

AD-A046 385

NAVAL POSTGRADUATE SCHOOL MONTEREY CALIF
A FINITE ELEMENT PREPROCESSOR FOR SAP IV AND ADINA. (U)
SEP 77 A E KIBLER

F/G 9/2
(U)

F/G 9/2

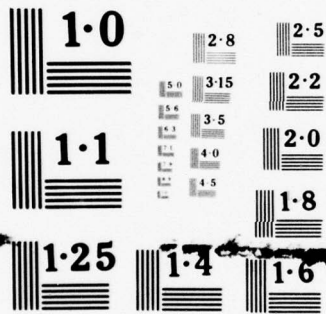
UNCLASSIFIED

1 of 2
ADA
046 385

NL

NL





NATIONAL BUREAU OF STANDARDS
MICROCOPY RESOLUTION TEST CHART

AD A 0 46385

2
B.S.

NAVAL POSTGRADUATE SCHOOL

Monterey, California



DDC
REFRIMED
NOV 15 1977
F

THESIS

AD No. _____
DDC FILE COPY:

A FINITE ELEMENT PREPROCESSOR
FOR SAP IV AND ADINA

by

Adrian Earl Kibler, Jr.

September 1977

Thesis Advisor: G. Cantin

Approved for public release, distribution unlimited.

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) A Finite Element Preprocessor for SAP IV and ADINA.		5. TYPE OF REPORT & PERIOD COVERED Master's Thesis September 1977
7. AUTHOR(s) Adrian Earl/Kibler, Jr		6. PERFORMING ORG. REPORT NUMBER
9. PERFORMING ORGANIZATION NAME AND ADDRESS Naval Postgraduate School Monterey, California 93940		8. CONTRACT OR GRANT NUMBER(s)
11. CONTROLLING OFFICE NAME AND ADDRESS Naval Postgraduate School Monterey, California 93940		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office) <i>12 130p.</i>		12. REPORT DATE September 1977
		13. NUMBER OF PAGES 131
		15. SECURITY CLASS. (of this report) Unclassified
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE
16. DISTRIBUTION STATEMENT (of this Report) Approved for public release; distribution unlimited.		
17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)		
18. SUPPLEMENTARY NOTES		
19. KEY WORDS (Continue on reverse side if necessary and identify by block number) Data Checking, Preprocessor, Postprocessor, Finite Element, Computer Code, SAP IV, ADINA		
20. ABSTRACT (Continue on reverse side if necessary and identify by block number) → The primary purpose of this thesis was to provide a method of checking the geometry and element connectivity input data for two finite element programs, ADINA and SAP IV. This preprocessor will accept the ADINA or SAP IV data deck, with minor modifications, and generate a graphical display of the finite element model. The display is an oblique orthographic projection, and any orientation may be specified. Several		

DD FORM 1473
1 JAN 73

EDITION OF 1 NOV 66 IS OBSOLETE
S/N 0102-014-6601

UNCLASSIFIED

1 SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

251 450

LB

(20. ABSTRACT Continued)

→ options are available: exploded plots, partial plots, node numbering, element numbering, and others. Elements with three nodes on the same edge are plotted with a continuous curve on each edge generated by an interpolated parabola. Displacement postprocessing capability also exists.

ACCESSION for

NEWS Write Section
DDC Buff Section

UNANNOUNCED
DISSEMINATION

BY
DISTRIBUTION AVAILABILITY CODES
SPECIAL

A

Approved for public release; distribution unlimited.

A Finite Element Preprocessor
for SAP IV and ADINA

by

Adrian Earl Kibler, Jr.
Lieutenant, United States Navy
B.S.E.E., Pennsylvania State University, 1972

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN MECHANICAL ENGINEERING

from the

NAVAL POSTGRADUATE SCHOOL
September 1977

Author

Adrian Earl Kibler Jr.

