                                          CTRN0011
      KS=NRT(NR)                                                        CTRN0012
      IF(KS.EQ.0)RETURN                                                 CTRN0013
      XC(1)=XO                                                          CTRN0014
      XC(2)=YO                                                          CTRN0015
      XC(3)=ZO                                                          CTRN0016
      DO 10 I=1,3                                                       CTRN0017
      XX(I)=0.0                                                         CTRN0018
      DO 10 J=1,3                                                       CTRN0019
   10 XX(I)=XX(I)+TI(J,I,KS)*XC(J)                                      CTRN002C
      XC(1)=XX(1)+X(1,KS)                                               CTRN0021
      XC(2)=XX(2)+Y(1,KS)                                               CTRN0022
      XC(3)=XX(3)+Z(1,KS)                                               CTRN0023
      XO=XC(1)                                                          CTRN0024
      YO=XC(2)                                                          CTRN0025
      ZO=XC(3)                                                          CTRN0026
      RETURN                                                            CTRNC027
      END                                                               CTRN0028
      SUBROUTINE CYLIND(R,THETA,Z)                                      CYLN00C1
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCYLN00C2
C                                                                       CCYLN0003
C  SUBROUTINE CYLIND PERFORMS COORDINATE TRANSFORMATIONS FROM           CCYLN00C4
C  CYLINDRICAL COORDINATES TO CARTESIAN COORDINATES                     CCYLN00C5
C                                                                       CCYLN00C6
C  CALLED BY:  CTRAN   ELTHRE  ELTWO   SEQ2                             CCYLN00C7
C                                                                       CCYLN00C8
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCYLN00C9
      DATA ST/0.0/,CT/1.0/                                              CYLN0010
      IF(THETA.EQ.THETAO)GO TO 10                                       CYLN0011
      ST=SIN(THETA/57.2958)                                             CYLN0012
      CT=COS(THETA/57.2958)                                             CYLN0013
   10 IF(R.EQ.RO.AND.THETA.EQ.THETAO)GO TO 20                           CYLN0014
      RST=R*ST                                                          CYLN0015
      RCT=R*CT                                                          CYLN0016
   20 CONTINUE                                                          CYLN0017
      X=RCT                                                             CYLN0018
      Y=RST                                                             CYLN0019
      R=X                                                               CYLN002C
      THETA=Y                                                           CYLN0021
      RO=R                                                              CYLN0022
      THETAO=THETA                                                      CYLN0023
      RETURN                                                            CYLN0024
      END                                                               CYLN0025
      SUBROUTINE DELETE(IB)                                             DLTE00C1
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCDLTE00C2
C                                                                       CDLTE0003
C  SUBROUTINE DELETE STORES COMMANDS FOR REGIONS WHICH ARE TO BE        CDLTE00C4
C  DELETED                                                              CDLTE00C5
C                                                                       CDLTECOC6
C  CALLED BY:  SEQ2                                                     CDLTE00C7
C                                                                       CDLTE00C8
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCDLTE00C9
      DIMENSION IB(6)                                                   DLTE0010
      COMMON/DEL/NDEL,IDEL(100,6)                                       DLTE0011
      NDEL=NDEL+1                                                       DLTE0012
      DO 10 I=1,6                                                       DLTECO13
      IDEL(NDEL,I)=IB(I)                                                DLTE0014
```

```
   10 CONTINUE                                                          DLTE0015
      RETURN                                                            DLTE0016
      END                                                               DLTE0017
      SUBROUTINE DELET2(IM,JM,KM,IX,JX,KX,ID)                           DLT20001
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCDLT20002
C                                                                       CDLT20003
C  SUBROUTINE DELET2 DETERMINES IF A REGION IS TO BE DELETED            CDLT20004
C                                                                       CDLT20005
C  CALLED BY:  ELTHRE  ELTWO                                            CDLT20006
C                                                                       CDLT20007
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCDLT20008
      COMMON/DEL/NDEL,IDEL(100,6)                                       DLT20009
      IF(NDEL.EQ.0)RETURN                                               DLT20010
      DO 10 I=1,NDEL                                                    DLT20011
      IF(IM.LT.IDEL(I,1).OR.IX.GT.IDEL(I,4))GO TO 10                    DLT20012
      IF(JM.LT.IDEL(I,2).OR.JX.GT.IDEL(I,5))GO TO 10                    DLT20013
      IF(KM.LT.IDEL(I,3).OR.KX.GT.IDEL(I,6))GO TO 10                    DLT20014
      ID=1                                                              DLT20015
      RETURN                                                            DLT20016
   10 CONTINUE                                                          DLT20017
      RETURN                                                            DLT20018
      END                                                               DLT20019
      SUBROUTINE DLEE (A,KLIM,N,NLIM)                                   DLEE0001
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCDLEE0002
C                                                                       CDLEE0003
C  SUBROUTINE DLEE CONTROLS THE MESH GENERATION ROUTINES                CDLEE0004
C                                                                       CDLEE0005
C  CALLED BY:  MAIN                                                     CDLEE0006
C                                                                       CDLEE0007
C  CALLS SUBROUTINES:  ELTHRE  ELTWO   MAX     SEQ2    TRANS   ZERO    CDLEE0008
C                                                                       CDLEE0009
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCDLEE0010
      COMMON/KNTS/NODE,IBEAM,ITWO,ITHRE,LOAD,LINES,MIN                  DLEE0011
      COMMON/PARTS/NPARTS,XMAXP(40,3),XMINP(40,3)                       DLEE0012
      COMMON/VECFNC/NFN,IFN(10C),IAPLY(100,6),COEF(100,5)               DLEE0013
      COMMON/LCAD/NL,LCASE(100),XF(100),YF(100),ZF(100),IND(100,6)      DLEE0014
      COMMON/BOUND/NBC,IBCAPL(10C,6),IBCS(100,6)                        DLEE0015
      COMMON/MID/NMID,MIDA(100,6),MIDC(100,3)                           DLEE0016
      COMMON/DEL/NDEL,IDEL(100,6)                                       DLEE0017
      COMMON/REPEAT/NR,NRT(50)                                          DLEE0018
      COMMON/REGION/IARRAY(200,6),NREG                                  DLEE0019
      DIMENSION A(KLIM), N(NLIM)                                        DLEE0020
   10 CONTINUE                                                          DLEE0021
      REWIND 11                                                         DLEE0022
C                                                                       DLEE0023
C  GET THE NEXT PART TO BE GENERATED                                    DLEE0024
C                                                                       DLEE0025
      CALL TRANS(IEND,MAXN)                                             DLEE0026
      IF(IEND.EQ.1)GO TO 40                                             DLEE0027
      NPARTS=NPARTS+1                                                   DLEE0028
C                                                                       DLEE0029
C  MAXN IS THE NUMBER OF TIMES THE PART IS TO BE REPEATED               DLEE0030
C                                                                       DLEE0031
      DO 30 NR=1,MAXN                                                   DLEE0032
      CALL ZERO(KLIM,NLIM,A,N)                                          DLEE0033
      IF(NR.GE.2)GO TO 20                                               DLEE0034
      REWIND 11                                                         DLEE0035
      NFN=0                                                             DLEE0036
      NREG=0                                                            DLEE0037
      NBC=0                                                             DLEE0038
      NDEL=0                                                            DLEE0039
```

```
      NMID=0                                                          DLEEC04C
      NL=0                                                            DLEE0041
C                                                                     DLEE0042
C  SET UP ARRAY INDICES                                               DLEE0043
C                                                                     DLEE0044
      CALL MAX(I,J,K)                                                 DLEE0045
      J1=I*J*K+1                                                      DLEE0046
      J2=J1+I*J*K                                                     DLEE0047
      J3=J2+I*J*K                                                     DLEE0048
      NODES=I*J*K                                                     DLEE0049
      REWIND 11                                                       DLEE005C
C                                                                     DLEE0051
C  READ IN THE PART AND GENERATE THE MESH                             DLEE0052
C                                                                     DLEE0053
      CALL SEQ2(I,J,K,A(1),A(J1),A(J2))                               DLEE0054
   20 CONTINUE                                                        DLEE0055
      CALL ELTWO(I,J,K,A(1),A(J1),A(J2),N)                            DLEE0056
      CALL ELTHRE(I,J,K,A(1),A(J1),A(J2),N)                           DLEE0057
   30 CONTINUE                                                        DLEE0058
      GO TO 10                                                        DLEE0059
      RETURN                                                          DLEE0060
   40 CONTINUE                                                        DLEE0061
C                                                                     DLEE0062
C  OUTPUT JOB STATISTICS                                              DLEE0063
C                                                                     DLEE0064
      NODE=NODE-1                                                     DLEE0065
      ITWO=ITWO-1                                                     DLEE0066
      ITHRE=ITHRE-1                                                   DLEE0067
      IBEAM=IBEAM-1                                                   DLEE0068
      WRITE(6,50)NODE,ITWO,ITHRE,IBEAM                                DLEE0069
      IF(LCAD.EQ.0)WRITE(13,60)LCAC                                   DLEE0070
      RETURN                                                          DLEE0071
   50 FORMAT(9H NODES = ,I5,/,16H 2-D ELEMENTS = ,I5,/                DLEE0072
     1,16H 3-D ELEMENTS = ,I5,/,9H BEAMS = ,I5)                       DLEE0073
   60 FORMAT(I5)                                                      DLEE0074
      END                                                             DLEE0075
      SUBROUTINE DOUBLE(I,J,K)                                        DBLE0001
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCDBLE0002
C                                                                     CDBLE0003
C  SUBROUTINE DOUBLE MULTIPLIES ALL THE INDICES BY 2 WHEN 8-NODE PLATESCDBLE0004
C  OR 20-NODE SOLIDS ARE TO BE GENERATED                              CDBLE0005
C                                                                     CDBLE0006
C  CALLED BY:  SEQ     SEQ2     MAX                                   CDBLE0007
C                                                                     CDBLE0008
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCDBLE0009
      DIMENSION I(16),J(16),K(16)                                      DBLE0010
      COMMON/PROP/MAT,NDOT,IC2,IT,TH,N8,N20                            DBLE0011
      IFACT=0                                                          DBLE0012
      IF(N8.EQ.1.OR.N20.EQ.1)IFACT=2                                   DBLE0013
      IF(IFACT.EQ.0)IFACT=1                                            DBLE0014
      IV=IFACT*IABS(I(1))-1                                            DBLE0015
      JV=IFACT*IABS(J(1))-1                                            DBLE0016
      KV=IFACT*IABS(K(1))-1                                            DBLE0017
      DO 10 M=1,16                                                     DBLE0018
      IF(I(M).GE.1)I(M)=I(M)*IFACT-IV                                  DBLE0019
      IF(J(M).GE.1)J(M)=J(M)*IFACT-JV                                  DBLE0020
      IF(K(M).GE.1)K(M)=K(M)*IFACT-KV                                  DBLE0021
      IF(I(M).LE.-1)I(M)=I(M)*IFACT+IV                                 DBLE0022
      IF(J(M).LE.-1)J(M)=J(M)*IFACT+JV                                 DBLE0023
      IF(K(M).LE.-1)K(M)=K(M)*IFACT+KV                                 DBLE0024
   10 CONTINUE                                                         DBLE0025
```

```
      RETURN                                                                DBLE0026
      END                                                                   DBLE0027
      SUBROUTINE ELTHRE (M1,M2,M3,X,Y,Z,N)                                  ELTH0001
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCELTH0002
C                                                                           CELTH0003
C  SUBROUTINE ELTHRE GENERATES SOLID ELEMENTS AND NODES                     CELTH0004
C                                                                           CELTH0005
C  CALLED BY:  DLEE                                                         CELTH0006
C                                                                           CELTH0007
C  CALLS SUBROUTINES:  BCS2    CTRAN2  CYLIND  DATA    DELET2  EXCHNG  CELTH0008
C       ICN1    LCADS2  MAXMIN  MESH    MIDSI2  MOVE    PLOTMK  SPHERI  CELTH0009
C                                                                           CELTH0010
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCELTH0011
      COMMON/KNTS/MODE,IBEAM,ITWO,ITHRE,LOAD,LINES,MIN                      ELTH0012
      COMMON/PROP/MAT,NDOT,IC2,IT,TH,N8,N20                                 ELTH0013
      COMMON/JUNK2/NFAST                                                    ELTH0014
      COMMON/REGION/IARRAY(200,6),NREG                                      ELTH0015
      COMMON/REPEAT/NR,NRT(50)                                              ELTH0016
      DIMENSION X(M1,M2,M3), Y(M1,M2,M3), Z(M1,M2,M3), N(M1,M2,M3)          ELTH0017
      DIMENSION IBC(6)                                                      ELTH0018
      EQUIVALENCE(ITHRE,NEL)                                                ELTH0019
C                                                                           ELTH0020
C  NODE GENERATION                                                          ELTH0021
C                                                                           ELTH0022
      DO 150 INDX=1,NREG                                                    ELTH0023
      IM=IARRAY(INDX,1)                                                     ELTH0024
      JM=IARRAY(INDX,2)                                                     ELTH0025
      KM=IARRAY(INDX,3)                                                     ELTH0026
      IX=IARRAY(INDX,4)                                                     ELTH0027
      JX=IARRAY(INDX,5)                                                     ELTH0028
      KX=IARRAY(INDX,6)                                                     ELTH0029
      IF(MAT.EQ.0)MAT=1                                                     ELTH0030
      IF(IM.EQ.IX.OR.JM.EQ.JX.OR.KM.EQ.KX)GO TO 150                         ELTH0031
      IDEL=0                                                                ELTH0032
      CALL DELET2(IM,JM,KM,IX,JX,KX,IDEL)                                   ELTH0033
      IF(IDEL.EQ.1)GO TO 150                                                ELTH0034
      IF(NR.GE.2)GO TO 10                                                   ELTH0035
C                                                                           ELTH0036
C  GENERATE NODES                                                           ELTH0037
C                                                                           ELTH0038
      CALL MESH(M1,M2,M3,X,Y,Z,N,IM,JM,KM,IX,JX,KX)                         ELTH0039
C                                                                           ELTH0040
C  MOVE NODES AS SPECIFIED IN ARC CARDS                                     ELTH0041
C                                                                           ELTH0042
      CALL MOVE(M1,M2,M3,X,Y,Z,N,IM,JM,KM,IX,JX,KX)                         ELTH0043
   10 CONTINUE                                                              ELTH0044
      DO 70 J=JM,JX                                                         ELTH0045
      DO 60 I=IM,IX                                                         ELTH0046
      DO 50 K=KM,KX                                                         ELTH0047
      IF(N(I,J,K).NE.0)GO TO 50                                             ELTH0048
      IF(NFAST.EQ.1)GO TO 30                                                ELTH0049
      ICNT=0                                                                ELTH0050
C                                                                           ELTH0051
C  CHECK FOR MIDSIDE NODAL POINT ELIMINATION                                ELTH0052
C                                                                           ELTH0053
      XM=FLOAT(MOD((I-IM),2))                                               ELTH0054
      YM=FLOAT(MOD((J-JM),2))                                               ELTH0055
      ZM=FLOAT(MOD((K-KM),2))                                               ELTH0056
      IF(XM.NE.0.0)ICNT=ICNT+1                                              ELTH0057
      IF(YM.NE.0.0)ICNT=ICNT+1                                              ELTH0058
      IF(ZM.NE.0.0)ICNT=ICNT+1                                              ELTH0059
```

```
      IF(ICNT.GE.2.AND.N20.EQ.1)GO TO 50                                ELTH0060
      INT=1                                                             ELTH0061
      IF(I.EQ.IM.OR.I.EQ.IX) INT=0                                      ELTH0062
      IF(J.EQ.JM.OR.J.EQ.JX) INT=0                                      ELTH0063
      IF(K.EQ.KM.OR.K.EQ.KX) INT=0                                      ELTH0064
      I1=I                                                              ELTH0065
      J1=J                                                              ELTH0066
      K1=K                                                              ELTH0067
      MFLAG=0                                                           ELTH0068
      CALL MIDSI2(I1,J1,K1,MFLAG,XM,YM,ZM)                              ELTH0069
      IF(MFLAG.EQ.1)GO TO 50                                            ELTH0070
      DO 20 L=1,6                                                       ELTH0071
      IBC(L)=0                                                          ELTH0072
   20 CONTINUE                                                          ELTH0073
      CALL BCS2(I1,J1,K1,IBC)                                           ELTH0074
      CALL LOADS2(I1,J1,K1,NCDE)                                        ELTH0075
   30 CONTINUE                                                          ELTH0076
      XO=X(I,J,K)                                                       ELTH0077
      YO=Y(I,J,K)                                                       ELTH0078
      ZO=Z(I,J,K)                                                       ELTH0079
      CALL EXCHNG(XO,YO,ZO)                                             ELTH0080
      IF(IC2.EQ.1)CALL CYLIND(XO,YO,ZO)                                 ELTH0081
      IF(IC2.EQ.2)CALL SPHERI(XO,YO,ZO)                                 ELTH0082
      CALL CTRAN2(XO,YO,ZO)                                             ELTH0083
      CALL MAXMIN(XO,YO,ZO)                                             ELTH0084
      NBCOUT=0                                                          ELTH0085
      DO 40 NBI=1,6                                                     ELTH0086
   40 NBCOUT=NBCOUT+IBC(NBI)*10**(NBI-1)                                ELTH0087
      CALL ICN1(XO,YO,ZO,NBCOUT)                                        ELTH0088
      N(I,J,K)=NCDE                                                     ELTH0089
      NCDE=NCDE+1                                                       ELTH0090
   50 CONTINUE                                                          ELTH0091
   60 CONTINUE                                                          ELTH0092
   70 CONTINUE                                                          ELTH0093
C                                                                       ELTH0094
C  ELEMENT GENERATION                                                   ELTH0095
C  ROUTINE PLOTMK OUTPUTS PLOTTING VECTORS                              ELTH0096
C                                                                       ELTH0097
      CALL PLOTMK(IM,JM,KM,IX,JX,KX,N20,M1,M2,M3,N)                     ELTH0098
        MIN=NODE                                                        ELTH0099
      IF(NFAST.EQ.1)GO TO 150                                           ELTH0100
      IF (IX.NE.IM)          IX=IX-1                                    ELTH0101
      IF (JX.NE.JM)          JX=JX-1                                    ELTH0102
      IF (KX.NE.KM)          KX=KX-1                                    ELTH0103
      NNODES=8                                                          ELTH0104
      IF(N20.EQ.0)GO TO 110                                             ELTH0105
      DO 100 K=KM,KX,2                                                  ELTH0106
      DO 90 J=JM,JX,2                                                   ELTH0107
      DO 80 I=IM,IX,2                                                   ELTH0108
      I2=I+2                                                            ELTH0109
      J2=J+2                                                            ELTH0110
      K2=K+2                                                            ELTH0111
      IDEL=0                                                            ELTH0112
      I1=I                                                              ELTH0113
      J1=J                                                              ELTH0114
      K1=K                                                              ELTH0115
      CALL DELET2(I1,J1,K1,I2,J2,K2,IDEL)                               ELTH0116
      IF(IDEL.EQ.1)GO TO 80                                             ELTH0117
      NNODES=20                                                         ELTH0118
      WRITE (9) NNODES,MAT,N(I,J,K),N(I+2,J,K),N(I+2,J+2,K),N(I,J+2,K),  ELTH0119
     1N(I,J,K+2),N(I+2,J,K+2),N(I+2,J+2,K+2),N(I,J+2,K+2),N(I+1,J,K),    ELTH0120
```

```
     2N(I+2,J+1,K),N(I+1,J+2,K),N(I,J+1,K),N(I+1,J,K+2),N(I+2,J+1,K+2),  ELTH0121
     3N(I+1,J+2,K+2),N(I,J+1,K+2),N(I,J,K+1),N(I+2,J,K+1),N(I+2,J+2,K+1)ELTH0122
     4,N(I,J+2,K+1)                                                      ELTH0123
      NEL=NEL+1                                                          ELTH0124
   80 CONTINUE                                                           ELTH0125
   90 CONTINUE                                                           ELTH0126
  100 CONTINUE                                                           ELTH0127
      GO TO 150                                                          ELTH0128
  110 CONTINUE                                                           ELTH0129
      DO 140 K=KM,KX                                                     ELTH0130
      DO 130 J=JM,JX                                                     ELTH0131
      DO 120 I=IM,IX                                                     ELTH0132
      I2=I+1                                                             ELTH0133
      J2=J+1                                                             ELTH0134
      K2=K+1                                                             ELTH0135
      IDEL=0                                                             ELTH0136
      I1=I                                                               ELTH0137
      J1=J                                                               ELTH0138
      K1=K                                                               ELTH0139
      IDUM=0                                                             ELTH0140
      CALL DELET2(I1,J1,K1,I2,J2,K2,IDEL)                                ELTH0141
      IF(IDEL.EQ.1)GO TO 120                                             ELTH0142
      WRITE(9)NNCODES,MAT,N(I,J,K+1),N(I,J+1,K+1),N(I+1,J+1,K+1),N(I+1,J,ELTH0143
     1K+1),N(I,J,K),N(I,J+1,K),N(I+1,J+1,K),N(I+1,J,K),IDUM,IDUM,IDUM,   ELTH0144
     2IDUM,IDUM,IDUM,IDUM,IDUM,IDUM,IDUM,IDUM,IDUM                       ELTH0145
      NEL=NEL+1                                                          ELTH0146
  120 CONTINUE                                                           ELTH0147
  130 CONTINUE                                                           ELTH0148
  140 CONTINUE                                                           ELTH0149
  150 CONTINUE                                                           ELTH0150
  160 CONTINUE                                                           ELTH0151
      RETURN                                                             ELTH0152
  170 FORMAT(7I5,3F10.4,3I5)                                             ELTH0153
  180 FORMAT(4I5,5X,I5,/,16I5,/,4I5)                                     ELTH0154
  190 FORMAT(4I5,5X,I5,/,8I5,/,5H      )                                 ELTH0155
      END                                                                ELTH0156
      SUBROUTINE ELTWO (M1,M2,M3,X,Y,Z,N)                                ELTW0001
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCELTW0002
C                                                                       CELTW0003
C  SUBROUTINE ELTWO GENERATES PLATE ELEMENTS                            CELTW0004
C                                                                       CELTW0005
C  CALLED BY:  DLEE                                                     CELTW0006
C                                                                       CELTW0007
C  CALLS SUBROUTINES:  BCS2    CTRAN2  CYLIND  DATA    DELET2  EXCHNG  CELTW0008
C       ICN1    LCADS2  MAXMIN  MESH    MIDSI2  MOVE    PLOTMK  SPHERI  CELTW0009
C                                                                       CELTW0010
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCELTW0011
      COMMON/KNTS/NODE,IBEAM,ITWO,ITHRE,LOAD,LINES,MIN                   ELTW0012
      COMMON/PROP/MAT,NDOT,IC2,IT,TH,N8,N20                              ELTW0013
      COMMON/JUNK2/NFAST                                                 ELTW0014
      COMMON/REGION/IARRAY(200,6),NREG                                   ELTW0015
      COMMON/REPEAT/NR,NRT(50)                                           ELTW0016
      DIMENSION X(M1,M2,M3), Y(M1,M2,M3), Z(M1,M2,M3), N(M1,M2,M3)       ELTW0017
      DIMENSION IBC(6)                                                   ELTW0018
      EQUIVALENCE(ITWO,NEL)                                              ELTW0019
C                                                                       ELTW0020
C  NODE GENERATION                                                      ELTW0021
C                                                                       ELTW0022
      DO 220 INDX=1,NREG                                                 ELTW0023
      IM=IARRAY(INDX,1)                                                  ELTW0024
      JM=IARRAY(INDX,2)                                                  ELTW0025
```

```
      KM=IARRAY(INCX,3)                                                 ELTW0026
      IX=IARRAY(INCX,4)                                                 ELTW0027
      JX=IARRAY(INCX,5)                                                 ELTW0028
      KX=IARRAY(INCX,6)                                                 ELTW0029
      IF(MAT.EC.0)MAT=1                                                 ELTW003C
      IF(IM.NE.IX.AND.JM.NE.JX.AND.KM.NE.KX)GO TO 220                   ELTW0031
      IDEL=0                                                            ELTW0032
      CALL DELET2(IM,JM,KM,IX,JX,KX,IDEL)                               ELTW0033
      IF(IDEL.EQ.1)GO TO 220                                            ELTW0034
      IF(NR.GE.2)GC TO 10                                               ELTW0035
C                                                                       ELTW0036
C  GENERATE NODES                                                       ELTW0037
C                                                                       ELTW0038
      CALL MESH(M1,M2,M3,X,Y,Z,N,IM,JM,KM,IX,JX,KX)                     ELTW0039
C                                                                       ELTW0040
C  MOVE NODES AS SPECIFIED IN ARC CARDS                                 ELTW0041
C                                                                       ELTW0042
      CALL MCVE(M1,M2,M3,X,Y,Z,N,IM,JM,KM,IX,JX,KX)                     ELTW0043
   10 CONTINUE                                                          ELTW0044
      DO 70 J=JM,JX                                                     ELTW0045
      DO 60 I=IM,IX                                                     ELTW0046
      DO 50 K=KM,KX                                                     ELTW0047
      IF(N(I,J,K).NE.0)GO TO 50                                         ELTW0048
      IF(NFAST.EC.1)GO TO 30                                            ELTW0049
      ICNT=0                                                            ELTW0050
C                                                                       ELTW0051
C  CHECK FOR MIDSIDE NODAL POINT ELIMINATION                            ELTW0052
C                                                                       ELTW0053
      XM=FLOAT(MOD((I-IM),2))                                           ELTW0054
      YM=FLOAT(MOD((J-JM),2))                                           ELTW0055
      ZM=FLOAT(MOD((K-KM),2))                                           ELTW0056
      IF(XM.NE.0.0)ICNT=ICNT+1                                          ELTW0057
      IF(YM.NE.0.0)ICNT=ICNT+1                                          ELTW0058
      IF(ZM.NE.0.0)ICNT=ICNT+1                                          ELTW0059
      IF(ICNT.GE.2.AND.N8.EQ.1)GC TO 50                                 ELTW006C
      DO 20 L=1,6                                                       ELTW0061
      IBC(L)=0                                                          ELTW0062
   20 CONTINUE                                                          ELTW0063
      I1=I                                                              ELTW0064
      J1=J                                                              ELTW0065
      K1=K                                                              ELTW0066
      MFLAG=0                                                           ELTW0067
      CALL MIDSI2(I1,J1,K1,MFLAG,XM,YM,ZM)                              ELTW0068
      IF(MFLAG.EC.1)GO TO 50                                            ELTW0069
      CALL BCS2(I1,J1,K1,IBC)                                           ELTW007C
      CALL LOADS2(I1,J1,K1,NODE)                                        ELTW0071
      INT=0                                                             ELTW0072
   30 CONTINUE                                                          ELTW0073
      XO=X(I,J,K)                                                       ELTW0074
      YO=Y(I,J,K)                                                       ELTW0075
      ZO=Z(I,J,K)                                                       ELTW0076
      CALL EXCHNG(XO,YO,ZO)                                             ELTW0077
      IF(IC2.EC.1)CALL CYLIND(XC,YC,ZC)                                 ELTW0078
      IF(IC2.EC.2)CALL SPHERI(XC,YC,ZC)                                 ELTW0079
      CALL CTRAN2(XC,YC,ZC)                                             ELTW008C
      CALL MAXMIN(XC,YC,ZC)                                             ELTW0081
      NBCOUT=0                                                          ELTW0082
      DO 40 NBI=1,6                                                     ELTW0083
   40 NBCOUT=NBCOUT+IBC(NBI)*10**(NBI-1)                                ELTW0084
      CALL ICN1(XC,YC,ZC,NBCOUT)                                        ELTW0085
      N(I,J,K)=NODE                                                     ELTW0086
```