Approved by:

Gilles Cantin

Thesis Advisor

R. E. Newton

Second Reader

Allen E. Fuchs

Chairman, Department of Mechanical Engineering

Robert A. Jinn

Dean of Science and Engineering

ABSTRACT

The primary purpose of this thesis was to provide a method of checking the geometry and element connectivity input data for two finite element programs, ADINA and SAP IV. This preprocessor will accept the ADINA or SAP IV data deck, with minor modifications, and generate a graphical display of the finite element model. The display is an oblique orthographic projection, and any orientation may be specified. Several options are available: exploded plots, partial plots, node numbering, element numbering, and others. Elements with three nodes on the same edge are plotted with a continuous curve on each edge generated by an interpolated parabola. Displacement postprocessing capability also exists.

TABLE OF CONTENTS

I.	INTRODUCTION -----	10
A.	GENERAL -----	10
	1. Preprocessing -----	10
	2. Postprocessing -----	11
B.	HISTORY OF DEVELOPMENT OF PSAP1 -----	12
	1. SUBROUTINE PSAP Implementation -----	12
	2. Motivation for SUBROUTINE PSAP1 -----	12
C.	PRESENT CAPABILITY -----	13
D.	EASE OF MODIFICATION -----	15
II.	PROGRAM ORGANIZATION AND DESCRIPTION OF OPERATION -----	16
A.	PSAP1 FLOW CHART -----	16
B.	NAMELIST AND EULER ANGLES -----	16
	1. NAMELIST OPTION -----	16
	2. NAMELIST PICT -----	18
	a. Oblique Orthographic Projections (Euler Angles) -----	18
	b. Scaling -----	20
	c. Partial Plots -----	20
C.	NODAL POINT (GEOMETRY) INFORMATION READ-IN -----	21
D.	ELEMENT (CONNECTIVITY) INFORMATION READ-IN -----	21
E.	DISPLACEMENT DATA READ-IN FOR DISPLACEMENT POSTPROCESSING -----	21
F.	PLOTTING LOGIC -----	24

III.	PSAP1 SAMPLE PROBLEMS -----	27
A.	ADINA EXAMPLES -----	27
1.	Reinforced Concrete Beam -----	27
2.	Flat Plate With Hole -----	35
B.	SAP IV EXAMPLES -----	35
1.	SAP IV Truss Problem -----	35
2.	Cylindrical Bar with Spherical Hole -----	46
IV.	CONCLUSIONS AND RECOMMENDATIONS -----	59
APPENDIX A:	PSAP1 USER'S MANUAL -----	60
APPENDIX B:	SUBROUTINE PSAP1 LISTING -----	78
LIST OF REFERENCES	-----	129
INITIAL DISTRIBUTION LIST	-----	130

LIST OF FIGURES

1.	Flow chart of program PSAP1 -----	17
2.	Flow chart for reading geometry and element connectivity data -----	22
3.	Nodal point and displacement storage arrays -----	23
4.	Flow chart for PSAP1 plotting subroutine, PLOTX --	25
5.	Example 1, ADINA truss and 8 node plane elements, Reinforced Concrete Beam (Given on page 84, reference 2) -----	28
6.	Example 1, ADINA input deck listing. Page 1 of 2 -----	29
	Page 2 of 2 -----	30
7.	Example 1, PSAP1 input deck listing. Page 1 of 2 -----	31
	Page 2 of 2 -----	32
8.	Example 1, PSAP1 output graphs. (a) Undeformed structure, nodes numbered -----	33
	(b) Exploded plot, elements numbered -----	34
9.	Example 2, flat plate with a hole in tension -----	36
10.	Example 2, PSAP1 input deck listing. Page 1 of 3 -----	37
	Page 2 of 3 -----	38
	Page 3 of 3 -----	39
11.	Example 2, PSAP1 output graphs. (a) Actual mesh, nodes numbered -----	40
	(b) Symmetric representation, elements numbered --	41
12.	Example 3, SAP IV truss element problem -----	42
13.	Example 3, PSAP1 input deck listing. Page 1 of 2 -----	43
	Page 2 of 2 -----	44
14.	Example 3, PSAP1 output graphs -----	45
15.	Example 4, six inch diameter cylindrical bar with a two inch diameter spherical void on the center line under pressure -----	47

LIST OF FIGURES (CONTINUED)

16.	Example 4, PSAP1 input deck listing.	
	Page 1 of 6 -----	48
	Page 2 of 6 -----	49
	Page 3 of 6 -----	50
	Page 4 of 6 -----	51
	Page 5 of 6 -----	52
	Page 6 of 6 -----	53
17.	Example 4, PSAP1 output graphs.	
	(a) Total structure -----	54
	(b) Elements 1-10, nodes numbered -----	55
	(c) Elements 1-10, elements numbered, exploded plot -----	56
	(d) Elements 11-20, nodes numbered -----	57
	(e) Elements 11-20, elements numbered, exploded plot -----	58
18.	Coordinate system and Euler angles (rotations) for oblique orthographic projection shown in X-Z viewing plane. (Taken from reference 4, page 127) -----	70

ACKNOWLEDGMENT

I would like to acknowledge my sincere and deep appreciation to all those who were instrumental in providing me with motivation and assistance in the development of this thesis. In particular I would like to thank Professor Gilles Cantin, my thesis advisor, for his friendship, assistance and excellent academic guidance; Professor R. E. Ball for providing me with the original deck of cards and helpful guidance; and my wife and family for the sacrifices they have made while I have been in pursuit of my profession.

I. INTRODUCTION

A. GENERAL

The continued developments and advancements of the finite element method this last decade have provided greater machine capabilities than ever before. Here at the Naval Postgraduate School (NPS), two of the more favorable finite element programs are SAP IV [reference 1] for linear analysis and ADINA [reference 2] for nonlinear analysis. However, with the large amounts of numerical input/output data and automatic mesh generation, it is impractical to check and reduce this data without a graphical representation. Data checking is divided into two areas: preprocessing and postprocessing.

1. Preprocessing

Preprocessing is the checking of the input data deck. Errors in a finite element program occur basically in two areas. First, how close is the mathematical model (boundary conditions, loading conditions, material properties, etc.) to the real problem? Second, are numerical errors present, or did misinterpretation of instructions occur in data deck preparation? Of the second type, most common errors are found in the geometry and element connectivity data. Preprocessing includes the forming of a graphical representation of the finite element model on which geometry and element errors are easily detected. When

node and element numbering options are available, the graph aids in the physical interpretation of the output data. Preprocessing is not a foolproof method of eliminating errors, but it does provide a tremendous advantage to the user. Preprocessors may be incorporated into the data check mode of the finite element program. However, to modify a large and complex program is dangerous. This may not be the best approach. A safer method is to develop a preprocessor which will read the finite element program deck separately with minimum modifications to that deck.

2. Postprocessing

Though not as important as preprocessing, postprocessing is extremely helpful in output analysis. Probably the most common and useful type of postprocessing is the contour plot. Appendix C of reference 4 lists a program developed to produce contour plots of stress data from finite element models. Contour plots can easily be adapted to a 2D system, but 3D requires plotting the contours on 2D surfaces, a bit more complicated. Two methods of postprocessing of displacements are the plotting of a deformed model or placing scaled vectors at the nodes. In the cases where the displacements are small, multiplication by a magnification factor produces an exaggerated representation. Like preprocessing, postprocessing can be incorporated into the finite element program directly, or done separately. When done separately, the finite element program must still be modified slightly to obtain a punched deck of the stresses

and displacements in the desired format. This thesis is primarily concerned with preprocessing.

B. HISTORY OF DEVELOPMENT OF PSAP1

A package [reference 4] containing digital computer programs for generating oblique orthographic projections and contour plots was produced by the National Aeronautics and Space Administration's (NASA) Langley Research Center (LRC) and distributed by the National Technical Information Service in January, 1975. The programs are completely general. Both programs contain options for selecting various plotting equipment including CALCOMP, VARIAN, and cathode ray tube (CRT) displays. With minor modifications, they can be adapted to most any system.