```
      NODE=NODE+1                                                       ELTWOCE7
   50 CONTINUE                                                          ELTWOC88
   60 CONTINUE                                                          ELTWOC89
   70 CONTINUE                                                          ELTWOC90
C                                                                       ELTWOC91
C  ELEMENT GENERATION                                                   ELTWOC92
C  ROUTINE PLOTMK OUTPUTS PLOTTING VECTORS                              ELTWOC93
C                                                                       ELTWOC94
      CALL PLOTMK(IM,JM,KM,IX,JX,KX,N8,M1,M2,M3,N)                      ELTWOC95
        MIN=NODE                                                        ELTWOC96
      IF(NFAST.EC.1)GO TO 220                                           ELTWOC97
      NNODES=4                                                          ELTWOC98
      IF(N8.EQ.1)NNODES=8                                               ELTWOC99
      IF(KM.LT.KX)GC TO 80                                              ELTWO1CC
C                                                                       ELTWO1C1
C  K-INDEX IS CONSTANT                                                  ELTWO1C2
C                                                                       ELTWO1C3
      I2=2                                                              ELTWC1C4
      J2=0                                                              ELTWO1C5
      K2=0                                                              ELTWO1C6
      I3=2                                                              ELTWO1C7
      J3=2                                                              ELTWO1C8
      K3=0                                                              ELTWO1C9
      I4=0                                                              ELTWO110
      J4=2                                                              ELTWC111
      K4=0                                                              ELTWO112
      IF (NNODES.EC.4)         GC TC 100                                ELTWO113
      I5=1                                                              ELTWC114
      J5=0                                                              ELTWO115
      K5=0                                                              ELTWO116
      I6=2                                                              ELTWO117
      J6=1                                                              ELTWO118
      K6=0                                                              ELTWC119
      I7=1                                                              ELTWO120
      J7=2                                                              ELTWC121
      K7=0                                                              ELTWO122
      I8=0                                                              ELTWO123
      J8=1                                                              ELTWC124
      K8=0                                                              ELTWO125
      GC TO 110                                                         ELTWC126
   80 IF (JM.LT.JX)         GO TC 9C                                    ELTWO127
C                                                                       ELTWO128
C  J-INDEX IS CONSTANT                                                  ELTWC129
C                                                                       ELTWO130
      I2=0                                                              ELTWC131
      J2=0                                                              ELTWO132
      K2=2                                                              ELTWO133
      I3=2                                                              ELTWC134
      J3=0                                                              ELTWO135
      K3=2                                                              ELTWO136
      I4=2                                                              ELTWO137
      J4=0                                                              ELTWO138
      K4=0                                                              ELTWC139
      IF (NNODES.EC.4)         CC TC 100                                ELTWO140
      I5=0                                                              ELTWC141
      J5=0                                                              ELTWO142
      K5=1                                                              ELTWO143
      I6=1                                                              ELTWO144
      J6=0                                                              ELTWO145
      K6=2                                                              ELTWO146
      I7=2                                                              ELTWO147
```

```
      J7=0                                                          ELTWO148
      K7=1                                                          ELTWO149
      I8=1                                                          ELTWO150
      J8=0                                                          ELTWO151
      K8=0                                                          ELTWO152
      GO TO 110                                                     ELTWO153
   90 CONTINUE                                                      ELTWO154
C                                                                   ELTWO155
C  I-INDEX IS CONSTANT                                              ELTWO156
C                                                                   ELTWO157
      I2=0                                                          ELTWO158
      J2=2                                                          ELTWO159
      K2=0                                                          ELTWO160
      I3=0                                                          ELTWO161
      J3=2                                                          ELTWO162
      K3=2                                                          ELTWO163
      I4=0                                                          ELTWO164
      J4=0                                                          ELTWO165
      K4=2                                                          ELTWO166
      IF (NNODES.EQ.4)          GO TO 100                           ELTWO167
      I5=0                                                          ELTWO168
      J5=1                                                          ELTWO169
      K5=0                                                          ELTWO170
      I6=0                                                          ELTWO171
      J6=2                                                          ELTWO172
      K6=1                                                          ELTWO173
      I7=0                                                          ELTWO174
      J7=1                                                          ELTWO175
      K7=2                                                          ELTWO176
      I8=0                                                          ELTWO177
      J8=0                                                          ELTWO178
      K8=1                                                          ELTWO179
      GO TO 110                                                     ELTWO180
  100 CONTINUE                                                      ELTWO181
      I2=I2/2                                                       ELTWO182
      J2=J2/2                                                       ELTWO183
      K2=K2/2                                                       ELTWO184
      I3=I3/2                                                       ELTWO185
      J3=J3/2                                                       ELTWO186
      K3=K3/2                                                       ELTWO187
      I4=I4/2                                                       ELTWO188
      J4=J4/2                                                       ELTWO189
      K4=K4/2                                                       ELTWO190
  110 CONTINUE                                                      ELTWO191
      IF (IX.NE.IM)          IX=IX-1                                ELTWO192
      IF (JX.NE.JM)          JX=JX-1                                ELTWO193
      IF (KX.NE.KM)          KX=KX-1                                ELTWO194
      IF (NNODES.EQ.4)           GO TO 150                         ELTWO195
C                                                                   ELTWO196
C                                                                   ELTWO197
      DO 140 K=KM,KX,2                                              ELTWO198
      DO 130 J=JM,JX,2                                              ELTWO199
      DO 120 I=IM,IX,2                                              ELTWO200
      IDEL=0                                                        ELTWO201
      IC=I+I3                                                       ELTWO202
      JC=J+J3                                                       ELTWO203
      KC=K+K3                                                       ELTWO204
      I1=I                                                          ELTWO205
      J1=J                                                          ELTWO206
      K1=K                                                          ELTWO207
      CALL DELET2(I1,J1,K1,IC,JC,KC,IDEL)                           ELTWO208
```

```
      IF(IDEL.EQ.1)GO TO 120                                                ELTWC209
      WRITE (8) N(I,J,K),N(I+I2,J+J2,K+K2),N(I+I3,J+J3, K+K3),N(I+I4,J+ ELTWC210
     1J4,K+K4),MAT,IT,TH,N(I+I5,J+J5,K+K5) ,N(I+I6,J+J6,K+K6),N(I+I7,J+ ELTWC211
     2J7,K+K7),N(I+I8,J+J8,K+K8)                                          ELTWC212
      NEL=NEL+1                                                             ELTWC213
  120 CONTINUE                                                              ELTWC214
  130 CONTINUE                                                              ELTWC215
  140 CONTINUE                                                              ELTWC216
      GO TO 220                                                             ELTWC217
  150 CONTINUE                                                              ELTWC218
      IDUM=0                                                                ELTWC219
      DO 210 K=KM,KX                                                        ELTWC220
      DO 200 J=JM,JX                                                        ELTWC221
      DO 190 I=IM,IX                                                        ELTWC222
      IDEL=0                                                                ELTWC223
      IC=I+I3                                                               ELTWC224
      JC=J+J3                                                               ELTWC225
      KC=K+K3                                                               ELTWC226
      I1=I                                                                  ELTWC227
      J1=J                                                                  ELTWC228
      K1=K                                                                  ELTWC229
      CALL DELET2(I1,J1,K1,IC,JC,KC,IDEL)                                   ELTWC230
      IF(IDEL.EQ.1)GO TO 190                                                ELTWC231
      IF(NDOT.EQ.0)GO TO 180                                                ELTWC232
C                                                                           ELTWC233
C  CHECK FOR 180 DEGREE ANGLE BETWEEN EDGES                                 ELTWC234
C                                                                           ELTWC235
      ND1=N(I,J,K)                                                          ELTWC236
      ND2=N(I+I2,J+J2,K+K2)                                                 ELTWC237
      ND3=N(I+I3,J+J3,K+K3)                                                 ELTWC238
      ND4=N(I+I4,J+J4,K+K4)                                                 ELTWC239
      XD1=X(I+I2,J+J2,K+K2)-X(I,J,K)                                        ELTWC240
      XD2=X(I+I3,J+J3,K+K3)-X(I+I2,J+J2,K+K2)                               ELTWC241
      XD3=X(I+I4,J+J4,K+K4)-X(I+I3,J+J3,K+K3)                               ELTWC242
      XD4=X(I,J,K)-X(I+I4,J+J4,K+K4)                                        ELTWC243
      YD1=Y(I+I2,J+J2,K+K2)-Y(I,J,K)                                        ELTWC244
      YD2=Y(I+I3,J+J3,K+K3)-Y(I+I2,J+J2,K+K2)                               ELTWC245
      YD3=Y(I+I4,J+J4,K+K4)-Y(I+I3,J+J3,K+K3)                               ELTWC246
      YD4=Y(I,J,K)-Y(I+I4,J+J4,K+K4)                                        ELTWC247
      ZD1=Z(I+I2,J+J2,K+K2)-Z(I,J,K)                                        ELTWC248
      ZD2=Z(I+I3,J+J3,K+K3)-Z(I+I2,J+J2,K+K2)                               ELTWC249
      ZD3=Z(I+I4,J+J4,K+K4)-Z(I+I3,J+J3,K+K3)                               ELTWC250
      ZD4=Z(I,J,K)-Z(I+I4,J+J4,K+K4)                                        ELTWC251
      V1=SQRT(XD1*XD1+YD1*YD1+ZD1*ZD1)                                      ELTWC252
      V2=SQRT(XD2*XD2+YD2*YD2+ZD2*ZD2)                                      ELTWC253
      V3=SQRT(XD3*XD3+YD3*YD3+ZD3*ZD3)                                      ELTWC254
      V4=SQRT(XD4*XD4+YD4*YD4+ZD4*ZD4)                                      ELTWC255
      AD1=(XD1*XD2+YD1*YD2+ZD1*ZD2)/(V1*V2)                                 ELTWC256
      AD2=(XD2*XD3+YD2*YD3+ZD2*ZD3)/(V2*V3)                                 ELTWC257
      AD3=(XD3*XD4+YD3*YD4+ZD3*ZD4)/(V3*V4)                                 ELTWC258
      AD4=(XD4*XD1+YD4*YD1+ZD4*ZD1)/(V4*V1)                                 ELTWC259
      IF(ABS(AD1).GT.0.94.OR.ABS(AD3).GT.0.94)GO TO 160                     ELTWC260
      IF(ABS(AD2).GT.0.94.OR.ABS(AD4).GT.0.94)GO TO 170                     ELTWC261
      GO TO 180                                                             ELTWC262
  160 CONTINUE                                                              ELTWC263
      WRITE(8)ND1,ND2,ND4,IDUM,MAT,IT,TH,IDUM,IDUM,IDUM,IDUM                ELTWC264
      NEL=NEL+1                                                             ELTWC265
      WRITE(8)ND2,ND3,ND4,IDUM,MAT,IT,TH,IDUM,IDUM,IDUM,IDUM                ELTWC266
      NEL=NEL+1                                                             ELTWC267
      GO TO 190                                                             ELTWC268
  170 CONTINUE                                                              ELTWC269
```

```
      WRITE(8)ND1,ND2,ND3,IDUM,MAT,IT,TH,IDUM,IDUM,IDUM,IDUM              ELTW0270
      NEL=NEL+1                                                           ELTW0271
      WRITE(8)ND3,ND4,ND1,IDUM,MAT,IT,TH,IDUM,IDUM,IDUM,IDUM             ELTW0272
      NEL=NEL+1                                                           ELTW0273
      GO TO 190                                                           ELTW0274
  180 CONTINUE                                                            ELTW0275
      WRITE (8) N(I,J,K),N(I+I2,J+J2,K+K2),N(I +I3,J+J3,K+K3),N(I+I4,J+ ELTW0276
     1J4,K+K4),MAT,IT,TH,IDUM,IDUM,IDUM,IDUM                              ELTW0277
      NEL=NEL+1                                                           ELTW0278
  190 CONTINUE                                                            ELTW0279
  200 CONTINUE                                                            ELTW0280
  210 CONTINUE                                                            ELTW0281
  220 CONTINUE                                                            ELTW0282
  230 CONTINUE                                                            ELTW0283
      RETURN                                                              ELTW0284
  240 FORMAT (7I5,3F10.4,3I5)                                             ELTW0285
  250 FORMAT(5I5,5X,I5,5X,F10.4,4I5)                                      ELTW0286
  260 FORMAT(4I5,10X,I5,5X,F10.4)                                         ELTW0287
      END                                                                 ELTW0288
      SUBROUTINE EL1D(IOUT,NEL2)                                          EL1D0001
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCEL1D0002
C                                                                         CEL1D0003
C  SUBROUTINE EL1D CONVERTS BEAM ELEMENTS TO THEIR PROPER FORMAT          CEL1D0004
C                                                                         CEL1D0005
C  CALLED BY:  OUTPUT                                                     CEL1D0006
C                                                                         CEL1D0007
C  CALLS SUBROUTINES:  BCS3                                               CEL1D0008
C                                                                         CEL1D0009
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCEL1D0010
      DIMENSION IB(6),I(2),ID(2)                                          EL1D0011
      COMMON/SYSARY/X(8000),Y(8000),Z(8000),N(8000)                       EL1D0012
     1,IBB(8000),IFLAG(8000),V(8000,3)                                    EL1D0013
      COMMON/JUNK/LCOF,IAXI,MOVIE,IE1,IE2,IE3,NPLOT,TITLE(20),IOUT2       EL1D0014
      NEL2=1                                                              EL1D0015
      NEL=1                                                               EL1D0016
      IF(IOUT.EQ.3)WRITE(14,50)                                           EL1D0017
   10 READ(10,END=40)I,ID                                                 EL1D0018
      IF(I(1).EQ.0)GO TO 10                                               EL1D0019
      IF(ID(1).NE.0)ID(1)=N(ID(1))                                        EL1D0020
      DO 30 J=1,2                                                         EL1D0021
      I(J)=N(I(J))                                                        EL1D0022
      CALL BCS3(IB,I(J),1)                                                EL1D0023
      DO 20 K=4,6                                                         EL1D0024
      IB(K)=MAXO(IB(K),2)                                                 EL1D0025
   20 CONTINUE                                                            EL1D0026
      CALL BCS3(IB,I(J),0)                                                EL1D0027
   30 CONTINUE                                                            EL1D0028
      IF(IOUT.EQ.1)WRITE(14,70)NEL,I,ID                                   EL1D0029
      IF(MOVIE.EQ.1)WRITE(14,70)NEL,I                                     EL1D0030
      IF(IOUT.EQ.2)WRITE(14,80)NEL2,I,ID                                  EL1D0031
      IF(IOUT.EQ.3)WRITE(14,60)NEL2,ID(2),I                               EL1D0032
      NEL2=NEL2+1                                                         EL1D0033
      NEL=NEL+1                                                           EL1D0034
      GO TO 10                                                            EL1D0035
   40 CONTINUE                                                            EL1D0036
      RETURN                                                              EL1D0037
   50 FORMAT(8HELEMENTS,/,39HNUMBER ELEMENT.TYPE PROPERTIES               EL1D0038
     1 TOPOLOGY)                                                          EL1D0039
   60 FORMAT(I5,7H 34000 ,3I5)                                            EL1D0040
   70 FORMAT(6I5)                                                         EL1D0041
   80 FORMAT(I5,4X,1H2,3I5,25X,2I5)                                       EL1D0042
```

```
      END                                                                EL1D0043
      SUBROUTINE EL2D(IOUT,NEL2)                                         EL2D0001
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCEL2D0002
C                                                                        CEL2D0003
C  SUBROUTINE EL2D CONVERTS PLATE ELEMENTS TO THEIR PROPER FORMAT        CEL2D0004
C                                                                        CEL2D0005
C  CALLED BY:  OUTPUT                                                    CEL2D0006
C                                                                        CEL2D0007
C  CALLS SUBROUTINES:  BCS3    FLAT                                      CEL2D0008
C                                                                        CEL2D0009
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCEL2D0010
      DIMENSION IB(6),I(8)                                               EL2D0011
      COMMON/SYSARY/X(8000),Y(8000),Z(8000),N(8000)                      EL2D0012
     1,IBB(8000),IFLAG(8000),V(8000,3)                                   EL2D0013
      COMMON/JUNK/LCOF,IAXI,MOVIE,IE1,IE2,IE3,NPLOT,TITLE(20),IOUT2      EL2D0014
      NEL=1                                                              EL2D0015
      IF(NEL2.LE.1.AND.IOUT.EQ.3)WRITE(14,50)                            EL2D0016
   10 READ(8,END=40)(I(J),J=1,4),MAT,IT,TH,(I(J),J=5,8)                  EL2D0017
        IF(MAT.EQ.0)MAT=1                                                EL2D0018
        IF(IT.EQ.0)IT=1                                                  EL2D0019
        IF(TH.EQ.0.0)TH=1.0                                              EL2D0020
      IF(I(1).EQ.0)GO TO 10                                              EL2D0021
      IF(LCOF.NE.1)CALL FLAT(I)                                          EL2D0022
      CALL NORM(I)                                                       EL2D0023
      DO 30 J=1,8                                                        EL2D0024
      IF(I(J).EQ.0)GO TO 30                                              EL2D0025
      I(J)=N(I(J))                                                       EL2D0026
      CALL BCS3(IB,I(J),1)                                               EL2D0027
      DO 20 K=4,6                                                        EL2D0028
      IB(K)=MAXO(IB(K),2)                                                EL2D0029
   20 CONTINUE                                                           EL2D0030
      CALL BCS3(IB,I(J),0)                                               EL2D0031
   30 CONTINUE                                                           EL2D0032
      IF(IOUT.EQ.1)WRITE(14,70)NEL,(I(J),J=1,4),MAT,TH                   EL2D0033
      IF(MOVIE.EQ.1)WRITE(21,70)NEL,(I(J),J=1,4)                         EL2D0034
      IF(NEL2.EQ.0)NEL2=1                                                EL2D0035
      IF(IOUT.EQ.2)WRITE(14,80)NEL2,I,MAT,IT                             EL2D0036
      IF(LCOF.EQ.0)IELTYP=4421C                                          EL2D0037
      IF(I(8).EQ.0)IELTYP=4420C                                          EL2D0038
      IF(I(4).EQ.0)IELTYP=4410C                                          EL2D0039
      IF(LCOF.EQ.1)IELTYP=4321C                                          EL2D0040
      IF(IAXI.EQ.1)IELTYP=3661C                                          EL2D0041
      IF(IOUT.EQ.3)WRITE(14,60)NEL2,IELTYP,IT,I(1),I(2),I(4),I(3),I(5),  EL2D0042
     1I(8),I(6),I(7)                                                     EL2D0043
      NEL2=NEL2+1                                                        EL2D0044
      NEL=NEL+1                                                          EL2D0045
      GO TO 10                                                           EL2D0046
   40 CONTINUE                                                           EL2D0047
      RETURN                                                             EL2D0048
   50 FORMAT(8HELEMENTS,/,39HNUMBER ELEMENT.TYPE PROPERTIES              EL2D0049
     1 TOPOLOGY)                                                         EL2D0050
   60 FORMAT(I5,1X,I5,1X,9I5)                                            EL2D0051
   70 FORMAT(5I5,5X,I5,5X,F10.4,4I5)                                     EL2D0052
   80 FORMAT(I5,4X,1H6,10I5)                                             EL2D0053
      END                                                                EL2D0054
      SUBROUTINE EL3D(IOUT,NEL2)                                         EL3D0001
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCEL3D0002
C                                                                        CEL3D0003
C  SUBROUTINE EL3D CONVERTS SOLID ELEMENTS TO THEIR PROPER FORMAT        CEL3D0004
C                                                                        CEL3D0005
C  CALLED BY:  OUTPUT                                                    CEL3D0006
```

```
C                                                                       CEL3D00C7
C  CALLS SUBROUTINES:  BCS3     VCLUME                                  CEL3D00C8
C                                                                       CEL3D00C9
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCEL3D01C
      DIMENSION IB(6),I(20)                                             EL3D0011
      COMMON/SYSARY/X(8000),Y(8000),Z(8000),N(8000)                     EL3D0012
     1,I3B(8000),IFLAG(8000),V(8000,3)                                  EL3D0013
      COMMON/JUNK/LCOF,IAXI,MOVIE,IE1,IE2,IE3,NPLOT,TITLE(20),IOUT2     EL3D0014
      NEL=1                                                             EL3D0015
      IF(NEL2.LE.0.AND.IOUT.EQ.3)WRITE(14,50)                           EL3D0016
   10 READ(9,END=40)NNODES,IMAT,I                                       EL3D0017
      CALL VCLUME(I,IOUT)                                               EL3D0018
      IF(I(1).EQ.0)GO TO 10                                             EL3D0019
      NNODES=0                                                          EL3D0020
      DO 30 J=1,20                                                      EL3D0021
      IF(I(J).EQ.0)GO TO 30                                             EL3D0022
      NNODES=NNODES+1                                                   EL3D0023
      CALL BCS3(IB,I(J),1)                                              EL3D0024
      DO 20 K=4,6                                                       EL3D0025
      IB(K)=MAXO(IB(K),1)                                               EL3D0026
   20 CONTINUE                                                          EL3D0027
      CALL BCS3(IB,I(J),0)                                              EL3D0028
      I(J)=N(I(J))                                                      EL3D0029
   30 CONTINUE                                                          EL3D0030
      IF(IOUT.EQ.1)WRITE(14,70)NEL,NNODES,IMAT,I                        EL3D0031
      IF(MOVIE.EQ.1)WRITE(21,70)NEL,NNODES,IMAT,I                       EL3D0032
      IF(NEL2.EQ.0)NEL2=1                                               EL3D0033
      IF(IOUT.EQ.2)WRITE(14,80)NEL2,(I(J),J=1,8),IMAT,(I(J),J=9,20)     EL3D0034
      IELTYP=37110                                                      EL3D0035
      IF(NNODES.EQ.8)IELTYP=37100                                       EL3D0036
      IF(IOUT.EQ.3)WRITE(14,60)NEL2,IELTYP,IMAT,I(1),I(2),I(4),I(3),I(5)EL3D0037
     1,I(6),I(8),I(7),I(9),I(12),I(10),I(11),I(17),I(18),I(20),I(19),   EL3D0038
     2I(13),I(16),I(14),I(15)                                           EL3D0039
      NEL2=NEL2+1                                                       EL3D0040
      NEL=NEL+1                                                         EL3D0041
      GO TO 10                                                          EL3D0042
   40 CONTINUE                                                          EL3D0043
      RETURN                                                            EL3D0044
   50 FORMAT(8HELEMENTS,/,39HNUMBER ELEMENT.TYPE PROPERTIES             EL3D0045
     1 TOPOLOGY)                                                        EL3D0046
   60 FORMAT(I5,1X,I5,1X,9I5,/,1H*,12I5)                                EL3D0047
   70 FORMAT(2I5,5X,I5,/,16I5,/,4I5)                                    EL3D0048
   80 FORMAT(I5,3X,2H10,9I5,/,12I5)                                     EL3D0049
      END                                                               EL3D0050
      SUBROUTINE EXCHNG(X,Y,Z)                                          EXCH0001
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCEXCH0002
C                                                                       CEXCH003
C  SUBROUTINE EXCHNG PERFORMS A PERMUTATION OF THE COORDINATE AXES      CEXCH004
C                                                                       CEXCH005
C  CALLED BY:  ELTHRE  ELTWO                                            CEXCH006
C                                                                       CEXCH007
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCEXCH008
      DIMENSION XT(3)                                                   EXCH0009
      COMMON/JUNK/LCOF,IAXIS,MOVIE,IE1,IE2,IE3,NPLOT,TITLE(20),IOUT2    EXCH0010
      XT(IE1)=X                                                         EXCH0011
      XT(IE2)=Y                                                         EXCH0012
      XT(IE3)=Z                                                         EXCH0013
      X=XT(1)                                                           EXCH0014
      Y=XT(2)                                                           EXCH0015
      Z=XT(3)                                                           EXCH0016
      RETURN                                                            EXCH0017
```

```
      END                                                                EXCH0018
      SUBROUTINE FLAT(I)                                                 FLAT0001
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCFLAT0002
C                                                                        CFLAT0003
C  SUBROUTINE FLAT MOVES THE MIDSIDE NODES OF A PLATE IN THE DIRECTION   CFLAT0004
C  NORMAL TO THE PLATE UNTIL THE PLATE LIES IN A PLANE                   CFLAT0005
C                                                                        CFLAT0006
C  CALLED BY:  EL20                                                      CFLAT0007
C                                                                        CFLAT0008
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCFLAT0009
      DIMENSION I(8)                                                     FLAT0010
      COMMON/SYSARY/X(8000),Y(8000),Z(8000),N(8000)                      FLAT0011
     1,IBB(8000),IFLAG(8000),V(8000,3)                                   FLAT0012
C                                                                        FLAT0013
C  FIND NORMAL DIRECTION                                                 FLAT0014
C                                                                        FLAT0015
      XV1=X(I(4))-X(I(1))                                                FLAT0016
      YV1=Y(I(4))-Y(I(1))                                                FLAT0017
      ZV1=Z(I(4))-Z(I(1))                                                FLAT0018
      XV2=X(I(2))-X(I(1))                                                FLAT0019
      YV2=Y(I(2))-Y(I(1))                                                FLAT0020
      ZV2=Z(I(2))-Z(I(1))                                                FLAT0021
      VX=YV1*ZV2-YV2*ZV1                                                 FLAT0022
      VY=XV2*ZV1-XV1*ZV2                                                 FLAT0023
      VZ=XV1*YV2-XV2*YV1                                                 FLAT0024
      D=SQRT(VX*VX+VY*VY+VZ*VZ)                                          FLAT0025
      UX=VX/D                                                            FLAT0026
      UY=VY/D                                                            FLAT0027
      UZ=VZ/D                                                            FLAT0028
C                                                                        FLAT0029
C  MOVE MIDSIDE NODES                                                    FLAT0030
C                                                                        FLAT0031
      IF(I(5).EQ.0)GO TO 10                                              FLAT0032
      VX=X(I(5))-X(I(1))                                                 FLAT0033
      VY=Y(I(5))-Y(I(1))                                                 FLAT0034
      VZ=Z(I(5))-Z(I(1))                                                 FLAT0035
      DOT=VX*UX+VY*UY+VZ*UZ                                              FLAT0036
      X(I(5))=X(I(5))-DOT*UX                                             FLAT0037
      Y(I(5))=Y(I(5))-DOT*UY                                             FLAT0038
      Z(I(5))=Z(I(5))-DOT*UZ                                             FLAT0039
   10 IF(I(6).EQ.0)GO TO 20                                              FLAT0040
      VX=X(I(6))-X(I(1))                                                 FLAT0041
      VY=Y(I(6))-Y(I(1))                                                 FLAT0042
      VZ=Z(I(6))-Z(I(1))                                                 FLAT0043
      DOT=VX*UX+VY*UY+VZ*UZ                                              FLAT0044
      X(I(6))=X(I(6))-DOT*UX                                             FLAT0045
      Y(I(6))=Y(I(6))-DOT*UY                                             FLAT0046
      Z(I(6))=Z(I(6))-DOT*UZ                                             FLAT0047
   20 IF(I(7).EQ.0)GO TO 30                                              FLAT0048
      VX=X(I(7))-X(I(1))                                                 FLAT0049
      VY=Y(I(7))-Y(I(1))                                                 FLAT0050
      VZ=Z(I(7))-Z(I(1))                                                 FLAT0051
      DOT=VX*UX+VY*UY+VZ*UZ                                              FLAT0052
      X(I(7))=X(I(7))-DOT*UX                                             FLAT0053
      Y(I(7))=Y(I(7))-DOT*UY                                             FLAT0054
      Z(I(7))=Z(I(7))-DOT*UZ                                             FLAT0055
   30 IF(I(8).EQ.0)GO TO 40                                              FLAT0056
      VX=X(I(8))-X(I(1))                                                 FLAT0057
      VY=Y(I(8))-Y(I(1))                                                 FLAT0058
      VZ=Z(I(8))-Z(I(1))                                                 FLAT0059
      DOT=VX*UX+VY*UY+VZ*UZ                                              FLAT0060
```