1. SUBROUTINE PSAP Implementation

Losh [reference 6], for his master's thesis in aeronautical engineering, implemented the preprocessor and postprocessor program, PSAP, at NPS in December, 1976. Modifying the LRC package [reference 4], Losh adapted SUBROUTINE PSAP to the NPS IBM 360/67 system using the CALCOMP model 765 plotter. PSAP serves as a preprocessor for SAP IV models, and serves as a postprocessor for displacements of those models. Unfortunately, PSAP is severely limited in the type of elements it can plot.

2. Motivation for SUBROUTINE PSAP1

With the introduction of the ADINA [reference 2] program at NPS in January, 1977, and with expectation of

doing future analysis on ceramic turbine blades, it was desired to expand PSAP to include all ADINA elements and the 8-20 node brick elements in SAP IV. Like PSAP, PSAP1 contains preprocessing and displacement postprocessing capabilities. PSAP1 is presented in this thesis and has the following improvements over PSAP:

- a. Preprocessing for all ADINA elements.
- b. SAP IV 8 and 8-20 node elements.
- c. Expansion of SUBROUTINE ERROR.
- d. Interpolation of curves using shape functions [reference 3] through three points on the edges of the 8-20 node brick elements and the 4-8 node plane elements.
- e. Improvements in defining the plot origin.
- f. Addition of an option (ISCALE = 0) to plot sections of a model without losing perspective.
- g. Several other minor modifications considered improvements.

C. PRESENT CAPABILITY

PSAP1 has the capability to plot all ADINA elements and all SAP IV elements except the pipe element. It will interpolate curves on the edges of 4-8 node plane elements and 8-20 node brick elements. Many options are given in Appendix A. Some of the more frequently used options are listed below:

1. Numbering of grid points (NOTAT = 1).
2. Numbering of the elements (NOTAT = 2).
3. Exploded plot (KDISP = 2).

4. Postprocessing of displacements (NUDISP or NVDISP or NWDISP = 1) in two forms: plot of deformed structure (KDISP = 1) or displacements represented by vectors at the nodes (KDISP = 3; see reference 6).

5. Symmetric representation about the XY (KSYMXY = 1), XZ (KSYMXZ = 1) or YZ (KSYMYZ = 1) planes.

6. Option to plot sections of the model (partial plot) to obtain a better view. Partial plots may be plotted to the scale of the complete model to avoid losing perspective (ISCALE = 0) or blown up to obtain a better view (ISCALE = 1). Multiple plots may be obtained using the same geometry and same displacement data (KODE = 1), same geometry and new displacement data (KODE = 2), or new geometry and new displacement data (KODE = 3; see figure 1).

In general, multiple plots (sections, partial plots, additional problems) present no problem. Plotting package user courtesy dictates that no more than 5 plots be placed on the CALCOMP plotter at any one time. Also, if the plots contain many elements (especially 8-20 node elements), it is possible to run out of space in the plotting data sets. When this happens, you will receive

ERROR IHC240I STAE, ABEND CODE IS: SYSTEM OB37 SYSPLIT.

The best thing to do is split the run into two jobs. If the job must be run on one job (i.e., a large number of elements in the model or an assembly drawing where the scale of multiple plots is the same), then SYSPLIT space

may be increased [references 7 and 8] by adding the card

```
//GO.SYSPLOT DD SPACE=(CYL,(needed space)),SYSOUT=C
```

just prior to card

```
//GO.FT10F001 DD UNIT = SYSDA
```

in Appendix A. It would be wise to seek advice from a consultant in Ingersoll 146 if additional plotting space is required.

D. EASE OF MODIFICATION

Both PSAP and PSAP1 are written to maintain as much generality as possible for ease of expansion and modification. Several FORTRAN statements, variables and subroutines are not used. They were left purposely unchanged. Although PSAP1 specifically reads ADINA and SAP IV data, it can easily be expanded to include any geometry and element data format. Simply study the read-in and storage methods (see Section II), and construct appropriate subroutines to read any particular format.

II. PROGRAM ORGANIZATION AND DESCRIPTION OF OPERATION

A. PSAP1 FLOW CHART

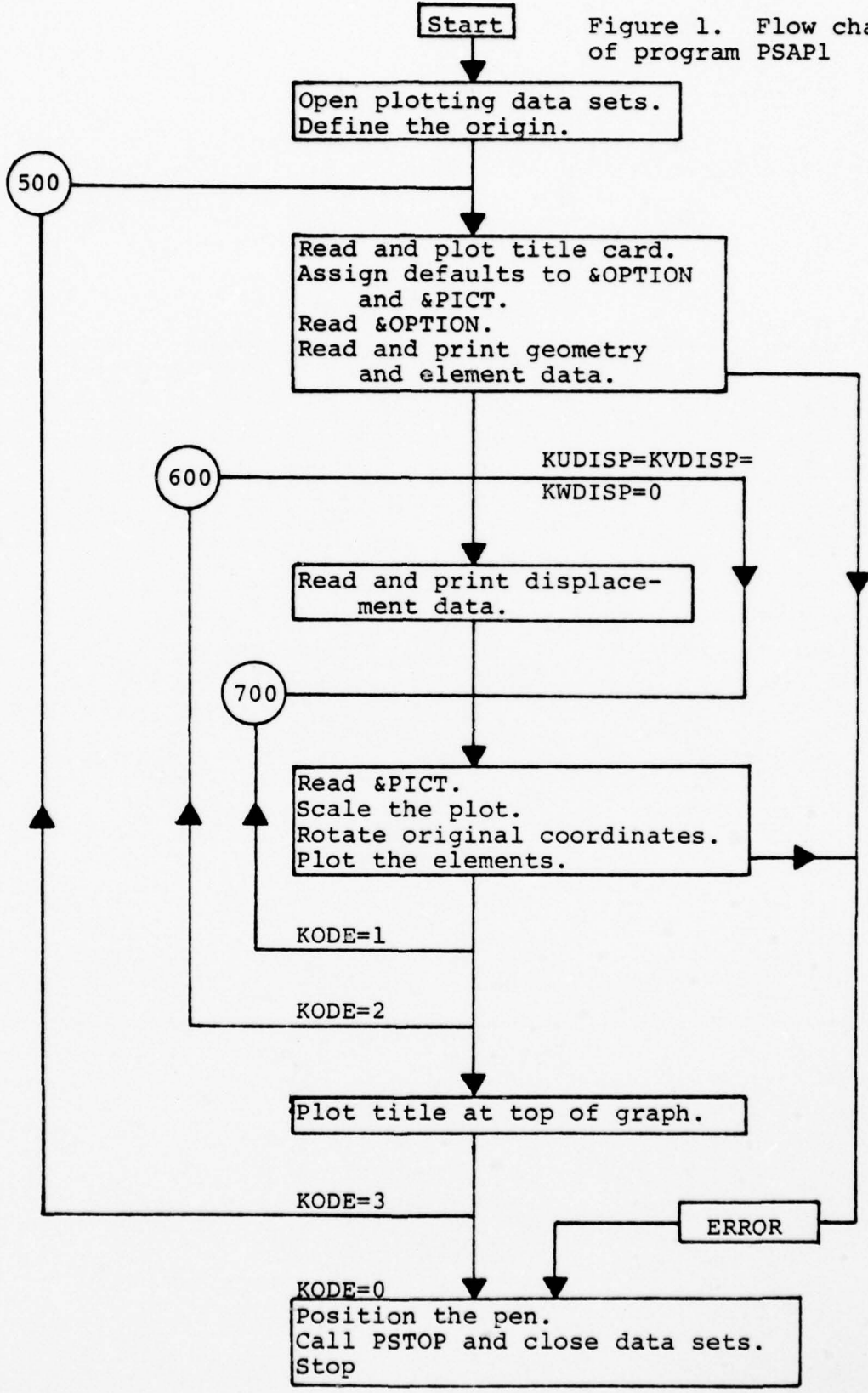
Figure 1 is a condensed flow chart of PSAP1. Probably the most important information given on this chart is the sequence in which the data cards, NAMELIST OPTION and NAMELIST PICT are read. Remember, when generating a sequence of plots, once a parameter has been defined, it retains that value until it is reassigned. Note that when $KODE = 1$ or 2 , the original values of NAMELIST OPTION and NAMELIST PICT are retained until they are changed. However, when $KODE = 3$ a new title card, NAMELIST OPTION and a set of problem data are read. All variables in NAMELIST OPTION and NAMELIST PICT are assigned their default values. A new problem begins in this case. It is important that the last NAMELIST PICT to be read must contain the value of $KODE = 0$.