```
      X(I(8))=X(I(8))-DOT*UX                                                FLAT0061
      Y(I(8))=Y(I(8))-DOT*UY                                                FLAT0062
      Z(I(8))=Z(I(8))-DOT*UZ                                                FLAT0063
   40 CONTINUE                                                              FLAT0064
      RETURN                                                                FLAT0065
      END                                                                   FLAT0066
      SUBROUTINE FN1(N1,N2,N3,S,XA,YA)                                      FN10001
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCFN10002
C                                                                          CFN10003
C  SUBROUTINE FN1 PERFORMS A QUADRATIC INTERPOLATION BETWEEN 3 NODES       CFN10004
C  USING THE PARAMETER S                                                   CFN10005
C                                                                          CFN10006
C  CALLED BY:  INPLOT                                                      CFN10007
C                                                                          CFN10008
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCFN10009
      COMMON/SYSARY/A(36000),B(36000)                                      FN10010
      S1=(S*S-S)/2.0                                                       FN10011
      S2=1-S*S                                                             FN10012
      S3=(S*S+S)/2.0                                                       FN10013
      XA=A(N1)*S1+A(N2)*S2+A(N3)*S3                                        FN10014
      YA=B(N1)*S1+B(N2)*S2+B(N3)*S3                                        FN10015
      RETURN                                                               FN10016
      END                                                                  FN10017
      SUBROUTINE FN2(N1,N2,N3)                                             FN20001
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCFN20002
C                                                                          CFN20003
C  SUBROUTINE FN2 DETERMINES IF 3 NODES LIE ALONG A STRAIGHT LINE          CFN20004
C                                                                          CFN20005
C  CALLED BY:  INPLOT                                                      CFN20006
C                                                                          CFN20007
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCFN20008
      COMMON/SYSARY/A(36000),B(36000)                                      FN20009
      IF(N2.EQ.0)RETURN                                                    FN20010
      PA=A(N2)-A(N1)                                                       FN20011
      PB=B(N2)-B(N1)                                                       FN20012
      QA=A(N3)-A(N1)                                                       FN20013
      QB=B(N3)-B(N1)                                                       FN20014
      QM=QA*QA+QB*QB                                                       FN20015
      PM=PA*PA+PB*PB                                                       FN20016
      T=PM*QM                                                              FN20017
      IF(T.LE.C.COC2)GO TO 10                                              FN20018
      D1=PA*QA+PB*QB                                                       FN20019
      DOT=D1*D1/T                                                          FN20020
      IF(DOT.GT.0.9995)N2=0                                                FN20021
      RETURN                                                               FN20022
   10 N2=0                                                                 FN20023
      RETURN                                                               FN20024
      END                                                                  FN20025
      SUBROUTINE FN3                                                       FN30001
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCFN30002
C                                                                          CFN30003
C  SUBROUTINE FN3 CONVERTS FROM GLOBAL INDEX SPACE TO LOCAL INDEX SPACE    CFN30004
C                                                                          CFN30005
C  CALLED BY:  SEQ2                                                        CFN30006
C                                                                          CFN30007
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCFN30008
      COMMON/SECD1/IB(6),I(16),J(16),K(16)                                 FN30009
      IB(1)=I(IB(1))                                                       FN30010
      IB(2)=J(IB(2))                                                       FN30011
      IB(3)=K(IB(3))                                                       FN30012
      IB(4)=I(IB(4))                                                       FN30013
```

```
      IB(5)=J(IB(5))                                                    FN3CC14
      IB(6)=K(IB(6))                                                    FN3CC15
      RETURN                                                            FN3CC16
      END                                                               FN3CC17
      SUBROUTINE FN4(X,Y,Z,XMAX),XMIN,XDIF,NDIR,NPCS)                   FN40001
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCFN40002
C                                                                       CFN40003
C  SUBROUTINE FN4 DETERMINES THE RELATIVE POSITION OF NODE X,Y,Z        CFN40004
C  IN THE STRUCTURE ALONG THE DIRECTION IN NDIR.                        CFN40005
C                                                                       CFN40006
C  CALLED BY:  OUTPUT                                                   CFN40007
C                                                                       CFN40008
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCFN40009
      DIMENSION XMAX(3),XMIN(3),XDIF(3)                                 FN40010
      IF(NDIR.NE.1)GO TO 10                                             FN40011
      POS=(X-XMIN(1))/XDIF(1)                                           FN40012
      GO TO 30                                                          FN40013
   10 IF(NDIR.NE.2)GO TO 20                                             FN40014
      POS=(Y-XMIN(2))/XDIF(2)                                           FN40015
      GO TO 30                                                          FN40016
   20 POS=(Z-XMIN(3))/XDIF(3)                                           FN40017
   30 CONTINUE                                                          FN40018
      NPCS=IFIX(POS)+1                                                  FN40019
      RETURN                                                            FN40020
      END                                                               FN40021
      SUBROUTINE INFO                                                   INFC0001
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCINFC0002
C                                                                       CINFC0003
C  SUBROUTINE INFO PRINTS OUT USEFUL INFORMATION                        CINFC0004
C                                                                       CINFC0005
C  CALLED BY:  INPLOT                                                   CINFC0006
C                                                                       CINFC0007
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCINFC0008
      COMMON/KNTS/NODE,IBEAM,ITHC,ITHRE,LOAD,LINES,MIN                  INFC0009
      COMMON/JUNK/LCOF,IAXI,MOVIE,IE1,IE2,IE3,NPLOT,TITLE(20),IOUT2     INFC0010
      COMMON/JUNK2/NFAST                                                INFC0011
      COMMON/PARTS/NPARTS,XMAXP(40,3),XMINP(40,3)                       INFC0012
      WRITE(5,10)TITLE                                                  INFC0013
      WRITE(5,20)NPARTS                                                 INFC0014
      WRITE(5,30)NODE,IBEAM,ITHC,ITHRE                                  INFC0015
      RETURN                                                            INFC0016
   10 FORMAT(20A4)                                                      INFC0017
   20 FORMAT(14H # OF PARTS = ,I2)                                      INFC0018
   30 FORMAT(14H # OF NODES = ,I6,/,14H # OF BEAMS = ,I3,/,             INFC0019
     115H # OF PLATES = ,I4,/,15H # OF SOLIDS = ,I4)                    INFC0020
      END                                                               INFC0021
      SUBROUTINE INIT(IOUT,IPLOT,TOLER,IMESH)                           INITC001
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCINITC002
C                                                                       CINITC003
C  SUBROUTINE INIT READS IN THE INITIAL CONTROL CARDS                   CINITC004
C                                                                       CINITC005
C  CALLED BY:  MAIN                                                     CINITC006
C                                                                       CINITC007
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCINITC008
      DIMENSION CARD(20),CHAR(12)                                       INITC009
      COMMON/KNTS/NODE,IBEAM,ITHC,ITHRE,LOAD,LINES,MIN                  INITC010
      COMMON/JUNK/LCOF,IAXI,MOVIE,IE1,IE2,IE3,NPLOT,TITLE(20),IOUT2     INITC011
      COMMON/JUNK2/NFAST                                                INITC012
      COMMON/PARTS/NPARTS,XMAXP(40,3),XMINP(40,3)                       INITC013
      DATA CHAR/4HSAP5,4HSAP6,4HPAPE,4HCOOR,4HBEAM,4HPLOT,4HTOLE,       INITC014
     14HAXIS,4HLCOF,4HMOVI,4HE>CH,4HFAST/,CMT/1HC/                      INITC015
```

```
      NPARTS=0                                                          INIT0016
      NODE=1                                                            INIT0017
      NPLOT=0                                                           INIT0018
      IBEAM=1                                                           INIT0019
      IE1=1                                                             INIT0020
      IE2=2                                                             INIT0021
      IE3=3                                                             INIT0022
      ITWO=1                                                            INIT0023
      ITHRE=1                                                           INIT0024
      IMESH=1                                                           INIT0025
      ICUT=0                                                            INIT0026
        MIN=0                                                           INIT0027
      NFAST=0                                                           INIT0028
      IPLOT=0                                                           INIT0029
      TOLER=0.0                                                         INIT0030
      MOVIE=0                                                           INIT0031
      LOOP=0                                                            INIT0032
      IAXI=0                                                            INIT0033
      READ(12,160)TITLE                                                 INIT0034
      WRITE(6,160)TITLE                                                 INIT0035
   10 CONTINUE                                                          INIT0036
C                                                                       INIT0037
C  READ COMMAND CARDS                                                   INIT0038
C                                                                       INIT0039
      READ(12,160,END=20)CARD(1)                                        INIT0040
      GO TO 30                                                          INIT0041
   20 CONTINUE                                                          INIT0042
      IMESH=0                                                           INIT0043
      RETURN                                                            INIT0044
   30 CONTINUE                                                          INIT0045
C                                                                       INIT0046
C  COMMAND = 'SAP5'                                                     INIT0047
C                                                                       INIT0048
      IF(CARD(1).NE.CHAR(1))GO TO 40                                    INIT0049
      IOUT=1                                                            INIT0050
      GO TO 10                                                          INIT0051
C                                                                       INIT0052
C  COMMAND = 'SAP6'                                                     INIT0053
C                                                                       INIT0054
   40 IF(CARD(1).NE.CHAR(2))GO TO 50                                    INIT0055
      IOUT=2                                                            INIT0056
      GO TO 10                                                          INIT0057
C                                                                       INIT0058
C  COMMAND = 'PAFEC'                                                    INIT0059
C                                                                       INIT0060
   50 IF(CARD(1).NE.CHAR(3))GO TO 60                                    INIT0061
      IOUT=3                                                            INIT0062
      GO TO 10                                                          INIT0063
C                                                                       INIT0064
C  COMMAND = 'COORDINATES'                                              INIT0065
C                                                                       INIT0066
   60 IF(CARD(1).NE.CHAR(4))GO TO 70                                    INIT0067
      CALL CTRAN                                                        INIT0068
      GO TO 10                                                          INIT0069
C                                                                       INIT0070
C  COMMAND = 'BEAMS'                                                    INIT0071
C                                                                       INIT0072
   70 IF(CARD(1).NE.CHAR(5))GO TO 80                                    INIT0073
      NPARTS=NPARTS+1                                                   INIT0074
      CALL BEAM                                                         INIT0075
      GO TO 10                                                          INIT0076
```

```
C
C  COMMAND = 'PLOTS'                                                  INIT0077
C                                                                     INIT0078
                                                                      INIT0079
   80 IF(CARD(1).NE.CHAR(6))GO TO 90                                  INIT0080
      IPLOT=1                                                         INIT0081
      NPLOT=1                                                         INIT0082
      GO TO 10                                                        INIT0083
C                                                                     INIT0084
C  COMMAND = 'TOLERANCE'                                              INIT0085
C                                                                     INIT0086
   90 IF(CARD(1).NE.CHAR(7))GO TO 100                                 INIT0087
      BACKSPACE 12                                                    INIT0088
      READ(12,170)TOLER                                               INIT0089
      GO TO 10                                                        INIT0090
C                                                                     INIT0091
C  COMMAND = 'AXISYMMETRIC'                                           INIT0092
C                                                                     INIT0093
  100 IF(CARD(1).NE.CHAR(8))GO TO 110                                 INIT0094
      IAXI=1                                                          INIT0095
      GO TO 10                                                        INIT0096
C                                                                     INIT0097
C  COMMAND = 'LOOP'                                                   INIT0098
C                                                                     INIT0099
  110 IF(CARD(1).NE.CHAR(9))GO TO 120                                 INIT0100
      LOOP=1                                                          INIT0101
      GO TO 10                                                        INIT0102
C                                                                     INIT0103
C  COMMAND = 'MOVIE'                                                  INIT0104
C                                                                     INIT0105
  120 IF(CARD(1).NE.CHAR(10))GO TO 130                                INIT0106
      MOVIE=1                                                         INIT0107
      GO TO 10                                                        INIT0108
C                                                                     INIT0109
C  COMMAND = 'EXCHANGE'                                               INIT0110
C                                                                     INIT0111
  130 IF(CARD(1).NE.CHAR(11))GO TO 140                                INIT0112
      BACKSPACE 12                                                    INIT0113
      READ(12,180)IE1,IE2,IE3                                         INIT0114
      GO TO 10                                                        INIT0115
C                                                                     INIT0116
C  COMMAND = 'FAST'                                                   INIT0117
C                                                                     INIT0118
  140 IF(CARD(1).NE.CHAR(12))GO TO 150                                INIT0119
      NFAST=1                                                         INIT0120
      GO TO 10                                                        INIT0121
  150 CONTINUE                                                        INIT0122
      BACKSPACE 12                                                    INIT0123
C                                                                     INIT0124
C  CHECK FOR COMMENTS                                                 INIT0125
C                                                                     INIT0126
      READ(12,190)CARD(1)                                             INIT0127
      IF(CARD(1).EQ.CMT)GO TO 10                                      INIT0128
      BACKSPACE 12                                                    INIT0129
      IOUT2=IOUT                                                      INIT0130
      RETURN                                                          INIT0131
  160 FORMAT(20A4)                                                    INIT0132
  170 FORMAT(10X,F10.0)                                               INIT0133
  180 FORMAT(12X,3I1)                                                 INIT0134
  190 FORMAT(A1)                                                      INIT0135
      END                                                             INIT0136
      SUBROUTINE INPLOT                                               INPL0001
```

```
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C   SUBROUTINE INPLOT CONTROLS THE MESH PLOTTING
C
C   CALLED BY: MAIN
C
C   CALLS SUBROUTINES:  FN2     INFC    IOP3    IOP4    MAP     PART
C        PLOT     PLOTS    SCALE    SCALE2
C
C   CALLS TCS ROUTINES: ERASE    TSEND    HOME    BELL
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      COMMON/SYSARY/A(36000),B(36000)
      COMMON/JUNK/LCOF,IAXI,MOVIE,IE1,IE2,IE3,NPLOT,TITLE(20),IOUT2
      COMMON/JUNK2/NFAST
      COMMON/KNTS/NNODES,IBEAM,ITWO,ITHRE,LOAD,LINES,MIN
      COMMON/PARTS2/PRTLST(40)
      DIMENSION IBUF(10),IBC(6),CHAR(10),CHAR2(8)
      DATA CHAR/4HCONT,4HEND ,4HSTOP,4HVIEW,4HPLOT,4HROTA,4HSCAL,
     1  4HPART,4HHELP,4HINFO/
      DATA CHAR2/4HAXIS,4HLOCF,4HSAP5,4HSAP6,4HPAFE,4HMOVI,4HFLAT,
     1  4HNUMO/
      DATA PRTLST/40*1.0/
      REWIND 20
      IAXIS=1
      ANGA=0.0
      ANGB=0.0
      CALL PLOTS(5,5,5)
      WRITE(5,280)
      CALL MAP(NNODES,IAXIS,ANGA,ANGB)
C
C   PROCESS INTERACTIVE PLOTTING COMMANDS
C
   10 CONTINUE
      WRITE(5,340)
      READ(5,330)CMD
C
C   COMMAND = 'CONTINUE'
C
      IF(CMD.NE.CHAR(1))GO TO 20
      IF(NFAST.EQ.1)STOP
      RETURN
C
C   COMMAND = 'END'
C
   20 IF(CMD.NE.CHAR(2))GO TO 30
      STOP
C
C   COMMAND = 'STOP'
C
   30 IF(CMD.NE.CHAR(3))GO TO 40
      STOP
C
C   COMMAND = 'VIEW'
C
   40 IF(CMD.NE.CHAR(4))GO TO 50
      CALL IOP4
      CALL ERASE
      CALL TSEND
      GO TO 210
C
```

```
C  COMMAND = 'PLOT'                                                    INPL0063
C                                                                      INPL0064
   50 IF(CMD.NE.CHAR(5))GO TO 60                                       INPL0065
      CALL IOP4                                                        INPL0066
      CALL ERASE                                                       INPL0067
      CALL TSEND                                                       INPL0068
      GO TO 210                                                        INPL0069
C                                                                      INPL0070
C  COMMAND = 'ROTATE'                                                  INPL0071
C                                                                      INPL0072
   60 IF(CMD.NE.CHAR(6))GO TO 70                                       INPL0073
      GO TO 200                                                        INPL0074
C                                                                      INPL0075
C  COMMAND = 'SCALE'                                                   INPL0076
C                                                                      INPL0077
   70 IF(CMD.NE.CHAR(7))GO TO 80                                       INPL0078
      CALL SCALE2                                                      INPL0079
      GO TO 10                                                         INPL0080
C                                                                      INPL0081
C  COMMAND = 'PARTS'                                                   INPL0082
C                                                                      INPL0083
   80 IF(CMD.NE.CHAR(8))GO TO 90                                       INPL0084
      CALL PART                                                        INPL0085
      GO TO 10                                                         INPL0086
C                                                                      INPL0087
C  COMMAND = 'HELP'                                                    INPL0088
C                                                                      INPL0089
   90 IF(CMD.NE.CHAR(9))GO TO 100                                      INPL0090
      WRITE(5,290)                                                     INPL0091
      GO TO 10                                                         INPL0092
C                                                                      INPL0093
C  COMMAND = 'INFORMATION'                                             INPL0094
C                                                                      INPL0095
  100 IF(CMD.NE.CHAR(10))GO TO 110                                     INPL0096
      CALL INFO                                                        INPL0097
      GO TO 10                                                         INPL0098
  110 IF(CMD.NE.CHAR2(1))GO TO 120                                     INPL0099
      IAXI=1                                                           INPL0100
      GO TO 10                                                         INPL0101
  120 IF(CMD.NE.CHAR2(2))GO TO 130                                     INPL0102
      LOOP=1                                                           INPL0103
      GO TO 10                                                         INPL0104
  130 IF(CMD.NE.CHAR2(3))GO TO 140                                     INPL0105
      ICUT2=1                                                          INPL0106
      GO TO 10                                                         INPL0107
  140 IF(CMD.NE.CHAR2(4))GO TO 150                                     INPL0108
      ICUT2=2                                                          INPL0109
      GO TO 10                                                         INPL0110
  150 IF(CMD.NE.CHAR2(5))GO TO 160                                     INPL0111
      ICUT2=3                                                          INPL0112
      GO TO 10                                                         INPL0113
  160 IF(CMD.NE.CHAR2(6))GO TO 170                                     INPL0114
      MOVIE=1                                                          INPL0115
      GO TO 10                                                         INPL0116
  170 IF(CMD.NE.CHAR2(7))GO TO 180                                     INPL0117
      LOOP=0                                                           INPL0118
      GO TO 10                                                         INPL0119
  180 IF(CMD.NE.CHAR2(8))GO TO 190                                     INPL0120
      MOVIE=0                                                          INPL0121
      GO TO 10                                                         INPL0122
  190 CONTINUE                                                         INPL0123
```

```
      GO TO 10                                                          INPL0124
  200 WRITE(5,310)                                                      INPL0125
      READ(5,*)IAXIS,ANGA,ANGB                                          INPL0126
      CALL MAP(NNODES,IAXIS,ANGA,ANGB)                                  INPL0127
      GO TO 10                                                          INPL0128
  210 CONTINUE                                                          INPL0129
C                                                                       INPL0130
C  PLOT THE MESH                                                        INPL0131
C                                                                       INPL0132
      CALL SCALE(IAXIS,ANGA,ANGB)                                       INPL0133
      READ(5,330)CMD                                                    INPL0134
      NLAST=0                                                           INPL0135
      IEMPTY=0                                                          INPL0136
      DO 260 NN=1,LINES                                                 INPL0137
      CALL IOP3(N1,N2,N3,IPART)                                         INPL0138
      IF(PRTLST(IPART).EQ.0.0)GO TO 260                                 INPL0139
      IF(N2.EQ.0)GO TO 230                                              INPL0140
      CALL FN2(N1,N2,N3)                                                INPL0141
      IF(N2.EQ.0)GO TO 230                                              INPL0142
      IF(IEMPTY.EQ.1)CALL PLTDMP(NLAST)                                 INPL0143
      IEMPTY=0                                                          INPL0144
      IPEN=3                                                            INPL0145
      B1=B(N1)                                                          INPL0146
      A1=A(N1)                                                          INPL0147
      IF(N1.NE.NLAST)CALL PLOT(B1,A1,IPEN)                              INPL0148
      NLAST=N3                                                          INPL0149
      IPEN=2                                                            INPL0150
      SINC=0.25                                                         INPL0151
      DO 220 I=1,8                                                      INPL0152
      S=I*SINC-1.0                                                      INPL0153
      CALL FN1(N1,N2,N3,S,XA,YA)                                        INPL0154
      CALL PLOT(YA,XA,IPEN)                                             INPL0155
  220 CONTINUE                                                          INPL0156
      GO TO 260                                                         INPL0157
  230 CONTINUE                                                          INPL0158
      IF(IEMPTY.EQ.1)GO TO 240                                          INPL0159
      CALL PLTADD(N1,N3)                                                INPL0160
      IEMPTY=1                                                          INPL0161
      GO TO 260                                                         INPL0162
  240 CALL PLTCHK(N1,N3,NCHK)                                           INPL0163
      IF(NCHK.EQ.0)GO TO 250                                            INPL0164
      CALL PLTDMP(NLAST)                                                INPL0165
      CALL PLTADD(N1,N3)                                                INPL0166
      GO TO 260                                                         INPL0167
  250 CALL PLTAD2(N1,N3)                                                INPL0168
  260 CONTINUE                                                          INPL0169
      IF(IEMPTY.EQ.1)CALL PLTDMP(NLAST)                                 INPL0170
  270 CALL TSEND                                                        INPL0171
      CALL HOME                                                         INPL0172
      CALL BELL                                                         INPL0173
      CALL TSEND                                                        INPL0174
      REWIND 20                                                         INPL0175
      GO TO 10                                                          INPL0176
  280 FORMAT(27H < INGRID PLOTTING PACKAGE >)                            INPL017
  290 FORMAT(38H < INGRID HAS THE FOLLOWING COMMANDS: >,/,               INPL017
     160H < CONTINUE,END,HELP,INFO,PART,PLOT,ROTATE,SCALE,STOP,VIEW >)  INPL0179
  300 FORMAT(35X,3E10.3)                                                INPL018C
  310 FORMAT(45H INPUT VERTICAL AXIS, ANGLE A, AND ANGLE B              INPL0181
     1 >>,$)                                                            INPL0182
  320 FORMAT(3I5)                                                       INPL0183
  330 FORMAT(A4)                                                        INPL0184
```

```
  340 FORMAT(3H >>,$)                                                    INPL0165
      END                                                                INPL0166
      SUBROUTINE PLTADD(N1,N2)                                           PLTD0001
      COMMON/PLTBUF/NA,NB                                                PLTD0002
      NA=N1                                                              PLTD0003
      NB=N2                                                              PLTD0004
      RETURN                                                             PLTD0005
      END                                                                PLTD0006
      SUBROUTINE PLTAD2(N,N2)                                            PLTD0007
      COMMON/PLTBUF/NA,NB                                                PLTD0008
      NB=N2                                                              PLTD0009
      RETURN                                                             PLTD0003
      END                                                                PLTD0004
      SUBROUTINE PLTCHK(N1,N2,NCHK)                                      PLTD0005
      COMMON/PLTBUF/NA,NB                                                PLTC0001
      NCHK=1                                                             PLTC0002
      IF(N1.NE.NB)RETURN                                                 PLTC0003
      NT1=NA                                                             PLTC0004
      NT2=NB                                                             PLTC0005
      NT3=N2                                                             PLTC0006
      CALL FN2(NT1,NT2,NT3)                                              PLTC0007
      NCHK=NT2                                                           PLTC0008
      RETURN                                                             PLTC0009
      END                                                                PLTC0010
      SUBROUTINE PLTDMP(NLAST)                                           PLTC0011
      COMMON/PLTBUF/NA,NB                                                PLTD0001
      COMMON/SYSARY/A(36000),B(36000)                                    PLTD0002
      IF(NLAST.EQ.NA)GO TO 10                                            PLTC0003
      BA=B(NA)                                                           PLTC0004
      AA=A(NA)                                                           PLTC0005
      IPEN=3                                                             PLTD0006
      CALL PLOT(BA,AA,IPEN)                                              PLTD0007
   10 AB=A(NB)                                                           PLTD0008
      BB=B(NB)                                                           PLTD0009
      IPEN=2                                                             PLTC0010
      CALL PLOT(BB,AB,IPEN)                                              PLTD0011
      RETURN                                                             PLTC0012
      END                                                                PLTD0013
      SUBROUTINE ICN1(X,Y,Z,IBC)                                         PLTD0014
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCION10001
C                                                                        ICN10002
C  SUBROUTINE ICN1 PUTS A NODE INTO THE OUTPUT BUFFER                    CION10003
C                                                                        CION10004
C  CALLED BY:  BEAM    ELTHRE  ELTWO                                     CION10005
C                                                                        CION10006
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCION10008
      COMMON/DUMPAR/ XX(100),YY(100),ZZ(100),IDUM,IB(100)                ION10009
      DATA I/0/                                                          ION10010
      I=I+1                                                              ION10011
      XX(I)=X                                                            ION10012
      YY(I)=Y                                                            ION10013
      ZZ(I)=Z                                                            ION10014
      IB(I)=IBC                                                          ION10015
      IF(I.LT.100)RETURN                                                 ION10016
C                                                                        ION10017
C  BUFFER IS FULL,  DUMP AND RETURN                                      ION10018
C                                                                        ION10019
      WRITE(7)XX,YY,ZZ,IB                                                ION10020
      I=0                                                                ION10021
      RETURN                                                             ION10022
      END                                                                ION10023
```

```
      SUBROUTINE ICN2                                                    ION20001
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCION20002
C                                                                        CION20003
C  SUBROUTINE ION2  DUMPS THE NODAL OUTPUT BUFFER                        CION20004
C                                                                        CION20005
C  CALLED BY:  MAIN                                                      CION20006
C                                                                        CION20007
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCION20008
      COMMON/DUMPAR/XX(100),YY(100),ZZ(100),IDUM,IB(100)                 ION20009
      WRITE(7)XX,YY,ZZ,IB                                                ION20010
      RETURN                                                             ION20011
      END                                                                ION20012
      SUBROUTINE ICN3(X,Y,Z,IBC)                                         ION30001
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCION30002
C                                                                        CION30003
C  SUBROUTINE ICN3 GETS A NODE FROM THE INPUT BUFFER                     CION30004
C                                                                        CION30005
C  CALLED BY:  INPLOT  OUTPUT                                            CION30006
C                                                                        CION30007
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCION30008
      COMMON/DUMPAR/ XX(100),YY(100),ZZ(100),I,IB(100)                   ION30009
      DATA I/0/                                                          ION30010
      I=I+1                                                              ION30011
      X=XX(I)                                                            ION30012
      Y=YY(I)                                                            ION30013
      Z=ZZ(I)                                                            ION30014
      IBC=IB(I)                                                          ION30015
      IF(I.LT.100)RETURN                                                 ION30016
C                                                                        ICN30017
C  BUFFER IS EMPTY,  REFRESH BUFFER                                      ION30018
C                                                                        ION30019
      IEND=0                                                             ION30020
      READ(7)XX,YY,ZZ,IB                                                 ION30021
      I=0                                                                ION30022
      RETURN                                                             ION30023
      END                                                                ION30024
      SUBROUTINE ICN4                                                    ION40001
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCION40002
C                                                                        CION40003
C  SUBROUTINE ICN4 INITIALIZES THE NODAL INPUT BUFFER                    CION40004
C                                                                        CION40005
C  CALLED BY:  INPLOT  OUTPUT                                            CION40006
C                                                                        CION40007
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCION40008
      COMMON/DUMPAR/XX(100),YY(100),ZZ(100),I,IB(100)                    ION40009
      I=0                                                                ION40010
      REWIND 7                                                           ION40011
      READ(7)XX,YY,ZZ,IB                                                 ION40012
      RETURN                                                             ION40013
      END                                                                ION40014
      SUBROUTINE ICP1(I,J,K)                                             IOP10001
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCIOP10002
C                                                                        CIOP10003
C  SUBROUTINE ICP1 PUTS A LINE INTO THE LINE BUFFER                      CIOP10004
C                                                                        CIOP10005
C  CALLED BY:  BEAM    PLOTMK                                            CIOP10006
C                                                                        CIOP10007
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCIOP10008
      COMMON/DUMPA2/ II(100),JJ(100),KK(100),NDUM,IP(100)                IOP10009
      COMMON/KNTS/NODE,IBEAM,ITHC,ITHRE,LOAD,LINE,MIN                    IOP10010
      COMMON/PARTS/NPARTS,XMAXP(40,3),XMINP(40,3)                        IOP10011
```

```
      DATA N/0/,LINE/0/                                                 IOP10012
        IF(I.LT.MIN.AND.J.LT.MIN.AND.K.LT.MIN)RETURN                    IOP10013
      N=N+1                                                             IOP10014
      LINE=LINE+1                                                       IOP10015
      II(N)=I                                                           IOP10016
      JJ(N)=J                                                           IOP10017
      KK(N)=K                                                           IOP10018
      IP(N)=NPARTS                                                      IOP10019
      IF(N.LT.100)RETURN                                                IOP10020
C                                                                       IOP10021
C  BUFFER IS FULL,  DUMP BUFFER                                         IOP10022
C                                                                       IOP10023
      WRITE(20)II,JJ,KK,IP                                              IOP10024
      N=0                                                               IOP10025
      RETURN                                                            IOP10026
      END                                                               IOP10027
      SUBROUTINE IOP2                                                   IOP20001
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCIOP20002
C                                                                      CIOP20003
C  SUBROUTINE IOP2 DUMPS THE LINES OUTPUT BUFFER                       CIOP20004
C                                                                      CIOP20005
C  CALLED BY:  MAIN                                                    CIOP20006
C                                                                      CIOP20007
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCIOP20008
      COMMON/DUMPA2/II(100),JJ(100),KK(100),N,IP(100)                   IOP20009
      WRITE(20)II,JJ,KK,IP                                              IOP20010
      RETURN                                                            IOP20011
      END                                                               IOP20012
      SUBROUTINE IOP3(I,J,K,IPART)                                      IOP30001
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCIOP30002
C                                                                      CIOP30003
C  SUBROUTINE IOP3 GETS A LINE FROM THE INPUT BUFFER                   CIOP30004
C                                                                      CIOP30005
C  CALLED BY:  INPLOT                                                  CIOP30006
C                                                                      CIOP30007
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCIOP30008
      COMMON/DUMPA2/ II(100),JJ(100),KK(100),N,IP(100)                  IOP30009
      DATA N/0/                                                         IOP30010
      N=N+1                                                             IOP30011
      I=II(N)                                                           IOP30012
      J=JJ(N)                                                           IOP30013
      K=KK(N)                                                           IOP30014
      IPART=IP(N)                                                       IOP30015
      IF(N.LT.100)RETURN                                                IOP30016
C                                                                       IOP30017
C  BUFFER IS EMPTY,  REFRESH BUFFER                                     IOP30018
C                                                                       IOP30019
      READ(20)II,JJ,KK,IP                                               IOP30020
      N=0                                                               IOP30021
      RETURN                                                            IOP30022
      END                                                               IOP30023
      SUBROUTINE IOP4                                                   IOP40001
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCIOP40002
C                                                                      CIOP40003
C  SUBROUTINE IOP4  INITIALIZES THE LINES INPUT BUFFER                 CIOP40004
C                                                                      CIOP40005
C  CALLED BY:  INPLOT                                                  CIOP40006
C                                                                      CIOP40007
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCIOP40008
      COMMON/DUMPA2/II(100),JJ(100),KK(100),N,IP(100)                   IOP40009
      N=0                                                               IOP40010
```

```
      REWIND 2C                                                          IOP40011
      READ(20)II,JJ,KK,IP                                                IOP40012
      RETURN                                                             IOP40013
      END                                                                IOP40014
      SUBROUTINE ITST(M,XC,YC,ZC)                                        ITST0001
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCITST0002
C                                                                        CITST0003
C  SUBROUTINE ITST TEST FOR 2 PLATES WITH THE SAME NORMAL VECTOR.  IF    CITST0004
C  THEY HAVE DIFFERENT NORMALS THEN THE ROTATIONAL DEGREES OF FREEDOM    CITST0005
C  ARE RELEASED.                                                         CITST0006
C                                                                        CITST0007
C  CALLED BY:  NORM                                                      CITST0008
C                                                                        CITST0009
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCITST0010
      DIMENSION M(8)                                                     ITST0011
      COMMON/SYSARY/X(8000),Y(8000),Z(8000),N(8000)                      ITST0012
     1,IBB(8000),IFLAG(8000),V(8000,3)                                   ITST0013
      DO 30 MI=1,8                                                       ITST0014
      IF(M(MI).EQ.0)GO TO 30                                             ITST0015
      IF(IFLAG(M(MI)).EQ.0)GO TO 10                                      ITST0016
      IF(IFLAG(M(MI)).EQ.1)GO TO 20                                      ITST0017
      GO TO 30                                                           ITST0018
   10 CONTINUE                                                           ITST0019
      IFLAG(M(MI))=1                                                     ITST0020
      V(M(MI),1)=XC                                                      ITST0021
      V(M(MI),2)=YC                                                      ITST0022
      V(M(MI),3)=ZC                                                      ITST0023
      GO TO 30                                                           ITST0024
   20 CONTINUE                                                           ITST0025
      XDIFF=ABS(XC-V(M(MI),1))                                           ITST0026
      YDIFF=ABS(YC-V(M(MI),2))                                           ITST0027
      ZDIFF=ABS(ZC-V(M(MI),3))                                           ITST0028
      IF(XDIFF.LT..2.AND.YDIFF.LT..2.AND.ZDIFF.LT..2)GO TO 30            ITST0029
      IFLAG(M(MI))=2                                                     ITST0030
   30 CONTINUE                                                           ITST0031
      RETURN                                                             ITST0032
      END                                                                ITST0033
      SUBROUTINE LOADS(IB,LC,FX,FY,FZ)                                   LADS0001
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCLADS0002
C                                                                        CLADS0003
C  SUBROUTINE LOADS STORES THE LOAD COMMANDS                             CLADS0004
C                                                                        CLADS0005
C  CALLED BY:  SEQ2                                                      CLADS0006
C                                                                        CLADS0007
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCLADS0008
      DIMENSION IB(6)                                                    LADS0009
      COMMON/LCAD/NL,LCASE(100),XF(100),YF(100),ZF(100),IND(100,6)       LADS0010
      NL=NL+1                                                            LADS0011
      XF(NL)=FX                                                          LADS0012
      YF(NL)=FY                                                          LADS0013
      ZF(NL)=FZ                                                          LADS0014
      LCASE(NL)=LC                                                       LADS0015
      DO 10 I=1,6                                                        LADS0016
      IND(NL,I)=IB(I)                                                    LADS0017
   10 CONTINUE                                                           LADS0018
      RETURN                                                             LADS0019
      END                                                                LADS0020
      SUBROUTINE LOADS2(I,J,K,N)                                         LDS20001
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCLDS20002
C                                                                        CLDS20003
C  SUBROUTINE LOADS2  APPLIES THE LOADS TO NODE POINTS                   CLDS20004
```

```
C                                                                               CLDS20005
C  CALLED BY:  ELTHRE  ELTWO                                                    CLDS20006
C                                                                               CLDS20007
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCLDS20008
      COMMON/LCAD/NL,LCASE(100),XF(100),YF(100),ZF(100),IND(100,6)               LDS20009
      COMMON/KNTS/NODE,IBEAM,ITWO,ITHRE,LCAD,LINES,MIN                          LDS20010
      IF(NL.EQ.0)RETURN                                                         LDS20011
      DO 10 L=1,NL                                                              LDS20012
      IF(I.LT.IND(L,1).OR.I.GT.IND(L,4))GO TO 10                                LDS20013
      IF(J.LT.IND(L,2).OR.J.GT.IND(L,5))GO TO 10                                LDS20014
      IF(K.LT.IND(L,3).OR.K.GT.IND(L,6))GO TO 10                                LDS20015
      FX=XF(L)                                                                  LDS20016
      FY=YF(L)                                                                  LDS20017
      FZ=ZF(L)                                                                  LDS20018
      WRITE(13,20)N,LCASE(L),FX,FY,FZ                                           LDS20019
      LCAD=LCAD+1                                                               LDS20020
   10 CONTINUE                                                                  LDS20021
      RETURN                                                                    LDS20022
   20 FORMAT(2I5,3F10.4)                                                        LDS20023
      END                                                                       LDS20024
      SUBROUTINE LCADS3                                                         LDS30001
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCLDS30002
C                                                                               LDS30003
C  SUBROUTINE LOADS3  PUTS LOAD CARDS IN THE PROPER FORMAT AND PRINTS           LDS30004
C  THEM.                                                                        LDS30005
C                                                                               LDS30006
C  CALLED BY:  OUTPUT                                                           LDS30007
C                                                                               LDS30008
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCLDS30009
      COMMON/SYSARY/X(8000),Y(8000),Z(8000),N(8000)                             LDS30010
     1,IBB(8000),IFLAG(8000),V(8000,3)                                          LDS30011
      REWIND 13                                                                 LDS30012
      DO 40 I=1,23                                                              LDS30013
   10 READ(13,50,END=30)NODE,LC,FX,FY,FZ                                        LDS30014
      IF(LC.NE.I)GO TO 20                                                       LDS30015
      NODE=N(NODE)                                                              LDS30016
      WRITE(14,60)NODE,LC,FX,FY,FZ                                              LDS30017
   20 CONTINUE                                                                  LDS30018
      GO TO 10                                                                  LDS30019
   30 CONTINUE                                                                  LDS30020
      REWIND 13                                                                 LDS30021
   40 CONTINUE                                                                  LDS30022
      RETURN                                                                    LDS30023
   50 FORMAT(2I5,3F10.0)                                                        LDS30024
   60 FORMAT(2I5,3F10.0)                                                        LDS30025
      END                                                                       LDS30026
      SUBROUTINE MAP(NNODES,IA,IS,ANGA,ANGB)                                    MAP0001
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCMAP0002
C                                                                               CMAP0003
C  SUBROUTINE MAP PROJECTS A NODE ONTO A 2-D SURFACE FOR PLOTTING               CMAP0004
C                                                                               CMAP0005
C  CALLED BY:  INPLOT                                                           CMAP0006
C                                                                               CMAP0007
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCMAP0008
      COMMON/SYSARY/A(36000),B(36000)                                           MAP0009
      COMMON/SCALES/FACTOR,ATRAN,BTRAN,F2                                       MAP0010
C                                                                               MAP0011
C  INITIALIZE THE NODAL INPUT BUFFER                                            MAP0012
C                                                                               MAP0013
      CALL ICN4                                                                 MAP0014
      SA=SIN(ANGA/57.2958)                                                      MAP0015
```

```
      SB=SIN(ANGB/57.2958)                                                    MAP0016
      CA=COS(ANGA/57.2958)                                                    MAPC017
      CB=COS(ANGB/57.2958)                                                    MAP0018
      CBSA=CB*SA                                                              MAPC019
      SBSA=SB*SA                                                              MAPC020
C                                                                             MAP0021
C  INPUT THE NODES                                                            MAP0022
C                                                                             MAP0023
      DO 50 I=1,NNODES                                                        MAPC024
      CALL ION3(X,Y,Z,IBC)                                                    MAP0025
C                                                                             MAPC026
C  CHECK VERTICAL AXIS                                                        MAPC027
C                                                                             MAP0028
      IF(IAXIS.EQ.1)GO TO 10                                                  MAP0029
      IF(IAXIS.EQ.2)GO TO 20                                                  MAP003C
      IF(IAXIS.EQ.3)GO TO 30                                                  MAP0031
   10 B(I)=-Z*SB+X*CA                                                         MAP0032
      A(I)=-Z*CBSA-X*SBSA+Y*CA                                                MAP0033
      GO TO 40                                                                MAPC034
   20 B(I)=-X*SB+Y*CA                                                         MAP0035
      A(I)=-X*CBSA-Y*SBSA+Z*CA                                                MAPC036
      GO TO 40                                                                MAP0037
   30 B(I)=-Y*SB+Z*CA                                                         MAP0038
      A(I)=-Y*CBSA-Z*SBSA+X*CA                                                MAPC039
   40 CONTINUE                                                                MAP0040
   50 CONTINUE                                                                MAPC041
      RETURN                                                                  MAP0042
      END                                                                     MAP0043
      SUBROUTINE MAX(IM,JM,KM)                                                MAX0001
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCMAXC002
C                                                                             CMAXC003
C  SUBROUTINE MAX DETERMINES THE MAXIMUM INDICES USED FOR A PART              CMAXC004
C                                                                             CMAX0005
C  CALLED BY:  DLEE                                                           CMAXC006
C                                                                             CMAX0007
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCMAXCCC8
      DIMENSION IO(16),JO(16),KO(16)                                          MAXC009
      COMMON/PROP/MAT,NDOT,IC2,IT,TH,N8,N20                                   MAXC01C
      READ(11,20)S,IO,JO,KO                                                   MAXC011
      IM=0                                                                    MAX0012
      JM=0                                                                    MAX0013
      KM=0                                                                    MAX0014
      CALL DOUBLE(IO,JO,KO)                                                   MAXC015
      DO 10 I=1,16                                                            MAXC016
      IM=AMAXO(IM,IABS(IO(I)))                                                MAXC017
      JM=AMAXO(JM,IABS(JO(I)))                                                MAXC018
      KM=AMAXO(KM,IABS(KO(I)))                                                MAXC019
   10 CONTINUE                                                                MAXCC2C
      RETURN                                                                  MAX0021
   20 FORMAT(A1,/,16I5,/,16I5,/,16I5)                                         MAXC022
      END                                                                     MAX0023
      SUBROUTINE MAXMIN(X1,X2,I3)                                             MXMN0001
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCMXMN0002
C                                                                             CMXMN0003
C  SUBROUTINE MAXMIN DETERMINES THE MAXIMUM AND MINIMUM COORDINATES           CMXMNCC4
C  FOR EACH PART OF THE STRUCTURE.                                            CMXMN0005
C                                                                             CMXMNCCC6
C  CALLED BY:  BEAM    ELTHRE  ELTWO                                          CMXMN0007
C                                                                             CMXMN0008
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCMXMNCCC9
      DIMENSION X(3)                                                          MXMN0010
```

```
      COMMON/PARTS/NPARTS,XMAXP(40,3),XMINP(40,3)                       MXMNC011
      DATA XMAXP/120*-999.0/,XMINP/120*999.0/                           MXMN0012
      X(1)=X1                                                           MXMN0013
      X(2)=X2                                                           MXMN0014
      X(3)=X3                                                           MXMN0015
      DO 10 I=1,3                                                       MXMNC016
      XMAXP(NPARTS,I)=AMAX1(XMAXP(NPARTS,I),X(I))                       MXMN0017
      IF(X(I).LT.XMINP(NPARTS,I))XMINP(NPARTS,I)=X(I)                   MXMNC018
   10 CONTINUE                                                          MXMN0019
      RETURN                                                            MXMN0020
      END                                                               MXMN0021
      SUBROUTINE MESH(M1,M2,M3,X,Y,Z,N,IM,JM,KM,IX,JX,KX)               MESH0001
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCMESH0002
C                                                                       CMESH0003
C  SUBROUTINE MESH USES LINEAR INTERPOLATION TO DETERMINE THE LOCATION  CMESH0004
C  OF A REGION.  THE REGIONS 8 VERTICES MUST HAVE ALREADY BEEN INPUT    CMESH0005
C  BY SUBROUTINE SEQ2.                                                  CMESH0006
C                                                                       CMESH0007
C  CALLED BY:  ELTHRE  ELTWO                                            CMESH0008
C                                                                       CMESH0009
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCMESH0010
      DIMENSION X(M1,M2,M3),Y(M1,M2,M3),Z(M1,M2,M3),N(M1,M2,M3)         MESH0011
      X1=X(IM,JM,KM)                                                    MESH0012
      X2=X(IX,JM,KM)                                                    MESH0013
      X3=X(IM,JX,KM)                                                    MESH0014
      X4=X(IM,JM,KX)                                                    MESH0015
      X5=X(IX,JX,KM)                                                    MESH0016
      X6=X(IX,JM,KX)                                                    MESH0017
      X7=X(IM,JX,KX)                                                    MESH0018
      X8=X(IX,JX,KX)                                                    MESH0019
      Y1=Y(IM,JM,KM)                                                    MESH0020
      Y2=Y(IX,JM,KM)                                                    MESH0021
      Y3=Y(IM,JX,KM)                                                    MESH0022
      Y4=Y(IM,JM,KX)                                                    MESH0023
      Y5=Y(IX,JX,KM)                                                    MESH0024
      Y6=Y(IX,JM,KX)                                                    MESH0025
      Y7=Y(IM,JX,KX)                                                    MESH0026
      Y8=Y(IX,JX,KX)                                                    MESH0027
      Z1=Z(IM,JM,KM)                                                    MESH0028
      Z2=Z(IX,JM,KM)                                                    MESH0029
      Z3=Z(IM,JX,KM)                                                    MESH0030
      Z4=Z(IM,JM,KX)                                                    MESH0031
      Z5=Z(IX,JX,KM)                                                    MESH0032
      Z6=Z(IX,JM,KX)                                                    MESH0033
      Z7=Z(IM,JX,KX)                                                    MESH0034
      Z8=Z(IX,JX,KX)                                                    MESH0035
      IXIM=IX-IM                                                        MESH0036
      JXJM=JX-JM                                                        MESH0037
      KXKM=KX-KM                                                        MESH0038
      DO 60 I=IM,IX                                                     MESH0039
      WI=1.0                                                            MESH0040
      IF((IXIM).NE.0)WI=FLOAT(IX-I)/(IXIM)                              MESH0041
      DO 50 J=JM,JX                                                     MESH0042
      WJ=1.0                                                            MESH0043
      IF((JXJM).NE.0)WJ=FLOAT(JX-J)/(JXJM)                              MESH0044
      FA=WI*WJ                                                          MESH0045
      DO 40 K=KM,KX                                                     MESH0046
      IF(N(I,J,K).NE.0)GO TO 40                                         MESH0047
      IF((I.EQ.IM.OR.I.EQ.IX).AND.(J.EQ.JM.OR.J.EQ.JX)                  MESH0048
     1.AND.(K.EQ.KM.OR.K.EQ.KX))GO TO 40                                MESH0049
      WK=1.0                                                            MESH0050
```

```
      IF((KXKM).NE.0)WK=FLOAT(KX-K)/(KXKM)                              MESH0051
      FB=WJ*WK                                                          MESH0052
      FC=WI*WK                                                          MESH0053
      F1=FA*WK                                                          MESH0054
      F2=FB-F1                                                          MESH0055
      F3=FC-F1                                                          MESH0056
      F4=FA-F1                                                          MESH0057
      F5=F1+WK-FB-FC                                                    MESH0058
      F6=F1+WJ-FA-FB                                                    MESH0059
      F7=F1+WI-FA-FC                                                    MESH0060
      F8=FA-WI-WJ+1-F5                                                  MESH0061
      IF((IXIM).EQ.0)GO TO 10                                           MESH0062
      IF((JXJM).EQ.0)GO TO 20                                           MESH0063
      IF((KXKM).EQ.0)GO TO 30                                           MESH0064
      X(I,J,K)=X1*F1+X2*F2+X3*F3+X4*F4+X5*F5+X6*F6+X7*F7+X8*F8          MESH0065
      Y(I,J,K)=Y1*F1+Y2*F2+Y3*F3+Y4*F4+Y5*F5+Y6*F6+Y7*F7+Y8*F8          MESH0066
      Z(I,J,K)=Z1*F1+Z2*F2+Z3*F3+Z4*F4+Z5*F5+Z6*F6+Z7*F7+Z8*F8          MESH0067
      GO TO 40                                                          MESH0068
   10 CONTINUE                                                          MESH0069
      X(I,J,K)=X1*F1+X4*F4+X3*F3+X7*F7                                  MESH0070
      Y(I,J,K)=Y1*F1+Y4*F4+Y3*F3+Y7*F7                                  MESH0071
      Z(I,J,K)=Z1*F1+Z4*F4+Z3*F3+Z7*F7                                  MESH0072
      GO TO 40                                                          MESH0073
   20 CONTINUE                                                          MESH0074
      X(I,J,K)=X1*F1+X2*F2+X6*F6+X4*F4                                  MESH0075
      Y(I,J,K)=Y1*F1+Y2*F2+Y6*F6+Y4*F4                                  MESH0076
      Z(I,J,K)=Z1*F1+Z2*F2+Z6*F6+Z4*F4                                  MESH0077
      GO TO 40                                                          MESH0078
   30 CONTINUE                                                          MESH0079
      X(I,J,K)=X1*F1+X2*F2+X5*F5+X3*F3                                  MESH0080
      Y(I,J,K)=Y1*F1+Y2*F2+Y5*F5+Y3*F3                                  MESH0081
      Z(I,J,K)=Z1*F1+Z2*F2+Z5*F5+Z3*F3                                  MESH0082
   40 CONTINUE                                                          MESH0083
   50 CONTINUE                                                          MESH0084
   60 CONTINUE                                                          MESH0085
      RETURN                                                            MESH0086
      END                                                               MESH0087
      SUBROUTINE MIDSID(IB,IX,IY,IZ)                                    MDSD0001
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCMDSD0002
C                                                                       CMDSD0003
C  SUBROUTINE MIDSID STORES COMMANDS FOR ELIMINATION OF MIDSIDE NODES   CMDSD0004
C                                                                       CMDSD0005
C  CALLED BY:  SEQ2                                                     CMDSD0006
C                                                                       CMDSD0007
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCMDSD0008
      DIMENSION IB(6)                                                   MDSD0009
      COMMON/MID/NMID,MIDA(100,6),MIDC(100,3)                           MDSD0010
      NMID=NMID+1                                                       MDSD0011
      DO 10 I=1,6                                                       MDSD0012
      MIDA(NMID,I)=IB(I)                                                MDSD0013
   10 CONTINUE                                                          MDSD0014
      MIDC(NMID,1)=IX                                                   MDSD0015
      MIDC(NMID,2)=IY                                                   MDSD0016
      MIDC(NMID,3)=IZ                                                   MDSD0017
      RETURN                                                            MDSD0018
      END                                                               MDSD0019
      SUBROUTINE MIDSI2(I,J,K,MFLAG,XO,YO,ZO)                           MDS20001
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCMDS20002
C                                                                       CMDS20003
C  SUBROUTINE MIDSI2 DETERMINES IF A MIDSIDE NODE IS TO BE DELETED      CMDS2C04
C                                                                       CMDS20005
```

```
C  CALLED BY:  ELTHRE  ELTWO                                                    CMDS20006
C                                                                              CMDS20007
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCMDS20008
      COMMON/MID/NMID,MIDA(100,6),MIDC(100,3)                                  MDS20009
      IF(NMID.EQ.0)RETURN                                                      MDS20010
      DO 10 L=1,NMID                                                           MDS20011
      IF(I.LT.MIDA(L,1).OR.I.GT.MIDA(L,4))GO TO 10                             MDS20012
      IF(J.LT.MIDA(L,2).OR.J.GT.MIDA(L,5))GO TO 10                             MDS20013
      IF(K.LT.MIDA(L,3).OR.K.GT.MIDA(L,6))GO TO 10                             MDS20014
      IF(MIDC(L,1).EQ.1.AND.XO.NE.C.0)MFLAG=1                                  MDS20015
      IF(MIDC(L,2).EQ.1.AND.YO.NE.C.0)MFLAG=1                                  MDS20016
      IF(MIDC(L,3).EQ.1.AND.ZO.NE.C.0)MFLAG=1                                  MDS20017
   10 CONTINUE                                                                 MDS20018
      RETURN                                                                   MDS20019
      END                                                                      MDS20020
      SUBROUTINE MOVE(M1,M2,M3,X,Y,Z,NUM,I1,J1,K1,I2,J2,K2)                    MOVE0001
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCMOVE0002
C                                                                              CMOVE0003
C  SUBROUTINE MOVE ADDS A VECTOR TO NODE POINTS ACCORDING TO THE ARC           CMOVE0004
C  DATA.                                                                       CMOVE0005
C                                                                              CMOVE0006
C  CALLED BY:  ELTHRE  ELTWO                                                   CMOVE0007
C                                                                              CMOVE0008
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCMOVE0009
      DIMENSION X(M1,M2,M3),Y(M1,M2,M3),Z(M1,M2,M3),IB(6)                      MOVE0010
     1  ,NUM(M1,M2,M3)                                                         MOVE0011
      COMMON/VECFNC/NFN,IFN(10C),IAPLY(100,6),COEF(100,5)                      MOVE0012
      IF(NFN.EC.C)RETURN                                                       MOVE0013
      DO 80 I=1,NFN                                                            MOVE0014
      DO 10 K=1,6                                                              MOVE0015
      IB(K)=IAPLY(I,K)                                                         MOVE0016
   10 CONTINUE                                                                 MOVE0017
      IF(IB(1).NE.I1.AND.IB(1).NE.I2)GO TO 80                                  MOVE0018
      IF(IB(4).NE.I1.AND.IB(4).NE.I2)GO TO 80                                  MOVE0019
      IF(IB(2).NE.J1.AND.IB(2).NE.J2)GO TO 80                                  MOVE0020
      IF(IB(5).NE.J1.AND.IB(5).NE.J2)GO TO 80                                  MOVE0021
      IF(IB(3).NE.K1.AND.IB(3).NE.K2)GO TO 80                                  MOVE0022
      IF(IB(6).NE.K1.AND.IB(6).NE.K2)GO TO 80                                  MOVE0023
      IF(IB(4).NE.IB(1))J=1                                                    MOVE0024
      IF(IB(5).NE.IB(2))J=2                                                    MOVE0025
      IF(IB(6).NE.IB(3))J=3                                                    MOVE0026
      I2I1=I2-I1                                                               MOVE0027
      J2J1=J2-J1                                                               MOVE0028
      K2K1=K2-K1                                                               MOVE0029
      DO 70 L=I1,I2                                                            MOVE0030
      DO 60 M=J1,J2                                                            MOVE0031
      DO 50 N=K1,K2                                                            MOVE0032
      IF(NUM(L,M,N).NE.0)GO TO 50                                              MOVE0033
C                                                                              MOVE0034
C  DETERMINE SCALE FACTORS                                                     MOVE0035
C                                                                              MOVE0036
      IF(J.EQ.1)T=FLOAT(L-I1)/(I2I1)                                           MOVE0037
      IF(J.EQ.2)T=FLOAT(M-J1)/(J2J1)                                           MOVE0038
      IF(J.EQ.3)T=FLOAT(N-K1)/(K2K1)                                           MOVE0039
      IF(T.EQ.0.0.OR.T.EQ.1.0)GO TO 50                                         MOVE0040
      WI=1.0                                                                   MOVE0041
      WJ=1.0                                                                   MOVE0042
      WK=1.0                                                                   MOVE0043
      IF((I2I1).NE.0)WI=1.0-ABS(FLOAT(L-IB(1))/(I2I1))                         MOVE0044
      IF((J2J1).NE.0)WJ=1.0-ABS(FLOAT(M-IB(2))/(J2J1))                         MOVE0045
      IF((K2K1).NE.0)WK=1.0-ABS(FLOAT(N-IB(3))/(K2K1))                         MOVE0046
```

```
      IF(J.EQ.1)WI=1.0                                                  MOVEC047
      IF(J.EQ.2)WJ=1.0                                                  MOVE0048
      IF(J.EQ.3)WK=1.0                                                  MOVEC049
      IF(WI.EQ.0.0)GO TO 70                                             MOVE005C
      IF(WJ.EQ.0.0)GO TO 60                                             MOVEC051
      IF(WK.EQ.0.0)GO TO 50                                             MOVEC052
      IF(T.EQ.TD)GO TO 40                                               MOVE0053
      IF(IFN(I).EQ.1)GO TO 20                                           MOVEC054
      IF(IFN(I).EQ.2)GO TO 30                                           MOVE0055
   20 CONTINUE                                                          MOVE0056
C                                                                       MOVEC057
C  VECTORS DEFINE A PARABOLA                                            MOVE0058
C                                                                       MOVE0059
      T2=T-T*T                                                          MOVE0060
      XV=COEF(I,1)*T2                                                   MOVE0061
      YV=COEF(I,2)*T2                                                   MOVE0062
      ZV=COEF(I,3)*T2                                                   MOVE0063
      GO TO 40                                                          MOVE0064
   30 CONTINUE                                                          MOVE0065
C                                                                       MOVE0066
C  VECTORS DEFINE A CIRCULAR ARC                                        MOVE0067
C                                                                       MOVE0068
      R=COEF(I,4)                                                       MOVE0069
      R2=R*R                                                            MOVE007C
      XM=COEF(I,5)/2.0                                                  MOVE0071
      XS=COEF(I,5)*(T-0.5)                                              MOVE0072
      YS=SQRT(-XS*XS+R2)-SQRT(-XM*XM+R2)                                MOVE0073
      XV=COEF(I,1)*YS                                                   MOVE0074
      YV=COEF(I,2)*YS                                                   MOVE0075
      ZV=COEF(I,3)*YS                                                   MOVE0076
      GO TO 40                                                          MOVE0077
   40 CONTINUE                                                          MOVE0078
C                                                                       MOVE0079
C  ADD VECTORS TO NODES                                                 MOVE0080
C                                                                       MOVE0081
      F=WI*WJ*WK                                                        MOVE0082
      XP=XV*F                                                           MOVE0083
      YP=YV*F                                                           MOVE0084
      ZP=ZV*F                                                           MOVE0085
      X(L,M,N)=X(L,M,N)+XP                                              MOVE0086
      Y(L,M,N)=Y(L,M,N)+YP                                              MOVE0087
      Z(L,M,N)=Z(L,M,N)+ZP                                              MOVE0088
      TD=T                                                              MOVE0089
   50 CONTINUE                                                          MOVE0090
   60 CONTINUE                                                          MOVE0091
   70 CONTINUE                                                          MOVE0092
      TD=0.0                                                            MOVE0093
   80 CONTINUE                                                          MOVE0094
      RETURN                                                            MOVE0095
      END                                                               MOVE0096
      SUBROUTINE NORM(M)                                                NORM0001
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCNORM0002
C                                                                       CNORM0003
C  SUBROUTINE NORM DETERMINES THE NORMAL VECTOR TO A SURFACE            CNORM0004
C                                                                       CNORM0005
C  CALLED BY:  EL2D                                                     CNORM0006
C                                                                       CNORM0007
C  CALLS SUBROUTINES:  ITST                                             CNORM0008
C                                                                       CNORM0009
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCNORM0010
      COMMON/SYSARY/X(8000),Y(8000),Z(8000),N(8000)                     NORM0011
```

```
     1,IBB(8000),IFLAG(8000),V(8000,3)                                  NORM0012
      DIMENSION M(8)                                                    NORM0013
      I1=M(1)                                                           NORM0014
      I2=M(2)                                                           NORM0015
      I3=M(3)                                                           NORM0016
      XC=(Y(I3)-Y(I1))*(Z(I2)-Z(I1))-(Y(I2)-Y(I1))*(Z(I3)-Z(I1))       NORM0017
      YC=(X(I2)-X(I1))*(Z(I3)-Z(I1))-(X(I3)-X(I1))*(Z(I2)-Z(I1))       NORM0018
      ZC=(X(I3)-X(I1))*(Y(I2)-Y(I1))-(X(I2)-X(I1))*(Y(I3)-Y(I1))       NORM0019
      D=SQRT(XC*XC+YC*YC+ZC*ZC)                                         NORM0020
      IF(D.LT.0.0001)RETURN                                             NORM0021
      XC=XC/D                                                           NORM0022
      YC=YC/D                                                           NORM0023
      ZC=ZC/D                                                           NORM0024
      IF(XC.EQ.0.0)GO TO 10                                             NORM0025
      IF(XC.GT.0.0)GO TO 30                                             NORM0026
      XC=-XC                                                            NORM0027
      YC=-YC                                                            NORM0028
      ZC=-ZC                                                            NORM0029
      GO TO 30                                                          NORM0030
   10 IF(YC.EQ.0.0)GO TO 20                                             NORM0031
      IF(YC.GT.0.0)GO TO 30                                             NORM0032
      YC=-YC                                                            NORM0033
      ZC=-ZC                                                            NORM0034
      GO TO 30                                                          NORM0035
   20 ZC=ABS(ZC)                                                        NORM0036
   30 CONTINUE                                                          NORM0037
      CALL ITST(M,XC,YC,ZC)                                             NORM0038
      RETURN                                                            NORM0039
      END                                                               NORM0040
      SUBROUTINE OUTPUT(IOUT,ICLER)                                     OTPT0001
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCOTPT0002
C                                                                       COTPT0003
C  SUBROUTINES OUTPUT COMPLETES PROCESSING THE MESH AND DETERMINES      COTPT0004
C  THE PROPER OUTPUT FORMATS.                                           COTPT0005
C                                                                       COTPT0006
C  CALLED BY:  MAIN                                                     COTPT0007
C                                                                       COTPT0008
C  CALLS SUBROUTINES:  BCS3    BOUT1   BOUT2   EL1D    EL2D    EL3D    COTPT0009
C     FN4      ICN3    ICN4                                            COTPT0010
C                                                                       COTPT0011
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCOTPT0012
      DIMENSION IB(6),IB2(6),IEA(6),XMAX(3),XMIN(3),DIF(3),             OTPT0013
     1IF(100),IF2(100),IBD(6)                                           OTPT0014
      COMMON/SYSARY/X(8000),Y(8000),Z(8000),N(8000)                     OTPT0015
     1,IBB(8000),IFLAG(8000),V(8000,3)                                  OTPT0016
      COMMON/KNTS/NODECT,IBEAM,ITWC,ITHRE,LOAD,LINES,MIN                OTPT0017
      COMMON/JUNK/LOOF,IAXI,MOVIE,IE1,IE2,IE3,NPLOT,TITLE(20),ICUT2     OTPT0018
      COMMON/PARTS/NPARTS,XMAXP(40,3),XMINP(40,3)                       OTPT0019
      DATA IBD/2HDX,2HDY,2HDZ,2HPX,2HRY,2HRZ/,YES/1HY/,NBLANK/2H  /     OTPT0020
      DO 20 I=1,8000                                                    OTPT0021
      N(I)=0                                                            OTPT0022
      IBB(I)=0                                                          OTPT0023
      DO 10 J=1,3                                                       OTPT0024
   10 V(I,J)=0.0                                                        OTPT0025
   20 CONTINUE                                                          OTPT0026
      DO 30 I=1,100                                                     OTPT0027
      IF(I)=0                                                           OTPT0028
      IF2(I)=0                                                          OTPT0029
   30   CONTINUE                                                       OTPT0030
      ICUT=ICUT2                                                        OTPT0031
      DIFMAX=0.0                                                        OTPT0032
```

```
      DO 40 I=1,3                                                       OTPTC033
      XMAX(I)=-999.0                                                    OTPTC034
   40 XMIN(I)=999.0                                                     OTPTC035
      DO 50 I=1,NPARTS                                                  OTPTC036
      DO 50 J=1,3                                                       OTPTC037
      XMAX(J)=AMAX1(XMAX(J),XMAXP(I,J))                                 OTPTC038
      XMIN(J)=AMIN1(XMIN(J),XMINP(I,J))                                 OTPTC039
   50 CONTINUE                                                          OTPTC040
      DO 60 I=1,3                                                       OTPTC041
      DIF(I)=XMAX(I)-XMIN(I)                                            OTPTC042
   60 DIFMAX=AMAX1(DIFMAX,DIF(I))                                       OTPTC043
      DO 70 I=1,3                                                       OTPTC044
      DIF(I)=DIF(I)/100.0                                               OTPTC045
   70 IF(DIFMAX.EQ.DIF(I))NDIR=I                                        OTPTC046
      CALL ICN4                                                         OTPTC047
      DO 80 I=1,NODECT                                                  OTPTC048
      CALL ICN3(XM,YM,ZM,IBBM)                                          OTPTC049
      X(I)=XM                                                           OTPTC050
      Y(I)=YM                                                           OTPTC051
      Z(I)=ZM                                                           OTPTC052
      IBB(I)=IBBM*3                                                     OTPTC053
      CALL FN4(XM,YM,ZM,XMAX,XMIN,DIF,NDIR,NPOS)                        OTPTC054
      IF2(NPOS)=IF2(NPOS)+1                                             OTPTC055
   80 CONTINUE                                                          OTPTC056
      DO 90 I=2,100                                                     OTPTC057
      IF2(I)=IF2(I)+IF2(I-1)                                            OTPTC058
   90 IF(I)=IF2(I-1)+1                                                  OTPTC059
      DO 100 I=1,NODECT                                                 OTPTC060
      XM=X(I)                                                           OTPTC061
      YM=Y(I)                                                           OTPTC062
      ZM=Z(I)                                                           OTPTC063
      CALL FN4(XM,YM,ZM,XMAX,XMIN,DIF,NDIR,NPOS)                        OTPTC064
      IFLAG(IF(NPOS))=I                                                 OTPTC065
      IF(NPOS)=IF(NPOS)+1                                               OTPTC066
  100 CONTINUE                                                          OTPTC067
C                                                                       OTPTC068
C  TOLERENCE NODES                                                      OTPTC069
C                                                                       OTPTC070
      N(1)=1                                                            OTPTC071
  110 CONTINUE                                                          OTPTC072
      NODE=2                                                            OTPTC073
      DO 150 MAX=2,NODECT                                               OTPTC074
      N(MAX)=NODE                                                       OTPTC075
      M1=MAX                                                            OTPTC076
      CALL BCS3(IB,M1,1)                                                OTPTC077
      XM=X(MAX)                                                         OTPTC078
      YM=Y(MAX)                                                         OTPTC079
      ZM=Z(MAX)                                                         OTPTC080
      CALL FN4(XM,YM,ZM,XMAX,XMIN,DIF,NDIR,NPOS)                        OTPTC081
      IBEG=1                                                            OTPTC082
      IEND=NODECT                                                       OTPTC083
      IF(NPOS.GT.2)IBEG=IF2(NPOS-2)+1                                   OTPTC084
      IF(NPOS.LT.99)IEND=IF2(NPOS+1)                                    OTPTC085
      DO 130 II=IBEG,IEND                                               OTPTC086
      I=IFLAG(II)                                                       OTPTC087
      IF(I.GE.MAX)GO TO 130                                             OTPTC088
      XDIF=ABS(X(I)-XM)                                                 OTPTC089
      IF(XDIF.GE.TOLER)GO TO 130                                        OTPTC090
      YDIF=ABS(Y(I)-YM)                                                 OTPTC091
      IF(YDIF.GE.TOLER)GO TO 130                                        OTPTC092
      ZDIF=ABS(Z(I)-ZM)                                                 OTPTC093
```

```
      IF(ZDIF.GE.TOLER)GO TO 130                                        OTPT0094
      N(MAX)=N(I)                                                       OTPT0095
      NODE=NODE-1                                                       OTPT0096
      CALL BCS3(IB2,I,1)                                                OTPT0097
      DO 120 J=1,6                                                      OTPT0098
      IB2(J)=MAXC(IB2(J),IB(J))                                         OTPT0099
  120 CONTINUE                                                          OTPT0100
      CALL BCS3(IB2,I,0)                                                OTPT0101
      GO TO 140                                                         OTPT0102
  130 CONTINUE                                                          OTPT0103
  140 CONTINUE                                                          OTPT0104
      NODE=NODE+1                                                       OTPT0105
  150 CONTINUE                                                          OTPT0106
        NDEL=MAX-NODE                                                   OTPT0107
        WRITE(5,240)NDEL,TOLER                                          OTPT0108
        READ(5,250)CMD                                                  OTPT0109
        IF(CMD.NE.YES)GO TO 160                                         OTPT0110
        WRITE(5,260)                                                    OTPT0111
        READ(5,*)TOLER                                                  OTPT0112
        GO TO 110                                                       OTPT0113
  160   CONTINUE                                                        OTPT0114
      DO 170 I=1,8000                                                   OTPT0115
  170 IFLAG(I)=0                                                        OTPT0116
C                                                                       OTPT0117
C  PROCESS ELEMENT DATA                                                 OTPT0118
C                                                                       OTPT0119
      IF(IBEAM.NE.0)CALL EL10(ICLT,NEL2)                                OTPT0120
      IF(ITWO.NE.0)CALL EL20(ICLT,NEL2)                                 OTPT0121
      IF(ITHRE.NE.0)CALL EL30(ICLT,NEL2)                                OTPT0122
      ILAST=MAX-1                                                       OTPT0123
C                                                                       OTPT0124
C  CHECK BOUNDARY CONDITIONS FOR PLATES                                 OTPT0125
C                                                                       OTPT0126
      CALL BCUT1(ILAST)                                                 OTPT0127
  180 CONTINUE                                                          OTPT0128
      ILAST=0                                                           OTPT0129
C                                                                       OTPT0130
C  OUTPUT NODES AND BOUNDARY CONDITIONS                                 OTPT0131
C                                                                       OTPT0132
      IF(IOUT.EQ.3)WRITE(11,280)                                        OTPT0133
      IF(IOUT.EQ.3)WRITE(15,290)                                        OTPT0134
      MAX1=MAX-1                                                        OTPT0135
      DO 230 I=1,MAX1                                                   OTPT0136
      IF(N(I).LE.ILAST)GO TO 230                                        OTPT0137
      ILAST=ILAST+1                                                     OTPT0138
      I1=I                                                              OTPT0139
      CALL BCS3(IB,I1,1)                                                OTPT0140
      NCONST=0                                                          OTPT0141
      NZERO=0                                                           OTPT0142
      IBO=0                                                             OTPT0143
      DO 190 J=1,6                                                      OTPT0144
      IF(IB(J).EQ.0)NZERO=NZERO+1                                       OTPT0145
      IBA(J)=NBLANK                                                     OTPT0146
      IF(IB(J).EQ.2)IB(J)=0                                             OTPT0147
      IF(IB(J).EQ.3)IB(J)=1                                             OTPT0148
      IF(IB(J).EQ.1)NCONST=NCONST+1                                     OTPT0149
  190 CONTINUE                                                          OTPT0150
      IF(IOUT.NE.2)GO TO 200                                            OTPT0151
      IF(IB(1).EQ.1)IBA(1)=IBO(1)                                       OTPT0152
      IF(IB(2).EQ.1)IBA(2)=IBO(2)                                       OTPT0153
      IF(IB(3).EQ.1)IBA(3)=IBO(3)                                       OTPT0154
```

```
      IF(IB(4).EQ.1)IBA(4)=IBD(4)                                          OTPT0155
      IF(IB(5).EQ.1)IBA(5)=IBD(5)                                          OTPT0156
      IF(IB(6).EQ.1)IBA(6)=IBD(6)                                          OTPT0157
  200 IF(IOUT.NE.3)GO TO 220                                               OTPT0158
      DO 210 J=1,6                                                         OTPT0159
      IF(IAXI.EQ.1.AND.J.EQ.6)GO TO 210                                    OTPT0160
      IF(IB(J).NE.0)IBQ=10*IBQ+J                                           OTPT0161
  210 CONTINUE                                                             OTPT0162
  220 CONTINUE                                                             OTPT0163
      IF(IOUT.EQ.1)WRITE(11,340)ILAST,IB,X(I),Y(I),Z(I)                    OTPT0164
      IF(MOVIE.EQ.1)WRITE(21,320)ILAST,X(I),Y(I),Z(I)                      OTPT0165
      IF(IOUT.EQ.2)WRITE(11,320)ILAST,X(I),Y(I),Z(I)                       OTPT0166
      IF(IOUT.EQ.3)WRITE(11,320)ILAST,X(I),Y(I),Z(I)                       OTPT0167
      IF(IOUT.EQ.2.AND.NZERO.LT.6.AND.NCONST.GT.0)WRITE(15,330)ILAST,IBAOTPT0168
      IF(IOUT.EQ.3.AND.IBQ.NE.0)WRITE(15,300)ILAST,IBQ                     OTPT0169
      IF(IOUT.EQ.2.AND.NZERO.EQ.6)WRITE(15,310)ILAST                       OTPT0170
  230 CONTINUE                                                             OTPT0171
C                                                                          OTPT0172
C  OUTPUT BOUNDARY ELEMENTS                                                OTPT0173
C                                                                          OTPT0174
      IF(LOOP.NE.1)CALL BOUT2(ILAST,IOUT)                                  OTPT0175
      IZERO=0                                                              OTPT0176
      IF(LOOP.EQ.1)WRITE(6,350)ILAST,IZERO                                 OTPT0177
C                                                                          OTPT0178
C  OUTPUT LOAD DATA                                                        OTPT0179
C                                                                          OTPT0180
      CALL LOADS3                                                          OTPT0181
      STOP                                                                 OTPT0182
  240 FORMAT(3H < ,I5,36H NODES WERE DELETED BY TOLERENCING >,/,           OTPT0183
     111H < TOLER = ,F9.4,40H  WOULD YOU LIKE TO TRY ANOTHER VALUE? >      OTPT0184
     2,$)                                                                  OTPT0185
  250   FORMAT(A1)                                                         OTPT0186
  260 FORMAT(30H < INPUT NEW VALUE FOR TOLER >,$)                          OTPT0187
  270 FORMAT(5X,6I5,3F10.0)                                                OTPT0188
  280 FORMAT(5HNODES,/,17HNODE.NUMBER X Y Z)                               OTPT0189
  290  FORMAT(10HRESTRAINTS,/,21HNODE.NUMBER DIRECTION)                    OTPT0190
  300 FORMAT(I5,3X,I6)                                                     OTPT0191
  310 FORMAT(I5,10X,7HFIX  NM)                                             OTPT0192
  320 FORMAT(I10,3F10.4)                                                   OTPT0193
  330 FORMAT(I5,10X,5HFIX  ,6A2)                                           OTPT0194
  340 FORMAT(7I5,3F10.4)                                                   OTPT0195
  350 FORMAT(16H NUMBER OF NODES,/,21H AFTER TOLERENCING = ,I5,            OTPT0196
     1/,31H NUMBER OF BOUNDARY ELEMENTS = ,I5)                             OTPT0197
      END                                                                  OTPT0198
      SUBROUTINE PARABO(X1,Y1,Z1,X2,Y2,Z2,X3,Y3,Z3,IB)                     PRBC0001
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCPRBC0002
C                                                                          CPRBC0003
C  SUBROUTINE PARABO STORES COMMANDS FOR MOVING NODES TO FIT A PARABOLA CPRBC0004
C                                                                          CPRBC0005
C  CALLED BY:  CIRCL2  SEQ2                                                CPRBC0006
C                                                                          CPRBC0007
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCPRBC0008
      DIMENSION IB(6)                                                      PRBC0009
      COMMON/VECFNC/NFN,IFN(100),IAPLY(100,6),COEF(100,5)                  PRBC0010
      NFN=NFN+1                                                            PRBC0011
      IFN(NFN)=1                                                           PRBC0012
      DO 10 I=1,6                                                          PRBC0013
      IAPLY(NFN,I)=IB(I)                                                   PRBC0014
   10 CONTINUE                                                             PRBC0015
      COEF(NFN,1)=4*X2-2*X1-2*X3                                           PRBC0016
      COEF(NFN,2)=4*Y2-2*Y1-2*Y3                                           PRBC0017
```

```
      COEF(NFN,3)=4*Z2-2*Z1-2*Z3                                        PRBC0018
      RETURN                                                            PRBC0019
      END                                                               PRBC0020
      SUBROUTINE PART                                                   PART0001
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCPART0002
C                                                                       CPART0003
C  SUBROUTINE PART DETERMINES WHICH PARTS OF THE STRUCTURE ARE TO BE    CPART0004
C  PLOTTED.                                                             CPART0005
C                                                                       CPART0006
C  CALLED BY:  INPLOT                                                   CPART0007
C                                                                       CPART0008
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCPART0009
      DIMENSION IPTS(40)                                                PART0010
      COMMON/PARTS2/PRTLST(40)                                          PART0011
      DO 10 I=1,40                                                      PART0012
      PRTLST(I)=1.0                                                     PART0013
   10 CONTINUE                                                          PART0014
C                                                                       PART0015
C  INPUT NUMBER OF PARTS TO BE PLOTTED                                  PART0016
C                                                                       PART0017
      WRITE(5,40)                                                       PART0018
      READ(5,*)NPTS                                                     PART0019
      IF(NPTS.EQ.0)RETURN                                               PART0020
C                                                                       PART0021
C  INPUT PARTS                                                          PART0022
C                                                                       PART0023
      WRITE(5,50)                                                       PART0024
      READ(5,*)(IPTS(I),I=1,NPTS)                                       PART0025
      DO 20 I=1,40                                                      PART0026
      PRTLST(I)=0.0                                                     PART0027
   20 CONTINUE                                                          PART0028
      DO 30 I=1,NPTS                                                    PART0029
      PRTLST(IPTS(I))=1.0                                               PART0030
   30 CONTINUE                                                          PART0031
      RETURN                                                            PART0032
   40 FORMAT(34H < NUMBER OF PARTS TO BE PLOTTED >,$)                   PART0033
   50 FORMAT(15H < LIST PARTS >,$)                                      PART0034
      END                                                               PART0035
      SUBROUTINE PLOT(X,Y,IPEN)                                         PLOT0001
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCPLOT0002
C                                                                       CPLOT0003
C  SUBROUTINE PLOT IS SIMILAR TO THE CALCOMP ROUTINE PLOT. A VECTOR IS  CPLOT0004
C  DRAWN TO POINT X,Y WITH IPEN=3 FOR PEN UP AND IPEN=2 FOR PEN DOWN.   CPLOT0005
C                                                                       CPLOT0006
C  CALLED BY:  INPLOT                                                   CPLOT0007
C                                                                       CPLOT0008
C  CALLS TCS ROUTINES: DRWABS  MOVABS                                   CPLOT0009
C                                                                       CPLOT0010
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCPLOT0011
      COMMON/SCALES/FACTOR,ATRAN,BTRAN,F2                               PLOT0012
      XO=(X-BTRAN)/FACTOR*F2+512.0                                      PLOT0013
      YO=(Y-ATRAN)/FACTOR*F2+384.0                                      PLOT0014
      IX=IFIX(XO)                                                       PLOT0015
      IY=IFIX(YO)                                                       PLOT0016
      IF(IPEN.EQ.2)CALL DRWABS(IX,IY)                                   PLOT0017
      IF(IPEN.EQ.3)CALL MOVABS(IX,IY)                                   PLOT0018
      RETURN                                                            PLOT0019
      END                                                               PLOT0020
      SUBROUTINE PLOTMK(IM,JM,KM,IX,JX,KX,ND,M1,M2,M3,N)                PLTM0001
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCPLTM0002
C                                                                       CPLTM0003
```

```
C  SUBROUTINE PLOTMK OUTPUTS THE LINES IN A REGION WHICH ARE TO BE      CPLTM0004
C  PLOTTED.                                                             CPLTM0005
C                                                                       CPLTM0006
C  CALLED BY:  ELTHRE  ELTWO                                            CPLTM0007
C                                                                       CPLTM0008
C  CALLS SUBROUTINES:  IOP1                                             CPLTM0009
C                                                                       CPLTM0010
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCPLTM0011
      DIMENSION N(M1,M2,M3)                                             PLTM0012
      COMMON/JUNK/LCOF,IAXI,MOVIE,IE1,IE2,IE3,NPLOT,TITLE(20),IOUT2     PLTM0013
      IF(NPLOT.EQ.0)RETURN                                              PLTM0014
      INCR=ND+1                                                         PLTM0015
      IF(IM.EQ.IX)GO TO 50                                              PLTM0016
      IF=IX-1                                                           PLTM0017
      DO 40 K=KM,KX,INCR                                                PLTM0018
      DO 30 J=JM,JX,INCR                                                PLTM0019
      DO 20 I=IM,IF,INCR                                                PLTM0020
      N1=N(I,J,K)                                                       PLTM0021
      N2=0                                                              PLTM0022
      N3=N(I+1,J,K)                                                     PLTM0023
      IF(ND.EQ.0)GO TO 10                                               PLTM0024
      N2=N3                                                             PLTM0025
      N3=N(I+2,J,K)                                                     PLTM0026
   10 CALL IOP1(N1,N2,N3)                                               PLTM0027
   20 CONTINUE                                                          PLTM0028
   30 CONTINUE                                                          PLTM0029
   40 CONTINUE                                                          PLTM0030
   50 CONTINUE                                                          PLTM0031
      IF(JM.EQ.JX)GO TO 100                                             PLTM0032
      JF=JX-1                                                           PLTM0033
      DO 90 K=KM,KX,INCR                                                PLTM0034
      DO 80 I=IM,IX,INCR                                                PLTM0035
      DO 70 J=JM,JF,INCR                                                PLTM0036
      N1=N(I,J,K)                                                       PLTM0037
      N2=0                                                              PLTM0038
      N3=N(I,J+1,K)                                                     PLTM0039
      IF(ND.EQ.0)GO TO 60                                               PLTM0040
      N2=N3                                                             PLTM0041
      N3=N(I,J+2,K)                                                     PLTM0042
   60 CALL IOP1(N1,N2,N3)                                               PLTM0043
   70 CONTINUE                                                          PLTM0044
   80 CONTINUE                                                          PLTM0045
   90 CONTINUE                                                          PLTM0046
  100 CONTINUE                                                          PLTM0047
      IF(KM.EQ.KX)RETURN                                                PLTM0048
      KF=KX-1                                                           PLTM0049
      DO 140 I=IM,IX,INCR                                               PLTM0050
      DO 130 J=JM,JX,INCR                                               PLTM0051
      DO 120 K=KM,KF,INCR                                               PLTM0052
      N1=N(I,J,K)                                                       PLTM0053
      N2=0                                                              PLTM0054
      N3=N(I,J,K+1)                                                     PLTM0055
      IF(ND.EQ.0)GO TO 110                                              PLTM0056
      N2=N3                                                             PLTM0057
      N3=N(I,J,K+2)                                                     PLTM0058
  110 CALL IOP1(N1,N2,N3)                                               PLTM0059
  120 CONTINUE                                                          PLTM0060
  130 CONTINUE                                                          PLTM0061
  140 CONTINUE                                                          PLTM0062
      RETURN                                                            PLTM0063
      END                                                               PLTM0064
```

```
      SUBROUTINE PLOTS(NA,NB,NC)                                        PLTS0001
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCPLTS0002
C                                                                       CPLTS0003
C  SUBROUTINE PLOTS INITIALIZES TCS PLOTTING ROUTINES                   CPLTS0004
C                                                                       CPLTS0005
C  CALLED BY:  INPLOT                                                   CPLTS0006
C                                                                       CPLTS0007
C  CALLS TCS ROUTINES: INITT  TSEND                                     CPLTS0008
C                                                                       CPLTS0009
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCPLTS0010
      CALL INITT(3C)                                                    PLTS0011
      CALL TSEND                                                        PLTS0012
      RETURN                                                            PLTS0013
      END                                                               PLTS0014
      SUBROUTINE SCALE(IAXIS,ANGA,ANGB)                                 SCLE0001
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCSCLE0002
C                                                                       CSCLE0003
C  SUBROUTINE SCALE DETERMINES THE SCALE FACTORS FOR A PLOT             CSCLE0004
C                                                                       CSCLE0005
C  CALLED BY: INPLOT                                                    CSCLE0006
C                                                                       CSCLE0007
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCSCLE0008
      COMMON/SCALES/FACTOR,ATRAN,BTRAN,F2                               SCLE0009
      COMMON/PARTS/NPARTS,XMAXP(40,3),XMINP(40,3)                       SCLE0010
      COMMON/PARTS2/PRTLST(40)                                          SCLE0011
      DIMENSION VEC(3),XMAX(3),XMIN(3)                                  SCLE0012
      DATA F2/1.C/                                                      SCLE0013
      SUM=0.0                                                           SCLE0014
      DO 10 I=1,3                                                       SCLE0015
      XMAX(I)=-999.0                                                    SCLE0016
      XMIN(I)=999.0                                                     SCLE0017
   10 CONTINUE                                                          SCLE0018
C                                                                       SCLE0019
C  DETERMINES THE LIMITS OF THE PARTS TO BE PLOTTED                     SCLE0020
C                                                                       SCLE0021
      DO 20 I=1,3                                                       SCLE0022
      DO 20 J=1,NPARTS                                                  SCLE0023
      IF(PRTLST(J).EQ.0.0)GO TO 20                                      SCLE0024
      XMAX(I)=AMAX1(XMAX(I),XMAXP(J,I))                                 SCLE0025
      XMIN(I)=AMIN1(XMIN(I),XMINP(J,I))                                 SCLE0026
   20 CONTINUE                                                          SCLE0027
      DO 30 I=1,3                                                       SCLE0028
      VEC(I)=XMAX(I)-XMIN(I)                                            SCLE0029
      SUM=SUM+VEC(I)*VEC(I)                                             SCLE0030
   30 CONTINUE                                                          SCLE0031
      FACTOR=SQRT(SUM)/850.0                                            SCLE0032
      XM=(XMAX(1)+XMIN(1))/2.0                                          SCLE0033
      YM=(XMAX(2)+XMIN(2))/2.0                                          SCLE0034
      ZM=(XMAX(3)+XMIN(3))/2.0                                          SCLE0035
C                                                                       SCLE0036
C  DETERMINE THE SIZE OF THE PLOT ON THE TERMINAL                       SCLE0037
C                                                                       SCLE0038
      SA=SIN(ANGA/57.2958)                                              SCLE0039
      SB=SIN(ANGB/57.2958)                                              SCLE0040
      CA=COS(ANGA/57.2958)                                              SCLE0041
      CB=COS(ANGB/57.2958)                                              SCLE0042
      CBSA=CB*SA                                                        SCLE0043
      SBSA=SB*SA                                                        SCLE0044
      IF(IAXIS.EQ.1)GO TO 40                                            SCLE0045
      IF(IAXIS.EQ.2)GO TO 50                                            SCLE0046
      IF(IAXIS.EQ.3)GO TO 60                                            SCLE0047
```

```
   40 BTRAN=-ZM*SB+XM*CA                                                SCLE0048
      ATRAN=-ZM*CBSA-XM*SBSA+YM*CA                                      SCLE0049
      GO TO 70                                                          SCLE0050
   50 BTRAN=-XM*SB+YM*CA                                                SCLE0051
      ATRAN=-XM*CBSA-YM*SBSA+ZM*CA                                      SCLE0052
      GO TO 70                                                          SCLE0053
   60 BTRAN=-YM*SB+ZM*CA                                                SCLE0054
      ATRAN=-YM*CBSA-ZM*SBSA+XM*CA                                      SCLE0055
   70 CONTINUE                                                          SCLE0056
      RETURN                                                            SCLE0057
      END                                                               SCLE0058
      SUBROUTINE SCALE2                                                 SCL20001
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCSCL20002
C                                                                       CSCL20003
C  SUBROUTINE SCALE2 ALLOWS THE USER TO INPUT A SCALE FACTOR            CSCL20004
C                                                                       CSCL20005
C  CALLED BY:  INPLOT                                                   CSCL20006
C                                                                       CSCL20007
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCSCL20008
      DIMENSION IPTS(20)                                                SCL20009
      COMMON/SCALES/FACTOR,ATRAN,BTRAN,F2                               SCL20010
      WRITE(5,10)                                                       SCL20011
      READ(5,*)F2                                                       SCL20012
      RETURN                                                            SCL20013
   10 FORMAT(17H < SCALE FACTOR >,$)                                    SCL20014
      END                                                               SCL20015
      SUBROUTINE SEG(NI,NJ,NK)                                          SEQ0001
      DIMENSION II(20,3),M(3),IMX(6),NI(16),NJ(16),NK(16)               SEQ0002
      COMMON/REGION/IARRAY(200,6),NREG                                  SEQ0003
      EQUIVALENCE (IMX(1),IM),(IMX(2),JM),(IMX(3),KM),(IMX(4),IX),      SEQ0004
     1(IMX(5),JX),(IMX(6),KX)                                           SEQ0005
      DO 10 I=1,16                                                      SEQ0006
      II(I,1)=NI(I)                                                     SEQ0007
      II(I,2)=NJ(I)                                                     SEQ0008
   10 II(I,3)=NK(I)                                                     SEQ0009
      DO 20 I=1,3                                                       SEQ0010
      DO 20 J=1,20                                                      SEQ0011
      IF(II(J,I).NE.0)M(I)=J                                            SEQ0012
   20 CONTINUE                                                          SEQ0013
      DO 30 I=1,3                                                       SEQ0014
   30 IF(II(M(I),I).GT.0)M(I)=M(I)-1                                    SEQ0015
      DO 130 I=1,M(1)                                                   SEQ0016
      DO 120 J=1,M(2)                                                   SEQ0017
      DO 110 K=1,M(3)                                                   SEQ0018
      IM=II(I,1)                                                        SEQ0019
      JM=II(J,2)                                                        SEQ0020
      KM=II(K,3)                                                        SEQ0021
      IX=II(I+1,1)                                                      SEQ0022
      JX=II(J+1,2)                                                      SEQ0023
      KX=II(K+1,3)                                                      SEQ0024
      IF(IM.EQ.0.OR.JM.EQ.0.OR.KM.EQ.0)GO TO 110                        SEQ0025
      IF(IM.LE.0.OR.JM.LE.0.OR.KM.LE.0)GO TO 50                         SEQ0026
      IF(IX.LE.0.OR.JX.LE.0.OR.KX.LE.0)GO TO 110                        SEQ0027
      NREG=NREG+1                                                       SEQ0028
      DO 40 L=1,6                                                       SEQ0029
   40 IARRAY(NREG,L)=IMX(L)                                             SEQ0030
      GO TO 110                                                         SEQ0031
   50 CONTINUE                                                          SEQ0032
      IF(IM.GE.0)GO TO 70                                               SEQ0033
      IF(JX.EQ.0.OR.KX.EQ.0)GO TO 70                                    SEQ0034
      NREG=NREG+1                                                       SEQ0035
```

```
      DO 60 L=1,6                                                        SEQC0036
   60 IARRAY(NREG,L)=IABS(IMX(L))                                        SEQC0037
      IARRAY(NREG,4)=-IM                                                 SEQ0038
   70 IF(JM.GE.0)GO TO 90                                                SEQC039
      IF(IX.EQ.0.OR.KX.EQ.0)GO TO 90                                     SEQC040
      NREG=NREG+1                                                        SEQ0041
      DO 80 L=1,6                                                        SEQC042
   80 IARRAY(NREG,L)=IABS(IMX(L))                                        SEQC043
      IARRAY(NREG,5)=-JM                                                 SEQC044
   90 IF(KM.GE.0)GO TO 110                                               SEQC045
      IF(IX.EQ.0.OR.JX.EQ.0)GO TO 110                                    SEQC046
      NREG=NREG+1                                                        SEQC047
      DO 100 L=1,6                                                       SEQC048
  100 IARRAY(NREG,L)=IABS(IMX(L))                                        SEQC049
      IARRAY(NREG,6)=-KM                                                 SEQC050
  110 CONTINUE                                                           SEQC051
  120 CONTINUE                                                           SEQ0052
  130 CONTINUE                                                           SEQ0053
      RETURN                                                             SEQ0054
      END                                                                SEQC055
      SUBROUTINE SEQ2(IDIM,JDIM,KDIM,XX,YY,ZZ)                           SEQ20001
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCSEQ20002
C                                                                       CSEQ20003
C  SUBROUTINE SEQ2 DETERMINES COORDINATES IN THE INDEX SPACE AND        CSEQ20004
C  PROCESSES THE FUNCTIONS WHICH MODIFY THE MESH.                       CSEQ20005
C                                                                       CSEQ20006
C  CALLED BY:  DLEE                                                     CSEQ20007
C                                                                       CSEQ20008
C  CALLS SUBROUTINES:  BCS     CIRCLE  CIRCL2  CYLIND  DOUBLE  LOADS    CSEQ20009
C       MIDSID  PARABO  SPHERI                                          CSEQ20010
C                                                                       CSEQ20011
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCSEQ20012
      DIMENSION X(16),Y(16),Z(16)                                       SEQ20013
      DIMENSION CC(16),IRX(16),IRY(16),IRZ(16)                          SEQ20014
      DIMENSION II(16),JJ(16),KK(16),IBC(6)                             SEQ20015
      DIMENSION XX(IDIM,JDIM,KDIM),YY(IDIM,JDIM,KDIM),ZZ(IDIM,          SEQ20016
     1JDIM,KDIM)                                                        SEQ20017
      COMMON/SEQC1/IB(6),I(16),J(16),K(16)                              SEQ20018
      COMMON/PROP/MAT,NOCT,IC2,IT,TH,N8,N20                             SEQ20019
      DATA CHAR/1HE/,NCHAR1/1H /,NCHARA/1HA/,NCHARB/1HB/,NCHARL         SEQ20020
     1/1HL/,NCHARE/1HE/,NCHARM/1HM/,NCHARD/1HD/,NCHARI/1HI/             SEQ20021
     2,NCHARJ/1HJ/,NCHARK/1HK/,NCHARQ/1HQ/,NCHARR/1HR/                  SEQ20022
   10 READ(11,61C,END=600)S                                             SEQ20023
      NI=16                                                             SEQ20024
      NJ=16                                                             SEQ20025
      NK=16                                                             SEQ20026
      NBLOCK=NI*NJ*NK                                                   SEQ20027
      READ(11,62C)(I(M),M=1,NI)                                         SEQ20028
      READ(11,62C)(J(M),M=1,NJ)                                         SEQ20029
      READ(11,62C)(K(M),M=1,NK)                                         SEQ2003C
      IF=1                                                              SEQ20031
      JF=1                                                              SEQ20032
      KF=1                                                              SEQ20033
   20 IF(I(NI).NE.0)GO TO 30                                            SEQ20034
      NI=NI-1                                                           SEQ20035
      GO TO 20                                                          SEQ20036
   30 IF(J(NJ).NE.0)GO TO 40                                            SEQ20037
      NJ=NJ-1                                                           SEQ20038
      GO TO 30                                                          SEQ20039
   40 IF(K(NK).NE.0)GO TO 50                                            SEQ2004C
      NK=NK-1                                                           SEQ20041
```

```
      GO TO 40                                                          SEQ20042
   50 CONTINUE                                                          SEQ20043
      CALL DOUBLE(I,J,K)                                                SEQ20044
      CALL SEQ(I,J,K)                                                   SEQ20045
C                                                                       SEQ20046
C  SET UP COORDINATES OF UNDEFORMED REGION                              SEQ20047
C                                                                       SEQ20048
      READ(11,630)(X(M),M=1,NI)                                         SEQ20049
      READ(11,630)(Y(M),M=1,NJ)                                         SEQ20050
      READ(11,630)(Z(M),M=1,NK)                                         SEQ20051
      DO 60 M=1,16                                                      SEQ20052
      IF(I(M).LT.0)I(M)=-I(M)                                           SEQ20053
      IF(J(M).LT.0)J(M)=-J(M)                                           SEQ20054
      IF(K(M).LT.0)K(M)=-K(M)                                           SEQ20055
   60 CONTINUE                                                          SEQ20056
      DO 90 M1=1,NI                                                     SEQ20057
      DO 80 M2=1,NJ                                                     SEQ20058
      DO 70 M3=1,NK                                                     SEQ20059
      IF(I(M1).EQ.0)GO TO 90                                            SEQ20060
      IF(J(M2).EQ.0)GO TO 80                                            SEQ20061
      IF(K(M3).EQ.0)GO TO 70                                            SEQ20062
      XX(I(M1),J(M2),K(M3))=X(M1)                                       SEQ20063
      YY(I(M1),J(M2),K(M3))=Y(M2)                                       SEQ20064
      ZZ(I(M1),J(M2),K(M3))=Z(M3)                                       SEQ20065
   70 CONTINUE                                                          SEQ20066
   80 CONTINUE                                                          SEQ20067
   90 CONTINUE                                                          SEQ20068
C                                                                       SEQ20069
C  READ FUNCTION CARDS                                                  SEQ20070
C                                                                       SEQ20071
  100 READ(11,640)ISEQ,IO,JO,KO,IT,(CO(M),M=1,6)                       SEQ20072
      IF(ISEQ.EQ.NCHARA)GO TO 310                                       SEQ20073
      IF(ISEQ.EQ.NCHARB)GO TO 400                                       SEQ20074
      IF(ISEQ.EQ.NCHARL)GO TO 420                                       SEQ20075
      IF(ISEQ.EQ.NCHARD)GO TO 440                                       SEQ20076
      IF(ISEQ.EQ.NCHARM)GO TO 460                                       SEQ20077
      IF(ISEQ.EQ.NCHARE)GO TO 560                                       SEQ20078
      IF(ISEQ.EQ.NCHARQ)GO TO 480                                       SEQ20079
      IF(ISEQ.EQ.NCHARR)GO TO 490                                       SEQ20080
      IF(ISEQ.NE.NCHAR1)GO TO 200                                       SEQ20081
C                                                                       SEQ20082
C  MOVE POINT IO,JO,KO                                                  SEQ20083
C                                                                       SEQ20084
      MX=0                                                              SEQ20085
      MY=0                                                              SEQ20086
      MZ=0                                                              SEQ20087
      IF(IT.EQ.23)MX=1                                                  SEQ20088
      IF(IT.EQ.13)MY=1                                                  SEQ20089
      IF(IT.EQ.12)MZ=1                                                  SEQ20090
      IF(IT.NE.1)GO TO 110                                              SEQ20091
      MY=1                                                              SEQ20092
      MZ=1                                                              SEQ20093
  110 IF(IT.NE.2)GO TO 120                                              SEQ20094
      MX=1                                                              SEQ20095
      MZ=1                                                              SEQ20096
  120 IF(IT.NE.3)GO TO 130                                              SEQ20097
      MX=1                                                              SEQ20098
      MY=1                                                              SEQ20099
  130 CONTINUE                                                          SEQ20100
      ICS=IO                                                            SEQ20101
      ICF=IO                                                            SEQ20102
```

```
      JOS=JC                                                            SEQ20103
      JOF=JC                                                            SEQ20104
      KOS=KC                                                            SEQ20105
      KOF=KC                                                            SEQ20106
      IF(IC.NE.0)GO TO 140                                              SEQ20107
      IOS=1                                                             SEQ20108
      IOF=NI                                                            SEQ20109
  140 IF(JC.NE.0)GO TO 150                                              SEQ20110
      JOS=1                                                             SEQ20111
      JOF=NJ                                                            SEQ20112
  150 IF(KC.NE.0)GO TO 160                                              SEQ20113
      KOS=1                                                             SEQ20114
      KOF=NK                                                            SEQ20115
  160 CONTINUE                                                          SEQ20116
      DO 190 IR=IOS,IOF                                                 SEQ20117
      DO 180 JR=JOS,JOF                                                 SEQ20118
      DO 170 KR=KOS,KOF                                                 SEQ20119
      IO=I(IR)                                                          SEQ20120
      JO=J(JR)                                                          SEQ20121
      KO=K(KR)                                                          SEQ20122
      IF(MX.EQ.0)XX(IO,JO,KO)=CC(1)                                     SEQ20123
      IF(MY.EQ.0)YY(IO,JO,KO)=CC(2)                                     SEQ20124
      IF(MZ.EQ.0)ZZ(IO,JO,KO)=CC(3)                                     SEQ20125
  170 CONTINUE                                                          SEQ20126
  180 CONTINUE                                                          SEQ20127
  190 CONTINUE                                                          SEQ20128
      GO TO 100                                                         SEQ20129
  200 CONTINUE                                                          SEQ20130
C                                                                       SEQ20131
C   CHANGE A PROGRESSION FOR CERTAIN VALUES OF I,J,OR K                 SEQ20132
C                                                                       SEQ20133
      ISEQT=ISEQ                                                        SEQ20134
      IF(IT.EQ.1)ISEQT=NCHARI                                           SEQ20135
      IF(IT.EQ.2)ISEQT=NCHARJ                                           SEQ20136
      IF(IT.EQ.3)ISEQT=NCHARK                                           SEQ20137
      IF(ISEQ.EQ.NCHARI)NC=NI                                           SEQ20138
      IF(ISEQ.EQ.NCHARJ)NC=NJ                                           SEQ20139
      IF(ISEQ.EQ.NCHARK)NC=NK                                           SEQ20140
      IF(NC.GT.6)READ(11,630)(CC(M5),M5=7,14)                           SEQ20141
      IF(NC.GT.14)READ(11,630)(CC(M5),M5=15,16)                         SEQ20142
      DO 210 M1=1,16                                                    SEQ20143
      IRX(M1)=0                                                         SEQ20144
      IRY(M1)=0                                                         SEQ20145
      IRZ(M1)=0                                                         SEQ20146
  210 CONTINUE                                                          SEQ20147
      IF(IO.EQ.-1)READ(11,620)IRX                                       SEQ20148
      IF(JC.EQ.-1)READ(11,620)IRY                                       SEQ20149
      IF(KC.EQ.-1)READ(11,620)IRZ                                       SEQ20150
      IF(IO.GT.0)IRX(1)=IO                                              SEQ20151
      IF(JO.GT.0)IRY(1)=JO                                              SEQ20152
      IF(KC.GT.0)IRZ(1)=KO                                              SEQ20153
      DO 300 IR=1,16                                                    SEQ20154
      DO 290 JR=1,16                                                    SEQ20155
      DO 280 KR=1,16                                                    SEQ20156
      IF(IRX(IR).EC.0.AND.IR.NE.1)GO TO 300                             SEQ20157
      IF(IRY(JR).EC.0.AND.JR.NE.1)GO TO 290                             SEQ20158
      IF(IRZ(KR).EC.0.AND.KR.NE.1)GO TO 280                             SEQ20159
      IC=IRX(IR)                                                        SEQ20160
      JC=IRY(JR)                                                        SEQ20161
      KC=IRZ(KR)                                                        SEQ20162
      DO 270 M1=1,NI                                                    SEQ20163
```

```
      DO 260 M2=1,NJ                                                    SEQ20164
      DO 250 M3=1,NK                                                    SEQ20165
      IF(I(M1).EQ.C.OR.J(M2).EQ.C.OR.K(M3).EQ.0)GO TO 250               SEQ20166
      IF(ISEQ.EQ.NCHARI)M4=M1                                           SEQ20167
      IF(ISEQ.EQ.NCHARJ)M4=M2                                           SEQ20168
      IF(ISEQ.EQ.NCHARK)M4=M3                                           SEQ20169
      IF(ISEQ.EQ.NCHARI.AND.I(M4).EQ.0)GO TO 250                        SEQ20170
      IF(ISEQ.EQ.NCHARJ.AND.J(M4).EQ.0)GO TO 250                        SEQ20171
      IF(ISEQ.EQ.NCHARK.AND.K(M4).EQ.0)GO TO 250                        SEQ20172
      IF(IC.EQ.0)GO TO 220                                              SEQ20173
      II(M1)=I(IC)                                                      SEQ20174
  220 IF(JC.EQ.0)GO TO 230                                              SEQ20175
      JJ(M2)=J(JC)                                                      SEQ20176
  230 IF(KC.EQ.0)GO TO 240                                              SEQ20177
      KK(M3)=K(KC)                                                      SEQ20178
  240 IF(IC.EQ.0)II(M1)=I(M1)                                           SEQ20179
      IF(JC.EQ.0)JJ(M2)=J(M2)                                           SEQ20180
      IF(KC.EQ.0)KK(M3)=K(M3)                                           SEQ20181
      IF(ISEQT.EQ.NCHARI)XX(II(M1),JJ(M2),KK(M3))=CO(M4)                SEQ20182
      IF(ISEQT.EQ.NCHARJ)YY(II(M1),JJ(M2),KK(M3))=CO(M4)                SEQ20183
      IF(ISEQT.EQ.NCHARK)ZZ(II(M1),JJ(M2),KK(M3))=CO(M4)                SEQ20184
  250 CONTINUE                                                          SEQ20185
  260 CONTINUE                                                          SEQ20186
  270 CONTINUE                                                          SEQ20187
  280 CONTINUE                                                          SEQ20188
  290 CONTINUE                                                          SEQ20189
  300 CONTINUE                                                          SEQ20190
      GO TO 100                                                         SEQ20191
C                                                                       SEQ20192
C  READ IN A ARC DEFINITION CARD                                        SEQ20193
C                                                                       SEQ20194
  310 BACKSPACE 11                                                      SEQ20195
      READ(11,660)IB,IA,X2,Y2,Z2,RADIUS                                 SEQ20196
      NCOUNT=0                                                          SEQ20197
      IF(IB(1).NE.IB(4))NCOUNT=NCOUNT+1                                 SEQ20198
      IF(IB(2).NE.IB(5))NCOUNT=NCOUNT+1                                 SEQ20199
      IF(IB(3).NE.IB(6))NCOUNT=NCOUNT+1                                 SEQ20200
      IF(NCOUNT.GT.1)GO TO 330                                          SEQ20201
      CALL FN3                                                          SEQ20202
      IF(IA.EQ.0)GO TO 320                                              SEQ20203
      X3=X2                                                             SEQ20204
      Y3=Y2                                                             SEQ20205
      Z3=Z2                                                             SEQ20206
      IC=IA                                                             SEQ20207
      X1=XX(IB(1),IB(2),IB(3))                                          SEQ20208
      Y1=YY(IB(1),IB(2),IB(3))                                          SEQ20209
      Z1=ZZ(IB(1),IB(2),IB(3))                                          SEQ20210
      X4=XX(IB(4),IB(5),IB(6))                                          SEQ20211
      Y4=YY(IB(4),IB(5),IB(6))                                          SEQ20212
      Z4=ZZ(IB(4),IB(5),IB(6))                                          SEQ20213
  320 CONTINUE                                                          SEQ20214
      IF(IC.EQ.1)CALL PARABO(X1,Y1,Z1,X3,Y3,Z3,X4,Y4,Z4,IB)             SEQ20215
      IF(IC.EQ.2)CALL CIRCLE(X1,Y1,Z1,X3,Y3,Z3,X4,Y4,Z4,IB)             SEQ20216
      IF(IC.EQ.3)CALL CIRCL2(X1,Y1,Z1,X3,Y3,Z3,X4,Y4,Z4,IB,RADIUS)      SEQ20217
      IF(IA.EQ.0)GO TO 100                                              SEQ20218
      XX(IB(1),IB(2),IB(3))=X1                                          SEQ20219
      YY(IB(1),IB(2),IB(3))=Y1                                          SEQ20220
      ZZ(IB(1),IB(2),IB(3))=Z1                                          SEQ20221
      XX(IB(4),IB(5),IB(6))=X4                                          SEQ20222
      YY(IB(4),IB(5),IB(6))=Y4                                          SEQ20223
      ZZ(IB(4),IB(5),IB(6))=Z4                                          SEQ20224
```

```
      GO TO 100                                                          SEQ20225
  330 CONTINUE                                                           SEQ20226
C                                                                        SEQ20227
C   INDICES DEFINE A REGION                                              SEQ20228
C                                                                        SEQ20229
      IS1=IB(1)                                                          SEQ20230
      IS2=IB(2)                                                          SEQ20231
      IS3=IB(3)                                                          SEQ20232
      IS4=IB(4)                                                          SEQ20233
      IS5=IB(5)                                                          SEQ20234
      IS6=IB(6)                                                          SEQ20235
      MB1=I(IB(1))                                                       SEQ20236
      MB2=J(IB(2))                                                       SEQ20237
      MB3=K(IB(3))                                                       SEQ20238
      MB4=I(IB(4))                                                       SEQ20239
      MB5=J(IB(5))                                                       SEQ20240
      MB6=K(IB(6))                                                       SEQ20241
      CALL FN3                                                           SEQ20242
      IF(IA.EQ.1)GO TO 340                                               SEQ20243
      IF(IA.EQ.2)GO TO 360                                               SEQ20244
      IF(IA.EQ.3)GO TO 380                                               SEQ20245
  340 CONTINUE                                                           SEQ20246
      DO 350 IBA=IS1,IS4                                                 SEQ20247
      IBB=I(IBA)                                                         SEQ20248
      PX=0.25*(XX(IBB,MB2,MB3)+XX(IBB,MB5,MB3)+XX(IBB,MB2,MB6)           SEQ20249
     1+XX(IBB,MB5,MB6))                                                  SEQ20250
      PY=0.25*(YY(IBB,MB2,MB3)+YY(IBB,MB5,MB3)+YY(IBB,MB2,MB6)           SEQ20251
     1+YY(IBB,MB5,MB6))                                                  SEQ20252
      PZ=0.25*(ZZ(IBB,MB2,MB3)+ZZ(IBB,MB5,MB3)+ZZ(IBB,MB2,MB6)           SEQ20253
     1+ZZ(IBB,MB5,MB6))                                                  SEQ20254
      X1=XX(IBB,MB2,MB3)                                                 SEQ20255
      Y1=YY(IBB,MB2,MB3)                                                 SEQ20256
      Z1=ZZ(IBB,MB2,MB3)                                                 SEQ20257
      X2=XX(IBB,MB5,MB3)                                                 SEQ20258
      Y2=YY(IBB,MB5,MB3)                                                 SEQ20259
      Z2=ZZ(IBB,MB5,MB3)                                                 SEQ20260
      X3=XX(IBB,MB2,MB6)                                                 SEQ20261
      Y3=YY(IBB,MB2,MB6)                                                 SEQ20262
      Z3=ZZ(IBB,MB2,MB6)                                                 SEQ20263
      X4=XX(IBB,MB5,MB6)                                                 SEQ20264
      Y4=YY(IBB,MB5,MB6)                                                 SEQ20265
      Z4=ZZ(IBB,MB5,MB6)                                                 SEQ20266
      IB(1)=IBB                                                          SEQ20267
      IB(2)=MB2                                                          SEQ20268
      IB(3)=MB3                                                          SEQ20269
      IB(4)=IBB                                                          SEQ20270
      IB(5)=MB5                                                          SEQ20271
      IB(6)=MB3                                                          SEQ20272
      CALL CIRCL2(X1,Y1,Z1,PX,PY,PZ,X2,Y2,Z2,IB,RADIUS)                  SEQ20273
      XX(IB(1),IB(2),IB(3))=X1                                           SEQ20274
      YY(IB(1),IB(2),IB(3))=Y1                                           SEQ20275
      ZZ(IB(1),IB(2),IB(3))=Z1                                           SEQ20276
      XX(IB(4),IB(5),IB(6))=X2                                           SEQ20277
      YY(IB(4),IB(5),IB(6))=Y2                                           SEQ20278
      ZZ(IB(4),IB(5),IB(6))=Z2                                           SEQ20279
      IB(1)=IBB                                                          SEQ20280
      IB(2)=MB5                                                          SEQ20281
      IB(3)=MB6                                                          SEQ20282
      CALL CIRCL2(X2,Y2,Z2,PX,PY,PZ,X4,Y4,Z4,IB,RADIUS)                  SEQ20283
      XX(IB(1),IB(2),IB(3))=X4                                           SEQ20284
      YY(IB(1),IB(2),IB(3))=Y4                                           SEQ20285
```

```
      ZZ(IB(1),IB(2),IB(3))=Z4                                          SEQ20285
      IB(4)=IBB                                                         SEQ20287
      IB(5)=MB2                                                         SEQ20288
      IB(6)=MB6                                                         SEQ20289
      CALL CIRCL2(X4,Y4,Z4,PX,PY,PZ,X3,Y3,Z3,IB,RADIUS)                 SEQ20290
      XX(IB(4),IB(5),IB(6))=X3                                          SEQ20291
      YY(IB(4),IB(5),IB(6))=Y3                                          SEQ20292
      ZZ(IB(4),IB(5),IB(6))=Z3                                          SEQ20293
      IB(1)=IBB                                                         SEQ20294
      IB(2)=MB2                                                         SEQ20295
      IB(3)=MB3                                                         SEQ20296
      CALL CIRCL2(X3,Y3,Z3,PX,PY,PZ,X1,Y1,Z1,IB,RADIUS)                 SEQ20297
  350 CONTINUE                                                          SEQ20298
      GO TO 100                                                         SEQ20299
  360 CONTINUE                                                          SEQ20300
      DO 370 IBA=IS2,IS5                                                SEQ20301
      IBB=J(IBA)                                                        SEQ20302
      PX=0.25*(XX(MB1,IBB,MB3)+XX(MB4,IBB,MB3)+XX(MB1,IBB,MB6)          SEQ20303
     1+XX(MB4,IBB,MB6))                                                 SEQ20304
      PY=0.25*(YY(MB1,IBB,MB3)+YY(MB4,IBB,MB3)+YY(MB1,IBB,MB6)          SEQ20305
     1+YY(MB4,IBB,MB6))                                                 SEQ20306
      PZ=0.25*(ZZ(MB1,IBB,MB3)+ZZ(MB4,IBB,MB3)+ZZ(MB1,IBB,MB6)          SEQ20307
     1+ZZ(MB4,IBB,MB6))                                                 SEQ20308
      X1=XX(MB1,IBB,MB3)                                                SEQ20309
      Y1=YY(MB1,IBB,MB3)                                                SEQ20310
      Z1=ZZ(MB1,IBB,MB3)                                                SEQ20311
      X2=XX(MB4,IBB,MB3)                                                SEQ20312
      Y2=YY(MB4,IBB,MB3)                                                SEQ20313
      Z2=ZZ(MB4,IBB,MB3)                                                SEQ20314
      X3=XX(MB1,IBB,MB6)                                                SEQ20315
      Y3=YY(MB1,IBB,MB6)                                                SEQ20316
      Z3=ZZ(MB1,IBB,MB6)                                                SEQ20317
      X4=XX(MB4,IBB,MB6)                                                SEQ20318
      Y4=YY(MB4,IBB,MB6)                                                SEQ20319
      Z4=ZZ(MB4,IBB,MB6)                                                SEQ20320
      IB(1)=MB1                                                         SEQ20321
      IB(2)=IB6                                                         SEQ20322
      IB(3)=MB3                                                         SEQ20323
      IB(4)=MB4                                                         SEQ20324
      IB(5)=IBB                                                         SEQ20325
      IB(6)=MB2                                                         SEQ20326
      CALL CIRCL2(X1,Y1,Z1,PX,PY,PZ,X2,Y2,Z2,IB,RADIUS)                 SEQ20327
      XX(IB(1),IB(2),IB(3))=X1                                          SEQ20328
      YY(IB(1),IB(2),IB(3))=Y1                                          SEQ20329
      ZZ(IB(1),IB(2),IB(3))=Z1                                          SEQ20330
      XX(IB(4),IB(5),IB(6))=X2                                          SEQ20331
      YY(IB(4),IB(5),IB(6))=Y2                                          SEQ20332
      ZZ(IB(4),IB(5),IB(6))=Z2                                          SEQ20333
      IB(1)=MB4                                                         SEQ20334
      IB(2)=IBB                                                         SEQ20335
      IB(3)=MB6                                                         SEQ20336
      CALL CIRCL2(X2,Y2,Z2,PX,PY,PZ,X4,Y4,Z4,IB,RADIUS)                 SEQ20337
      XX(IB(1),IB(2),IB(3))=X4                                          SEQ20338
      YY(IB(1),IB(2),IB(3))=Y4                                          SEQ20339
      ZZ(IB(1),IB(2),IB(3))=Z4                                          SEQ20340
      IB(4)=MB1                                                         SEQ20341
      IB(5)=IBB                                                         SEQ20342
      IB(6)=MB6                                                         SEQ20343
      CALL CIRCL2(X4,Y4,Z4,PX,PY,PZ,X3,Y3,Z3,IB,RADIUS)                 SEQ20344
      XX(IB(4),IB(5),IB(6))=X3                                          SEQ20345
      YY(IB(4),IB(5),IB(6))=Y3                                          SEQ20346
```

```
      ZZ(IB(4),IB(5),IB(6))=Z3                                          SEQ20347
      IB(1)=MB1                                                         SEQ20348
      IB(2)=IBB                                                         SEQ20349
      IB(3)=MB3                                                         SEQ20350
      CALL CIRCL2(X3,Y3,Z3,PX,PY,PZ,X1,Y1,Z1,IB,RADIUS)                 SEQ20351
  370 CONTINUE                                                          SEQ20352
      GO TO 100                                                         SEQ20353
  380 CONTINUE                                                          SEQ20354
      DO 390 IBA=IS3,IS6                                                SEQ20355
      IBB=K(IBA)                                                        SEQ20356
      PX=0.25*(XX(MB1,MB2,IBB)+XX(MB4,MB2,IBB)+XX(MB1,MB5,IBB)          SEQ20357
     1+XX(MB4,MB5,IBB))                                                 SEQ20358
      PY=0.25*(YY(MB1,MB2,IBB)+YY(MB4,MB2,IBB)+YY(MB1,MB5,IBB)          SEQ20359
     1+YY(MB4,MB5,IBB))                                                 SEQ20360
      PZ=0.25*(ZZ(MB1,MB2,IBB)+ZZ(MB4,MB2,IBB)+ZZ(MB1,MB5,IBB)          SEQ20361
     1+ZZ(MB4,MB5,IBB))                                                 SEQ20362
      X1=XX(MB1,MB2,IBB)                                                SEQ20363
      Y1=YY(MB1,MB2,IBB)                                                SEQ20364
      Z1=ZZ(MB1,MB2,IBB)                                                SEQ20365
      X2=XX(MB4,MB2,IBB)                                                SEQ20366
      Y2=YY(MB4,MB2,IBB)                                                SEQ20367
      Z2=ZZ(MB4,MB2,IBB)                                                SEQ20368
      X3=XX(MB1,MB5,IBB)                                                SEQ20369
      Y3=YY(MB1,MB5,IBB)                                                SEQ20370
      Z3=ZZ(MB1,MB5,IBB)                                                SEQ20371
      X4=XX(MB4,MB5,IBB)                                                SEQ20372
      Y4=YY(MB4,MB5,IBB)                                                SEQ20373
      Z4=ZZ(MB4,MB5,IBB)                                                SEQ20374
      IB(1)=MB1                                                         SEQ20375
      IB(2)=MB2                                                         SEQ20376
      IB(3)=IBB                                                         SEQ20377
      IB(4)=MB4                                                         SEQ20378
      IB(5)=MB2                                                         SEQ20379
      IB(6)=IBB                                                         SEQ20380
      CALL CIRCL2(X1,Y1,Z1,PX,PY,PZ,X2,Y2,Z2,IB,RADIUS)                 SEQ20381
      XX(IB(1),IB(2),IB(3))=X1                                          SEQ20382
      YY(IB(1),IB(2),IB(3))=Y1                                          SEQ20383
      ZZ(IB(1),IB(2),IB(3))=Z1                                          SEQ20384
      XX(IB(4),IB(5),IB(6))=X2                                          SEQ20385
      YY(IB(4),IB(5),IB(6))=Y2                                          SEQ20386
      ZZ(IB(4),IB(5),IB(6))=Z2                                          SEQ20387
      IB(1)=MB4                                                         SEQ20388
      IB(2)=MB5                                                         SEQ20389
      IB(3)=IBB                                                         SEQ20390
      CALL CIRCL2(X2,Y2,Z2,PX,PY,PZ,X4,Y4,Z4,IB,RADIUS)                 SEQ20391
      XX(IB(1),IB(2),IB(3))=X4                                          SEQ20392
      YY(IB(1),IB(2),IB(3))=Y4                                          SEQ20393
      ZZ(IB(1),IB(2),IB(3))=Z4                                          SEQ20394
      IB(4)=MB1                                                         SEQ20395
      IB(5)=MB5                                                         SEQ20396
      IB(6)=IBB                                                         SEQ20397
      CALL CIRCL2(X4,Y4,Z4,PX,PY,PZ,X3,Y3,Z3,IB,RADIUS)                 SEQ20398
      XX(IB(4),IB(5),IB(6))=X3                                          SEQ20399
      YY(IB(4),IB(5),IB(6))=Y3                                          SEQ20400
      ZZ(IB(4),IB(5),IB(6))=Z3                                          SEQ20401
      IB(1)=MB1                                                         SEQ20402
      IB(2)=MB2                                                         SEQ20403
      IB(3)=IBB                                                         SEQ20404
      CALL CIRCL2(X3,Y3,Z3,PX,PY,PZ,X1,Y1,Z1,IB,RADIUS)                 SEQ20405
  390 CONTINUE                                                          SEQ20406
      GO TO 100                                                         SEQ20407
```

```
  400 CONTINUE                                                          SEQ20408
C                                                                       SEQ20409
C  BOUNDARY CONDITION CARD                                              SEQ20410
C                                                                       SEQ20411
      BACKSPACE 11                                                      SEQ20412
      READ(11,670)IB,IBC,IG                                             SEQ20413
      IF(IG.EQ.1)GO TO 410                                              SEQ20414
      CALL FN3                                                          SEQ20415
  410 CONTINUE                                                          SEQ20416
      CALL BCS(IB,IBC)                                                  SEQ20417
      GO TO 100                                                         SEQ20418
  420 CONTINUE                                                          SEQ20419
C                                                                       SEQ20420
C  LOAD CARD                                                            SEQ20421
C                                                                       SEQ20422
      BACKSPACE 11                                                      SEQ20423
      READ(11,680)IB,IG,LC,FX,FY,FZ                                     SEQ20424
      IF(IG.EQ.1)GO TO 430                                              SEQ20425
      CALL FN3                                                          SEQ20426
  430 CONTINUE                                                          SEQ20427
      CALL LOADS(IB,LC,FX,FY,FZ)                                        SEQ20428
      GO TO 100                                                         SEQ20429
  440 CONTINUE                                                          SEQ20430
      BACKSPACE 11                                                      SEQ20431
      READ(11,690)IB,IG                                                 SEQ20432
      IF(IG.EQ.1)GO TO 450                                              SEQ20433
      CALL FN3                                                          SEQ20434
  450 CONTINUE                                                          SEQ20435
      CALL DELETE(IB)                                                   SEQ20436
      GO TO 100                                                         SEQ20437
  460 CONTINUE                                                          SEQ20438
C                                                                       SEQ20439
C  MIDSIDE NODE DELETION CARD                                           SEQ20440
C                                                                       SEQ20441
      BACKSPACE 11                                                      SEQ20442
      READ(11,700)IB,IX,IY,IZ,IG                                        SEQ20443
      IF(IG.EQ.1)GO TO 470                                              SEQ20444
      CALL FN3                                                          SEQ20445
  470 CONTINUE                                                          SEQ20446
      CALL MIDSID(IB,IX,IY,IZ)                                          SEQ20447
      GO TO 100                                                         SEQ20448
  480 CONTINUE                                                          SEQ20449
C                                                                       SEQ20450
C  EQUIVALENT NODES                                                     SEQ20451
C                                                                       SEQ20452
      BACKSPACE 11                                                      SEQ20453
      READ(11,690)I1,J1,K1,I2,J2,K2                                     SEQ20454
      I1=I(I1)                                                          SEQ20455
      I2=I(I2)                                                          SEQ20456
      J1=J(J1)                                                          SEQ20457
      J2=J(J2)                                                          SEQ20458
      K1=K(K1)                                                          SEQ20459
      K2=K(K2)                                                          SEQ20460
      XX(I1,J1,K1)=XX(I2,J2,K2)                                         SEQ20461
      YY(I1,J1,K1)=YY(I2,J2,K2)                                         SEQ20462
      ZZ(I1,J1,K1)=ZZ(I2,J2,K2)                                         SEQ20463
      GO TO 100                                                         SEQ20464
  490 CONTINUE                                                          SEQ20465
C                                                                       SEQ20466
C  REGION COMMAND                                                       SEQ20467
C                                                                       SEQ20468
```

```
      BACKSPACE 11                                                      SEQ20469
      READ(11,690)I1,J1,K1,I2,J2,K2                                     SEQ20470
      I3=I(I1)                                                          SEQ20471
      I4=I(I2)                                                          SEQ20472
      J3=J(J1)                                                          SEQ20473
      J4=J(J2)                                                          SEQ20474
      K3=K(K1)                                                          SEQ20475
      K4=K(K2)                                                          SEQ20476
      NC=0                                                              SEQ20477
      IF(I1.NE.I2)NC=NC+1                                               SEQ20478
      IF(J1.NE.J2)NC=NC+1                                               SEQ20479
      IF(K1.NE.K2)NC=NC+1                                               SEQ20480
      IF(NC.EQ.3)GO TO 520                                              SEQ20481
      IF(NC.EQ.2)GO TO 510                                              SEQ20482
C                                                                       SEQ20483
C       LINE                                                            SEQ20484
C                                                                       SEQ20485
      IW=I2-I1                                                          SEQ20486
      JW=J2-J1                                                          SEQ20487
      KW=K2-K1                                                          SEQ20488
      M=IW+JW+KW-1                                                      SEQ20489
      IF(IW.NE.0)IW=1                                                   SEQ20490
      IF(JW.NE.0)JW=1                                                   SEQ20491
      IF(KW.NE.0)KW=1                                                   SEQ20492
      X3=XX(I3,J3,K3)                                                   SEQ20493
      Y3=YY(I3,J3,K3)                                                   SEQ20494
      Z3=ZZ(I3,J3,K3)                                                   SEQ20495
      X4=XX(I4,J4,K4)                                                   SEQ20496
      Y4=YY(I4,J4,K4)                                                   SEQ20497
      Z4=ZZ(I4,J4,K4)                                                   SEQ20498
      VX=X4-X3                                                          SEQ20499
      VY=Y4-Y3                                                          SEQ20500
      VZ=Z4-Z3                                                          SEQ20501
      AMD1=I4+J4+K4-I3-J3-K3                                            SEQ20502
      DO 500 M1=1,M                                                     SEQ20503
      IX=I1+IW*M1                                                       SEQ20504
      JX=J1+JW*M1                                                       SEQ20505
      KX=K1+KW*M1                                                       SEQ20506
      IX=I(IX)                                                          SEQ20507
      JX=J(JX)                                                          SEQ20508
      KX=K(KX)                                                          SEQ20509
      AMD=IX+JX+KX-I3-J3-K3                                             SEQ20510
      FAC=AMD/AMD1                                                      SEQ20511
      XX(IX,JX,KX)=X3+FAC*VX                                            SEQ20512
      YY(IX,JX,KX)=Y3+FAC*VY                                            SEQ20513
      ZZ(IX,JX,KX)=Z3+FAC*VZ                                            SEQ20514
  500 CONTINUE                                                          SEQ20515
      GO TO 100                                                         SEQ20516
  510 CONTINUE                                                          SEQ20517
  520 CONTINUE                                                          SEQ20518
C                                                                       SEQ20519
C       PLANE AND VOLUME GENERATION                                     SEQ20520
C                                                                       SEQ20521
      PX=1.0                                                            SEQ20522
      PY=1.0                                                            SEQ20523
      PZ=1.0                                                            SEQ20524
      MI=I4-I3-1                                                        SEQ20525
      MJ=J4-J3-1                                                        SEQ20526
      MK=K4-K3-1                                                        SEQ20527
      IF(MI.LT.0)PX=0.0                                                 SEQ20528
      IF(MJ.LT.0)PY=0.0                                                 SEQ20529
```

```
      IF(MK.LT.0)PZ=0.0                                                 SEQ20530
        IF(MI.LT.1)MI=1                                                 SEQ20531
        IF(MJ.LT.1)MJ=1                                                 SEQ20532
        IF(MK.LT.1)MK=1                                                 SEQ20533
      DO 550 M1=1,MI                                                    SEQ20534
      DO 540 M2=1,MJ                                                    SEQ20535
      DO 530 M3=1,MK                                                    SEQ20536
      IX=M1+I1                                                          SEQ20537
      JX=M2+J1                                                          SEQ20538
      KX=M3+K1                                                          SEQ20539
        IF(I2.EQ.I1)IX=I1                                               SEQ20540
        IF(J2.EQ.J1)JX=J1                                               SEQ20541
        IF(K2.EQ.K1)KX=K1                                               SEQ20542
      IX=I(IX)                                                          SEQ20543
      JX=J(JX)                                                          SEQ20544
      KX=K(KX)                                                          SEQ20545
      XX(IX,JX,KX)=(PY*XX(IX,J3,KX)+PZ*XX(IX,JX,K3)+PX*XX(I4,JX,KX)+   SEQ20546
     1PX*XX(I3,JX,KX)+PZ*XX(IX,JX,K4)+PY*XX(IX,J4,KX))/6.0              SEQ20547
      YY(IX,JX,KX)=(PY*YY(IX,J3,KX)+PZ*YY(IX,JX,K3)+PX*YY(I4,JX,KX)+   SEQ20548
     1PX*YY(I3,JX,KX)+PZ*YY(IX,JX,K4)+PY*YY(IX,J4,KX))/6.0              SEQ20549
      ZZ(IX,JX,KX)=(PY*ZZ(IX,J3,KX)+PZ*ZZ(IX,JX,K3)+PX*ZZ(I4,JX,KX)+   SEQ20550
     1PX*ZZ(I3,JX,KX)+PZ*ZZ(IX,JX,K4)+PY*ZZ(IX,J4,KX))/6.0              SEQ20551
  530 CONTINUE                                                          SEQ20552
  540 CONTINUE                                                          SEQ20553
  550 CONTINUE                                                          SEQ20554
      GO TO 100                                                         SEQ20555
  560 CONTINUE                                                          SEQ20556
      IF(IC2.LE.2)RETURN                                                SEQ20557
C                                                                       SEQ20558
C  PERFORM A COORDINATE TRANSFORMATION                                  SEQ20559
C                                                                       SEQ20560
      DO 590 M1=1,MI                                                    SEQ20561
      DO 580 M2=1,MJ                                                    SEQ20562
      DO 570 M3=1,MK                                                    SEQ20563
      IF(I(M1).EQ.0.OR.J(M2).EQ.0.OR.K(M3).EQ.0)GO TO 570               SEQ20564
      IF(IC2.EQ.3)CALL CYLIND(XX(I(M1),J(M2),K(M3)),YY(I(M1),J(M2),K(M3SEQ20565
     1),ZZ(I(M1),J(M2),K(M3)))                                          SEQ20566
      IF(IC2.EQ.4)CALL SPHERI(XX(I(M1),J(M2),K(M3)),YY(I(M1),J(M2),K(M3)SEQ20567
     1),ZZ(I(M1),J(M2),K(M3)))                                          SEQ20568
  570 CONTINUE                                                          SEQ20569
  580 CONTINUE                                                          SEQ20570
  590 CONTINUE                                                          SEQ20571
  600 RETURN                                                            SEQ20572
  610 FORMAT(A1,4X,10X,I5)                                              SEQ20573
  620 FORMAT(16I5)                                                      SEQ20574
  630 FORMAT(8F10.0)                                                    SEQ20575
  640 FORMAT(A1,I4,3I5,6F10.0)                                          SEQ20576
  650 FORMAT(1X,A1,3X,4I5,F10.4,I5)                                     SEQ20577
  660 FORMAT(2X,7I2,4X,4F10.0)                                          SEQ20578
  670 FORMAT(2X,13I2)                                                   SEQ20579
  680 FORMAT(2X,6I2,11,I5,3F10.0)                                       SEQ20580
  690 FORMAT(2X,7I2)                                                    SEQ20581
  700 FORMAT(2X,10I2)                                                   SEQ20582
      END                                                               SEQ20583
      SUBROUTINE SPHERI(R,THETA,PHI)                                    SPHR0001
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCSPHR0002
C                                                                      CSPHR0003
C  SUBROUTINE SPHERI CONVERTS FROM SPHERICAL COORDINATES TO            CSPHR0004
C  CYLINDRICAL COORDINATES.                                            CSPHR0005
C                                                                      CSPHR0006
C  CALLED BY:  CTRAN   ELTHRE  ELTWO   SEQ2                            CSPHR0007
```

```
C                                                                       CSPHR00C8
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCSPHRCCC9
      DATA ST/0.C/,CT/1.0/                                              SPHR0010
      PSP=R*SIN(PHI/57.2958)                                            SPHRC011
      IF(THETA.EC.THETAD)GO TO 10                                       SPHR0012
      CT=CCS(THETA/57.2958)                                             SPHR0013
      ST=SIN(THETA/57.2958)                                             SPHRC014
   10 CONTINUE                                                          SPHR0015
      X=RSP*CT                                                          SPHR0016
      Y=RSP*ST                                                          SPHR0017
      Z=R*CCS(PHI/57.2958)                                              SPHRC018
      R=X                                                               SPHRC019
      THETA=Y                                                           SPHRC020
      PHI=Z                                                             SPHRC021
      THETAD=THETA                                                      SPHR0022
      RETURN                                                            SPHRC023
      END                                                               SPHR0024
      SUBRCUTINE TRANS(IEND,MAXN)                                       TRNS0001
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCTRNSCCC2
C                                                                       CTRNS0003
C  SUBRCUTINE TRANS TRANSFERS DATA FROM UNIT 12 TO UNIT 11.  CONTROL    CTRNSCCC4
C  CARDS ARE PROCESSED AND DELETED .                                    CTRNS0005
C                                                                       CTRNS0006
C  CALLED BY:  DLEE                                                     CTRNSCCC7
C                                                                       CTRNSCCC8
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCTRNSCCC9
      DIMENSION D(21),CHAR(10)                                          TRNS0010
      COMMCN/REPEAT/NR,NRT(50)                                          TRNS0011
      COMMCN/PROP/MAT,IDOT,IC2,IT,TH,N8,N20                             TRNSC012
      DATA CHAR/4HSTAR,4HEND-,4HENC ,4H8NOD,4H2ONO,4HCYLI,4HSPHE,       TRNSC013
     14HMATE,4HTHIC,4HREPE/,CCMAT/1HC/                                  TRNSC014
      MAT=1                                                             TRNS0015
      IDOT=0                                                            TRNSC016
      IC2=0                                                             TRNS0017
      IT=1                                                              TRNS0018
      N8=0                                                              TRNS0019
      N20=0                                                             TRNSC020
      MAXN=1                                                            TRNSCC21
      IEND=0                                                            TRNSC022
      NR=0                                                              TRNSC023
      NRT(1)=0                                                          TRNSC024
C                                                                       TRNSCC25
C  CHECK FCR CCNTRCL CARDS                                              TRNSCC26
C                                                                       TRNSCC27
   10 CONTINUE                                                          TRNSCC28
      READ(12,13C,END=120)D(1)                                          TRNS0029
C                                                                       TRNSC030
C  COMMAND = 'START'                                                    TRNS0031
C                                                                       TRNS0032
      IF(D(1).NE.CHAR(1))GO TO 20                                       TRNSCC33
      WRITE(11,140)                                                     TRNSC034
      GO TO 10                                                          TRNS0035
C                                                                       TRNSC036
C  COMMAND = 'END-'                                                     TRNSC037
C                                                                       TRNSC038
   20 IF(D(1).NE.CHAR(2))GO TC 3C                                       TRNS0039
      WRITE(11,150)                                                     TRNS004C
      RETURN                                                            TRNS0041
C                                                                       TRNS0042
C  COMMAND = 'ENC '                                                     TRNSC043
C                                                                       TRNS0044
```

```
      30 IF(D(1).NE.CHAR(3))GO TO 40                                    TRNS0045
         WRITE(11,150)                                                  TRNS0046
         RETURN                                                         TRNS0047
C                                                                       TRNS0048
C  COMMAND = '8NODE'                                                    TRNS0049
C                                                                       TRNS0050
      40 IF(D(1).NE.CHAR(4))GO TO 50                                    TRNS0051
         N8=1                                                           TRNS0052
         N20=1                                                          TRNS0053
         GO TO 10                                                       TRNS0054
C                                                                       TRNS0055
C  COMMAND = '20NODE'                                                   TRNS0056
C                                                                       TRNS0057
      50 IF(D(1).NE.CHAR(5))GO TO 60                                    TRNS0058
         N20=1                                                          TRNS0059
         N8=1                                                           TRNS0060
         GO TO 10                                                       TRNS0061
C                                                                       TRNS0062
C  COMMAND = 'CYLINDRICAL'                                              TRNS0063
C                                                                       TRNS0064
      60 IF(D(1).NE.CHAR(6))GO TO 70                                    TRNS0065
         IC2=1                                                          TRNS0066
         BACKSPACE 12                                                   TRNS0067
         READ(12,160)ID                                                 TRNS0068
         IF(ID.EQ.1)IC2=3                                               TRNS0069
         GO TO 10                                                       TRNS0070
C                                                                       TRNS0071
C  COMMAND = 'SPHERICAL'                                                TRNS0072
C                                                                       TRNS0073
      70 IF(D(1).NE.CHAR(7))GO TO 80                                    TRNS0074
         IC2=2                                                          TRNS0075
         BACKSPACE 12                                                   TRNS0076
         READ(12,160)ID                                                 TRNS0077
         IF(ID.EQ.1)IC2=4                                               TRNS0078
         GO TO 10                                                       TRNS0079
C                                                                       TRNS0080
C  COMMAND 'MATERIAL'                                                   TRNS0081
C                                                                       TRNS0082
      80 IF(D(1).NE.CHAR(8))GO TO 90                                    TRNS0083
         BACKSPACE 12                                                   TRNS0084
         READ(12,160)MAT                                                TRNS0085
         GO TO 10                                                       TRNS0086
C                                                                       TRNS0087
C  COMMAND = 'THICKNESS'                                                TRNS0088
C                                                                       TRNS0089
      90 IF(D(1).NE.CHAR(9))GO TO 100                                   TRNS0090
         BACKSPACE 12                                                   TRNS0091
         READ(12,180)TH,IT                                              TRNS0092
         GO TO 10                                                       TRNS0093
C                                                                       TRNS0094
C  COMMAND = 'REPEAT'                                                   TRNS0095
C                                                                       TRNS0096
     100 IF(D(1).NE.CHAR(10))GO TO 110                                  TRNS0097
         NR=NR+1                                                        TRNS0098
         BACKSPACE 12                                                   TRNS0099
         READ(12,160)NRT(NR)                                            TRNS0100
         MAXN=NR                                                        TRNS0101
         GO TO 10                                                       TRNS0102
C                                                                       TRNS0103
C  CHECK FOR COMMENTS                                                   TRNS0104
C                                                                       TRNS0105
```

```
  110 BACKSPACE 12                                                        TRNS01C6
      READ(12,170)D                                                       TRNS01C7
      IF(D(1).EQ.COMNT)GO TO 1C                                           TRNSC1C8
      WRITE(11,170)D                                                      TRNS01C9
      GO TO 10                                                            TRNS0110
  120 CONTINUE                                                            TRNS0111
      IEND=1                                                              TRNS0112
      RETURN                                                              TRNS0113
  130 FORMAT(A4)                                                          TRNS0114
  140 FORMAT('S')                                                         TRNS0115
  150 FORMAT('E')                                                         TRNS0116
  160 FORMAT(10X,I5)                                                      TRNS0117
  170 FORMAT(A1,A3,19A4)                                                  TRNS0118
  180 FORMAT(10X,F10.0,I5)                                                TRNS0119
      END                                                                 TRNSC12C
      SUBROUTINE VOLUME(I,IOUT)                                           VLME00C1
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCVLME00C2
C                                                                         CVLME0003
C  SUBROUTINE VOLUME CHECKS FOR ELEMENTS WHICH HAVE A NEGATIVE VOLUME     CVLME00C4
C                                                                         CVLME00C5
C  CALLED BY:  EL3C                                                       CVLME00C6
C                                                                         CVLME00C7
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCVLME00C8
      DIMENSION I(20),J(20)                                               VLME00C9
      COMMON/SYSARY/X(8000),Y(8000),Z(8000),N(8000)                       VLME0010
     1,I88(8000),IFLAG(8000),V(8000,3)                                    VLME0011
      XV1=X(I(2))-X(I(1))                                                 VLME0012
      YV1=Y(I(2))-Y(I(1))                                                 VLME0013
      ZV1=Z(I(2))-Z(I(1))                                                 VLME0014
      XV2=X(I(4))-X(I(1))                                                 VLME0015
      YV2=Y(I(4))-Y(I(1))                                                 VLME0016
      ZV2=Z(I(4))-Z(I(1))                                                 VLME0017
      XV3=X(I(5))-X(I(1))                                                 VLME0018
      YV3=Y(I(5))-Y(I(1))                                                 VLME0019
      ZV3=Z(I(5))-Z(I(1))                                                 VLME0020
      VOLM=XV1*YV2*ZV3+YV1*ZV2*XV3+ZV1*XV2*YV3-XV3*YV2*ZV1-               VLME0021
     1YV3*ZV2*XV1-ZV3*XV2*YV1                                             VLME0022
      IF(IOUT.EQ.3)VOLM=-VOLM                                             VLME0023
      IF(IOUT.EQ.2)VOLM=-VOLM                                             VLME0024
      IF(VOLM.LT.0.0)RETURN                                               VLME0025
C                                                                         VLME0026
C  VOLUME IS NEGATIVE,  RENUMBER AND RETURN                               VLME0027
C                                                                         VLME0028
      DO 10 K=1,20                                                        VLME0029
   10 J(K)=I(K)                                                           VLME0030
      I(1)=J(5)                                                           VLME0031
      I(2)=J(6)                                                           VLME0032
      I(3)=J(7)                                                           VLME0033
      I(4)=J(8)                                                           VLME0034
      I(5)=J(1)                                                           VLME0035
      I(6)=J(2)                                                           VLME0036
      I(7)=J(3)                                                           VLME0037
      I(8)=J(4)                                                           VLME0038
      I(9)=J(13)                                                          VLME0039
      I(10)=J(14)                                                         VLME04C0
      I(11)=J(15)                                                         VLME0041
      I(12)=J(16)                                                         VLME0042
      I(13)=J(9)                                                          VLME0043
      I(14)=J(10)                                                         VLME0044
      I(15)=J(11)                                                         VLME0045
      I(16)=J(12)                                                         VLME0046
```

```
      RETURN                                                            VLMEC047
      END                                                               VLMEC048
      SUBROUTINE ZERO(KLIM,NLIM,C1,N)                                   ZERC0001
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCZERC0002
C                                                                       CZERC0003
C  SUBROUTINE ZERO SETS THE SYSTEM MATRIX TC ZERC.                      CZERC0004
C                                                                       CZERC0005
C  CALLED BY:  DLEE                                                     CZERC0006
C                                                                       CZERC0007
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCZERC0008
      DIMENSION C1(KLIM),N(NLIM)                                        ZERC0009
      DO 10 I=1,NLIM                                                    ZERC0010
      N(I)=0                                                            ZERC0011
   10 CONTINUE                                                          ZERC0012
      RETURN                                                            ZERC0013
      END                                                               ZERC0014
```

2. Fortran Listing of Program MOVE

```
C                                                                       MAIN0001
C     PROGRAM MOVE REFORMATS DATA FOR MOVIE.BYU AND REMOVES HIDDEN      MAIN0002
C     SURFACES.                                                         MAIN0003
C                                                                       MAIN0004
      COMMON/ARRAY/IAR(12000,4),IF(12000)                               MAIN0005
      READ(5,100)N2D,N3D,N1D,NCDE,N8ND                                  MAIN0006
      N=0                                                               MAIN0007
      IF(N1D.EQ.0)GO TO 20                                              MAIN0008
C                                                                       MAIN0009
C  PROCESS BEAM ELEMENTS                                                MAIN0010
C                                                                       MAIN0011
      DO 10 I=1,N1D                                                     MAIN0012
      READ(21,110)I1,I2                                                 MAIN0013
      N=N+1                                                             MAIN0014
      IAR(N,1)=I1                                                       MAIN0015
      IAR(N,2)=I2                                                       MAIN0016
      IAR(N,3)=0                                                        MAIN0017
      IAR(N,4)=0                                                        MAIN0018
   10 CONTINUE                                                          MAIN0019
   20 IF(N2D.EQ.0)GO TO 40                                              MAIN0020
C                                                                       MAIN0021
C  PROCESS PLATE ELEMENTS                                               MAIN0022
C                                                                       MAIN0023
      DO 30 I=1,N2D                                                     MAIN0024
      READ(21)I1,I2,I3,I4                                               MAIN0025
      N=N+1                                                             MAIN0026
      IAR(N,1)=I1                                                       MAIN0027
      IAR(N,2)=I2                                                       MAIN0028
      IAR(N,3)=I3                                                       MAIN0029
      IAR(N,4)=I4                                                       MAIN0030
   30 CONTINUE                                                          MAIN0031
   40 IF(N3D.EQ.0)GO TO 50                                              MAIN0032
C                                                                       MAIN0033
C  PROCESS SOLID ELEMENTS                                               MAIN0034
C                                                                       MAIN0035
      NMIN=N+1                                                          MAIN0036
      CALL SOLID(N8D,N3D,NMIN)                                          MAIN0037
   50 CONTINUE                                                          MAIN0038
      NEL=N1D+N2D+N3D                                                   MAIN0039
      NMAX=N1D+N2D+N3D                                                  MAIN0040
      NODES=NODE-NEND                                                   MAIN0041
      NPARTS=1                                                          MAIN0042
C                                                                       MAIN0043
C  WRITE CONTROL INFORMATION                                            MAIN0044
C                                                                       MAIN0045
      WRITE(10,140)NPARTS,NODES,NEL                                     MAIN0046
      WRITE(10,140)NPARTS,NEL                                           MAIN0047
      N=0                                                               MAIN0048
   60 CONTINUE                                                          MAIN0049
C                                                                       MAIN0050
C  PROCESS NODE POINT DATA                                              MAIN0051
C                                                                       MAIN0052
      READ(21,END=80)X1,Y1,Z1                                           MAIN0053
      N=N+1                                                             MAIN0054
      IF(N.GT.NODES)GO TO 80                                            MAIN0055
      READ(21,END=70)X2,Y2,Z2                                           MAIN0056
      N=N+1                                                             MAIN0057
   70 WRITE(10,160)X1,Y1,Z1,X2,Y2,Z2                                    MAIN0058
      GO TO 60                                                          MAIN0059
C                                                                       MAIN0060
```

```
C  OUTPUT THE ELEMENTS                                                    MAIN0061
C                                                                         MAIN0062
      DO 80 N=1,NMAX                                                      MAIN0063
      IF(IF(I).EQ.1)GO TO 80                                              MAIN0064
      WRITE(10,130)(IAR(I,J),J=1,4)                                       MAIN0065
   80 CONTINUE                                                            MAIN0066
   90 CONTINUE                                                            MAIN0067
      STOP                                                                MAIN0068
  100 FORMAT(///,16X,I5,/,16X,I5,/,9X,I5,//,21X,I5,                       MAIN0069
     1/,31X,I5)                                                           MAIN0070
  110 FORMAT(5X,2I5)                                                      MAIN0071
  120 FORMAT(5X,4I5)                                                      MAIN0072
  130 FORMAT(4I5)                                                         MAIN0073
  140 FORMAT(3I5)                                                         MAIN0074
  150 FORMAT(I10,3F10.0)                                                  MAIN0075
  160 FORMAT(6F12.5)                                                      MAIN0076
      END                                                                 MAIN0077
      SUBROUTINE SCLID(NNMAX,N3D,NMIN)                                    SLID0001
C                                                                         SLID0002
C  SUBROUTINE SCLID PROCESSES SOLID INFORMATION AND REMOVES HIDDEN        SLID0003
C  SURFACES WHICH OCCUR WHEN TWO SOLID ELEMENTS ARE ADJACENT.             SLID0004
C                                                                         SLID0005
      DIMENSION IM(8)                                                     SLID0006
      COMMON I(12000,4),IF(12000)                                         SLID0007
      EQUIVALENCE (IM(1),I1),(IM(2),I2),(IM(3),I3),(IM(4),I4),(IM(5),I5)SLID0008
     1,(IM(6),I6),(IM(7),I7),(IM(8),I8)                                   SLID0009
      N3D=N3D*6                                                           SLID0010
      N=NMIN                                                              SLID0011
      ND=1                                                                SLID0012
   10 CONTINUE                                                            SLID0013
C                                                                         SLID0014
C  INPUT ELEMENTS                                                         SLID0015
C                                                                         SLID0016
      IF(ND.GE.N3D)GO TO 80                                               SLID0017
      READ(21,END=80)NDEL,IM                                              SLID0018
C                                                                         SLID0019
C  NDEL CONTAINS INFORMATION FOR DELETING ELEMENT SURFACES                SLID0020
C                                                                         SLID0021
      CALL DIVIDE(NDEL,ID1,ID2,ID3,ID4,ID5,ID6)                          SLID0022
      IF(ID6.EQ.1)GO TO 20                                                SLID0023
      I(N,1)=I1                                                           SLID0024
      I(N,2)=I2                                                           SLID0025
      I(N,3)=I3                                                           SLID0026
      I(N,4)=I4                                                           SLID0027
      N=N+1                                                               SLID0028
   20 IF(ID5.EQ.1)GO TO 30                                                SLID0029
      I(N,1)=I5                                                           SLID0030
      I(N,2)=I6                                                           SLID0031
      I(N,3)=I7                                                           SLID0032
      I(N,4)=I8                                                           SLID0033
      N=N+1                                                               SLID0034
   30 IF(ID1.EQ.1)GO TO 40                                                SLID0035
      I(N,1)=I1                                                           SLID0036
      I(N,2)=I2                                                           SLID0037
      I(N,3)=I6                                                           SLID0038
      I(N,4)=I5                                                           SLID0039
      N=N+1                                                               SLID0040
   40 IF(ID2.EQ.1)GO TO 50                                                SLID0041
      I(N,1)=I4                                                           SLID0042
      I(N,2)=I3                                                           SLID0043
      I(N,3)=I7                                                           SLID0044
```

```
      I(N,4)=I8                                                        SLID0045
      N=N+1                                                            SLID0046
   50 IF(ID4.EQ.1)GO TO 60                                             SLID0047
      I(N,1)=I2                                                        SLID0048
      I(N,2)=I3                                                        SLID0049
      I(N,3)=I7                                                        SLID0050
      I(N,4)=I6                                                        SLID0051
      N=N+1                                                            SLID0052
   60 IF(ID3.EQ.1)GO TO 70                                             SLID0053
      I(N,1)=I1                                                        SLID0054
      I(N,2)=I4                                                        SLID0055
      I(N,3)=I8                                                        SLID0056
      I(N,4)=I5                                                        SLID0057
      N=N+1                                                            SLID0058
   70 CONTINUE                                                         SLID0059
      ND=ND+6                                                          SLID0060
      GO TO 10                                                         SLID0061
   80 CONTINUE                                                         SLID0062
      NMAX=N-1                                                         SLID0063
C                                                                      SLID0064
C  ELEMENTS FACES ARE GIVEN A UNIQUE NUMBERING ORDER.                  SLID0065
C                                                                      SLID0066
      DO 140 N=NMIN,NMAX                                               SLID0067
      I1=I(N,1)                                                        SLID0068
      I2=I(N,2)                                                        SLID0069
      I3=I(N,3)                                                        SLID0070
      I4=I(N,4)                                                        SLID0071
      MIN=MINO(I1,I2,I3,I4)                                            SLID0072
      IF(I4.EQ.MIN)GO TO 90                                            SLID0073
      IF(I3.EQ.MIN)GO TO 100                                           SLID0074
      IF(I2.EQ.MIN)GO TO 110                                           SLID0075
      IF(I1.EQ.MIN)GO TO 120                                           SLID0076
   90 CONTINUE                                                         SLID0077
      J1=I4                                                            SLID0078
      J2=I1                                                            SLID0079
      J3=I2                                                            SLID0080
      J4=I3                                                            SLID0081
      GO TO 130                                                        SLID0082
  100 CONTINUE                                                         SLID0083
      J1=I3                                                            SLID0084
      J2=I4                                                            SLID0085
      J3=I1                                                            SLID0086
      J4=I2                                                            SLID0087
      GO TO 130                                                        SLID0088
  110 CONTINUE                                                         SLID0089
      J1=I2                                                            SLID0090
      J2=I3                                                            SLID0091
      J3=I4                                                            SLID0092
      J4=I1                                                            SLID0093
      GO TO 130                                                        SLID0094
  120 CONTINUE                                                         SLID0095
      J1=I1                                                            SLID0096
      J2=I2                                                            SLID0097
      J3=I3                                                            SLID0098
      J4=I4                                                            SLID0099
  130 CONTINUE                                                         SLID0100
      I(N,1)=J1                                                        SLID0101
      I(N,2)=J2                                                        SLID0102
      I(N,3)=J3                                                        SLID0103
      I(N,4)=J4                                                        SLID0104
  140 CONTINUE                                                         SLID0105
```

```
C
C  REMOVE DUPLICATE SURFACES                                          SLID0106
C                                                                     SLID0107
      DO 160 N=NMIN,NMAX                                              SLID0108
      IF(IF(N).EQ.1)GO TO 160                                         SLID0109
      I1=I(N,1)                                                       SLID0110
      I3=I(N,3)                                                       SLID0111
      NN=N+1                                                          SLID0112
      DO 150 M=NN,NMAX                                                SLID0113
      IF(I1.NE.I(M,1))GO TO 150                                       SLID0114
      IF(I3.NE.I(M,3))GO TO 150                                       SLID0115
      IF(N)=1                                                         SLID0116
      IF(M)=1                                                         SLID0117
      GO TO 160                                                       SLID0118
  150 CONTINUE                                                        SLID0119
  160 CONTINUE                                                        SLID0120
      NNMAX=0                                                         SLID0121
      REWIND 9                                                        SLID0122
C                                                                     SLID0123
C  COUNT THE NUMBER OF VISIBLE SURFACES                               SLID0124
C                                                                     SLID0125
      DO 170 N=1,NMAX                                                 SLID0126
      IF(IF(N).EQ.1)GO TO 170                                         SLID0127
      NNMAX=NNMAX+1                                                   SLID0128
  170 CONTINUE                                                        SLID0129
C                                                                     SLID0130
C  OUTPUT JOB STATISTICS                                              SLID0131
C                                                                     SLID0132
      WRITE(5,200)NMAX,NNMAX                                          SLID0133
      RETURN                                                          SLID0134
  180 FORMAT(I10,8I5)                                                 SLID0135
  190 FORMAT(4I5)                                                     SLID0136
  200 FORMAT(' NUMBER OF ELEMENTS WAS: ',I5,/,' AFTER PROCESSING      SLID0137
     1: ',I5)                                                         SLID0138
      END                                                             SLID0139
      SUBROUTINE DIVIDE(NDEL,I6,I5,I4,I3,I2,I1)                       SLID0140
C                                                                     DVDE0001
C  SUBROUTINE DIVIDE UNPACKS AN ARRAY                                 DVDE0002
C                                                                     DVDE0003
      DIMENSION I(6)                                                  DVDE0004
      I(1)=I1                                                         DVDE0005
      I(2)=I2                                                         DVDE0006
      I(3)=I3                                                         DVDE0007
      I(4)=I4                                                         DVDE0008
      I(5)=I5                                                         DVDE0009
      I(6)=I6                                                         DVDE0010
      DO 10 J=1,6                                                     DVDE0011
      I(J)=NDEL/(10**(6-J))                                           DVDE0012
   10 NDEL=NDEL-I(J)*(10**(6-J))                                      DVDE0013
      I1=I(1)                                                         DVDE0014
      I2=I(2)                                                         DVDE0015
      I3=I(3)                                                         DVDE0016
      I4=I(4)                                                         DVDE0017
      I5=I(5)                                                         DVDE0018
      I6=I(6)                                                         DVDE0019
      RETURN                                                          DVDE0020
      END                                                             DVDE0021
                                                                      DVDE0022
```

3. Fortran Listing of Free Format Routines

```
C                                                                       GET0001
C  ROUTINES FOR SIMULATING FREE FORMAT I/O FOR INGRID                   GET0002
C                                                                       GET0003
      SUBROUTINE GET                                                    GET0004
C                                                                       GET0005
C  SUBROUTINE GET PROCESS THE NEXT COMMAND                              GET0006
C                                                                       GET0007
      COMMON/REGION/IARRAY(100,6),NREG                                  GET0008
      DIMENSION CHAR(400)                                               GET0009
      NREG=0                                                            GET0010
      IEND=0                                                            GET0011
C                                                                       GET0012
C  GET THE COMMAND                                                      GET0013
C                                                                       GET0014
      DO 20 I=1,5                                                       GET0015
      IF=I*80                                                           GET0016
      IS=IF-79                                                          GET0017
      READ(6,50)(CHAR(J),J=IS,IF)                                       GET0018
      DO 10 J=IS,IF                                                     GET0019
      IF(CHAR(J).NE.';')GO TO 10                                        GET0020
      IEND=J                                                            GET0021
      GO TO 30                                                          GET0022
   10 CONTINUE                                                          GET0023
   20 CONTINUE                                                          GET0024
   30 CONTINUE                                                          GET0025
      IF(IEND.EQ.1)RETURN                                               GET0026
      I=1                                                               GET0027
C                                                                       GET0028
C  PROCESS THE COMMAND STRING                                           GET0029
C                                                                       GET0030
   40 CONTINUE                                                          GET0031
      IF(I.EQ.IEND)RETURN                                               GET0032
C                                                                       GET0033
C  PROCESS INDEX INFORMATION                                            GET0034
C                                                                       GET0035
      IF(CHAR(I).EQ.'(')CALL INDEX(CHAR,I)                              GET0036
C                                                                       GET0037
C  PROCESS COMMANDS                                                     GET0038
C                                                                       GET0039
      IF(CHAR(I).EQ.'/')CALL LIST(CHAR,I)                               GET0040
      I=I+1                                                             GET0041
      GO TO 40                                                          GET0042
   50 FORMAT(80A1)                                                      GET0043
   60 FORMAT(1X,A1)                                                     GET0044
      END                                                               GET0045
      SUBROUTINE INDEX(CHAR,I)                                          INDX0001
C                                                                       INDX0002
C  SUBROUTINE INDEX DETERMINES WHICH REGIONS ARE NEEDED                 INDX0003
C                                                                       INDX0004
      DIMENSION CHAR(400)                                               INDX0005
      COMMON/REGION/IARRAY(100,6),NREG                                  INDX0006
      K=1                                                               INDX0007
      NREG=NREG+1                                                       INDX0008
      J=NREG                                                            INDX0009
   10 CONTINUE                                                          INDX0010
      I=I+1                                                             INDX0011
      IF(CHAR(I).EQ.')')RETURN                                          INDX0012
      IF(CHAR(I).NE.'(')GO TO 20                                        INDX0013
C                                                                       INDX0014
C  INFORMATION IS AN INDEX PROGRESSION                                  INDX0015
```

```
C                                                                       INDX0016
      NREQ=NREQ-1                                                       INDX0017
      CALL PROGSN(CHAR,I)                                               INDX0018
      RETURN                                                            INDX0019
   20 IF(CHAR(I).EQ.' ')GO TO 10                                        INDX0020
      IF(CHAR(I).EQ.',')GO TO 10                                        INDX0021
      CALL NUMBER(CHAR,I,X)                                             INDX0022
      IARRAY(J,K)=IFIX(X)                                               INDX0023
      K=K+1                                                             INDX0024
      GO TO 10                                                          INDX0025
      END                                                               INDX0026
      SUBROUTINE NUMBER(CHAR,I,X)                                       NMBR0001
C                                                                       NMBR0002
C  SUBROUTINE NUMBER DETERMINES WHAT NUMBER A CHARACTER STRING          NMBR0003
C  REPRESENTS                                                           NMBR0004
C                                                                       NMBR0005
      DIMENSION CHAR(400)                                               NMBR0006
      X=0.0                                                             NMBR0007
      IEXP=0                                                            NMBR0008
      R1=10.0                                                           NMBR0009
      SX=1.0                                                            NMBR0010
      E=0.0                                                             NMBR0011
      R2=1.0                                                            NMBR0012
      SE=1.0                                                            NMBR0013
   10 CONTINUE                                                          NMBR0014
      C=CHAR(I)                                                         NMBR0015
      IF(C.EQ.' ')GO TO 60                                              NMBR0016
      IF(C.EQ.',')GO TO 60                                              NMBR0017
      IF(C.EQ.')')GO TO 60                                              NMBR0018
      IF(C.EQ.'/')GO TO 60                                              NMBR0019
      IF(C.EQ.';')GO TO 60                                              NMBR0020
      IF(C.NE.'-')GO TO 20                                              NMBR0021
      IF(IEXP.EQ.0)SX=-1.0                                              NMBR0022
      IF(IEXP.EQ.1)SE=-1.0                                              NMBR0023
      GO TO 50                                                          NMBR0024
   20 IF(C.NE.'.')GO TO 30                                              NMBR0025
      R1=1.0                                                            NMBR0026
      R2=0.1                                                            NMBR0027
      GO TO 50                                                          NMBR0028
   30 IF(C.NE.'E')GO TO 40                                              NMBR0029
      R1=10.0                                                           NMBR0030
      R2=1.0                                                            NMBR0031
      IEXP=1                                                            NMBR0032
      GO TO 50                                                          NMBR0033
   40 CONTINUE                                                          NMBR0034
      CALL DIGIT(C,NUM)                                                 NMBR0035
      IF(IEXP.EQ.0)X=X*R1+R2*NUM                                        NMBR0036
      IF(IEXP.EQ.1)E=E*R1+R2*NUM                                        NMBR0037
      IF(R2.LT.0.5)R2=R2/10.0                                           NMBR0038
   50 CONTINUE                                                          NMBR0039
      I=I+1                                                             NMBR0040
      GO TO 10                                                          NMBR0041
   60 CONTINUE                                                          NMBR0042
      E=SE*E                                                            NMBR0043
      E=10.0**E                                                         NMBR0044
      X=X*E*SX                                                          NMBR0045
      I=I-1                                                             NMBR0046
      RETURN                                                            NMBR0047
      END                                                               NMBR0048
      SUBROUTINE DIGIT(CHAR,NUM)                                        DGIT0001
C                                                                       DGIT0002
```

```
C  CONVERT A CHARACTER TO A DIGIT                                    DGIT0003
C                                                                    DGIT0004
      DIMENSION D(10)                                                DGIT0005
      DATA D/1H0,1H1,1H2,1H3,1H4,1H5,1H6,1H7,1H8,1H9/                DGIT0006
      DO 10 J=1,10                                                   DGIT0007
      IF(CHAR.EQ.D(J))GO TO 20                                       DGIT0008
   10 CONTINUE                                                       DGIT0009
   20 NUM=J-1                                                        DGIT0010
      RETURN                                                         DGIT0011
      END                                                            DGIT0012
      SUBROUTINE PROGSN(CHAR,I)                                      PRGS0001
C                                                                    PRGS0002
C  INPUT AN INDEX PROGRESSION                                        PRGS0003
C                                                                    PRGS0004
      DIMENSION CHAR(400),II(20,3)                                   PRGS0005
      N=1                                                            PRGS0006
      DO 10 K=1,3                                                    PRGS0007
      DO 10 J=1,20                                                   PRGS0008
   10 II(J,K)=0                                                      PRGS0009
      J=1                                                            PRGS0010
   20 CONTINUE                                                       PRGS0011
      I=I+1                                                          PRGS0012
      C=CHAR(I)                                                      PRGS0013
      IF(N.EQ.3.AND.C.EQ.')')GO TO 40                                PRGS0014
      IF(C.NE.')')GO TO 30                                           PRGS0015
      J=1                                                            PRGS0016
      N=N+1                                                          PRGS0017
      GO TO 20                                                       PRGS0018
   30 IF(C.EQ.' ')GO TO 20                                           PRGS0019
      IF(C.EQ.',')GO TO 20                                           PRGS0020
      IF(C.EQ.'(')GO TO 20                                           PRGS0021
      CALL NUMBER(CHAR,I,X)                                          PRGS0022
      II(J,N)=IFIX(X)                                                PRGS0023
      J=J+1                                                          PRGS0024
      GO TO 20                                                       PRGS0025
   40 CONTINUE                                                       PRGS0026
      CALL SEQ(II)                                                   PRGS0027
      RETURN                                                         PRGS0028
   50 FORMAT(20I4)                                                   PRGS0029
      END                                                            PRGS0030
      SUBROUTINE SEQ(II)                                             SEQ0001
C                                                                    SEQ0002
C  CONVERT AN INDEX PROGRESSION TO ITS INDIVIDUAL REGIONS            SEQ0003
C                                                                    SEQ0004
      DIMENSION II(20,3),M(3),IMX(6)                                 SEQ0005
      COMMON/REGION/IARRAY(100,6),NREG                               SEQ0006
      EQUIVALENCE (IMX(1),IM),(IMX(2),JM),(IMX(3),KM),(IMX(4),IX),   SEQ0007
     1(IMX(5),JX),(IMX(6),KX)                                        SEQ0008
      DO 10 I=1,3                                                    SEQ0009
      DO 10 J=1,20                                                   SEQ0010
      IF(II(J,I).NE.0)M(I)=J                                         SEQ0011
   10 CONTINUE                                                       SEQ0012
      DO 20 I=1,3                                                    SEQ0013
   20 IF(II(M(I),I).GT.0)M(I)=M(I)-1                                 SEQ0014
      DO 120 I=1,M(1)                                                SEQ0015
      DO 110 J=1,M(2)                                                SEQ0016
      DO 100 K=1,M(3)                                                SEQ0017
      IM=II(I,1)                                                     SEQ0018
      JM=II(J,2)                                                     SEQ0019
      KM=II(K,3)                                                     SEQ0020
      IX=II(I+1,1)                                                   SEQ0021
```

```
      JX=II(J+1,2)                                                      SEQ0022
      KX=II(K+1,3)                                                      SEQ0023
      IF(IM.EQ.0.OR.JM.EQ.0.OR.KM.EQ.0)GO TO 100                        SEQ0024
C                                                                       SEQ0025
C  CHECK FOR SOLID REGIONS                                              SEQ0026
C                                                                       SEQ0027
      IF(IM.LE.0.OR.JM.LE.0.OR.KM.LE.0)GO TO 40                         SEQ0028
      IF(IX.LE.0.OR.JX.LE.0.OR.KX.LE.0)GO TO 100                        SEQ0029
      NREG=NREG+1                                                       SEQ0030
      DO 30 L=1,6                                                       SEQ0031
   30 IARRAY(NREG,L)=IMX(L)                                             SEQ0032
      GO TO 100                                                         SEQ0033
   40 CONTINUE                                                          SEQ0034
C                                                                       SEQ0035
C  CHECK FOR PLANAR REGIONS                                             SEQ0036
C                                                                       SEQ0037
      IF(IM.GE.0)GO TO 60                                               SEQ0038
      IF(JX.EQ.0.OR.KX.EQ.0)GO TO 60                                    SEQ0039
      NREG=NREG+1                                                       SEQ0040
      DO 50 L=1,6                                                       SEQ0041
   50 IARRAY(NREG,L)=IABS(IMX(L))                                       SEQ0042
      IARRAY(NREG,4)=-IM                                                SEQ0043
   60 IF(JM.GE.0)GO TO 80                                               SEQ0044
      IF(IX.EQ.0.OR.KX.EQ.0)GO TO 80                                    SEQ0045
      NREG=NREG+1                                                       SEQ0046
      DO 70 L=1,6                                                       SEQ0047
   70 IARRAY(NREG,L)=IABS(IMX(L))                                       SEQ0048
      IARRAY(NREG,5)=-JM                                                SEQ0049
   80 IF(KM.GE.0)GO TO 100                                              SEQ0050
      IF(IX.EQ.0.OR.JX.EQ.0)GO TO 100                                   SEQ0051
      NREG=NREG+1                                                       SEQ0052
      DO 90 L=1,6                                                       SEQ0053
   90 IARRAY(NREG,L)=IABS(IMX(L))                                       SEQ0054
      IARRAY(NREG,6)=-KM                                                SEQ0055
  100 CONTINUE                                                          SEQ0056
  110 CONTINUE                                                          SEQ0057
  120 CONTINUE                                                          SEQ0058
      RETURN                                                            SEQ0059
      END                                                               SEQ0060
      SUBROUTINE LIST(CHAR,I)                                           LIST0001
C                                                                       LIST0002
C  SUBROUTINE LIST PROCESS COMMAND LISTS                                LIST0003
C                                                                       LIST0004
      DIMENSION CHAR(400)                                               LIST0005
      COMMON/FUNCTN/ICMD(10),NPT(10),PARAM(100),NPARAM(10),NCMD         LIST0006
      NPT(1)=1                                                          LIST0007
      NCMD=NCMD+1                                                       LIST0008
      IF(NCMD.GT.1)NPT(NCMD)=NPT(NCMD-1)+NPARAM(NCMD-1)                 LIST0009
      CALL WORD(CHAR,I,NW)                                              LIST0010
      ICMD(NCMD)=NW                                                     LIST0011
   10 CONTINUE                                                          LIST0012
      I=I+1                                                             LIST0013
      C=CHAR(I)                                                         LIST0014
      IF(C.EQ.' ')GO TO 10                                              LIST0015
      IF(C.EQ.',')GO TO 10                                              LIST0016
      IF(C.EQ.'=')GO TO 10                                              LIST0017
      IF(C.EQ.'/')GO TO 20                                              LIST0018
      IF(C.EQ.';')GO TO 20                                              LIST0019
      CALL NUMBER(CHAR,I,X)                                             LIST0020
      J=NPT(NCMD)+NPARAM(NCMD)                                          LIST0021
      PARAM(J)=X                                                        LIST0022
```

```
      NPARAM(NCMD)=NPARAM(NCMD)+1                                       LIST0023
      GO TO 10                                                          LIST0024
   20 CONTINUE                                                          LIST0025
      I=I-1                                                             LIST0026
      RETURN                                                            LIST0027
      END                                                               LIST0028
      SUBROUTINE WORD(CHAR,I,NW)                                        WORD0001
C                                                                       WORD00C2
C  SUBROUTINE WORD DETERMINES WHAT WORD A CHARACTER STRING REPRESENTS    WORD00C3
C                                                                       WORD0004
      DIMENSION CHAR(400)                                               WORD00C5
      NW=0                                                              WORD00C6
      NCNT=0                                                            WORD00C7
   10 CONTINUE                                                          WORD00C8
      I=I+1                                                             WORD00C9
      C=CHAR(I)                                                         WORD001C
      IF(C.EQ.' ')GO TO 20                                              WORD0011
      IF(C.EQ.',')GO TO 20                                              WORD0012
      IF(C.EQ.'=')GO TO 20                                              WORD0013
      IF(C.EQ.';')GO TO 20                                              WORD0014
      IF(C.EQ.'(')GO TO 20                                              WORD0015
      IF(C.EQ.'/')GO TO 20                                              WORD0016
      NCNT=NCNT+1                                                       WORD0017
      IF(NCNT.GT.4)GO TO 10                                             WORD0018
      CALL ALPHA(C,NUM)                                                 WORD0019
      NW=NW*100+NUM                                                     WORD002C
      GO TO 10                                                          WORC0021
   20 CONTINUE                                                          WORD0022
      I=I-1                                                             WORD0023
      RETURN                                                            WORD0024
      END                                                               WORD0025
      SUBROUTINE ALPHA(CHAR,NUM)                                        ALPH00C1
C                                                                       ALPH00C2
C  SUBROUTINE ALPHA CONVERTS A CHARACTER TO EQUIVALENT NUMBER           ALPH00C3
C                                                                       ALPH00C4
      DIMENSION D(27)                                                   ALPH00C5
      DATA D/'A','B','C','D','E','F','G','H','I','J','K','L',          ALPH00C6
     1'M','N','O','P','Q','R','S','T','U','V','W','X','Y','Z','.'/     ALPH00C7
      NUM=0                                                             ALPH00C8
      DO 10 I=1,27                                                      ALPH00C9
      IF(CHAR.NE.D(I))GO TO 10                                          ALPH001C
      NUM=I+10                                                          ALPH0011
      GO TO 20                                                          ALPH0012
   10 CONTINUE                                                          ALPH0013
      CALL DIGIT(CHAR,NUM)                                              ALPH0014
   20 CONTINUE                                                          ALPH0015
      RETURN                                                            ALPH0016
      END                                                               ALPH0017
```

VITA

Douglas Walter Stillman was born in Philadelphia, Pennsylvania on September 12, 1959. He entered the University of Tennessee in December 1976 and in March 1980 received a Bachelor of Science degree in Engineering Science from the University of Tennessee.

After completing the requirements for a Master of Science degree, he will begin work at Lawrence Livermore National Laboratory in Livermore, California.

The author is a member of the Society for Engineering Science and the American Society of Civil Engineers.

He is married to the former Ying Chi Wu of Kaoshiung, Taiwan.