B. NAMELISTS AND EULER ANGLES

1. NAMELIST OPTION

Description and default values of NAMELIST OPTION are given in Appendix A. Basically NAMELIST OPTION variables pertain to the given problem: the number of nodes, geometry format, displacement format, space between plots, and paper size. Default values for NAMELIST OPTION are set, and NAMELIST OPTION is read at the beginning of the problem. Assigned values will remain until exit from the

Figure 1. Flow chart of program PSAP1



program (KODE = 0) or a new set of problem data is read (KODE = 3).

2. NAMELIST PICT

Like NAMELIST OPTION, NAMELIST PICT variable descriptions and default values are given in Appendix A. Basically NAMELIST PICT variables pertain to a given plot. One NAMELIST OPTION may apply to several successive plots, but each NAMELIST PICT defines a unique plot. That plot may include the whole model, part of the model and any options defined in NAMELIST PICT. NAMELIST PICT also specifies the viewing plane through the Euler angles (figure 18, Appendix A).

a. Oblique Orthographic Projections (Euler Angles)

An example of an oblique orthographic projection of a finite element model is given in figure 18 in Appendix A. The model can be viewed in any selected orientation. Euler angle transformations are used to specify orientation of the model to be projected. As described in reference 4, this transformation resolves the coordinate system of the model to a principal viewing plane (i.e., X_0Y_0 , X_0Z_0 , Y_0Z_0) on which the display is to be plotted. Prior to rotation, the model coordinate system (X,Y,Z) is coincident with the coordinate system containing the viewing planes (X_0, Y_0, Z_0). The viewing planes are fixed, and the model is rotated about its model coordinate system. The rotations (ψ, θ, ϕ) of the body about the model axes (X,Y,Z) are shown in figure 18, Appendix A. The NAMELIST PICT variables KHORZ (horizontal

axis), KVERT (vertical axis), PSI (ψ), THETA (θ), and PHI (ϕ) specify the viewing plane and Euler angles. The order of the Euler angle rotations must be PSI, THETA and then PHI. Mathematical transformations are:

$$\begin{Bmatrix} X_0 \\ Y_0 \\ Z_0 \end{Bmatrix} = [\underline{A}_\phi] [\underline{A}_\theta] [\underline{A}_\psi] \begin{Bmatrix} X \\ Y \\ Z \end{Bmatrix}$$

$$[\underline{A}_\psi] = \begin{bmatrix} \cos \psi & -\sin \psi & 0 \\ \sin \psi & \cos \psi & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$[\underline{A}_\theta] = \begin{bmatrix} \cos \theta & 0 & \sin \theta \\ 0 & 1 & 0 \\ -\sin \theta & 0 & \cos \theta \end{bmatrix}$$

$$[\underline{A}_\phi] = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \phi & -\sin \phi \\ 0 & \sin \phi & \cos \phi \end{bmatrix}$$

SUBROUTINE ROTAT calculates the transformation matrices for every NAMELIST PICT, except if ISCALE = 0. ISCALE = 0 directs the scale of the plot to be the same as that of the

previous plot. Should a rotation occur with ISCALE = 0, the plot width could exceed the paper width.

b. Scaling

The safest scaling method is automatic scaling (ISCALE = 1). The user may specify a scale (ISCALE = 2) and the plot origin (XORGN,YORGN), but one must be careful not to run the plotting pen off the graph paper. ISCALE = 0 is a very useful option. The plot will use the same scale as the previous plot. It is useful in an assembly graph where examination of a mesh in sections without losing perspective is desired. Example 3, Section III, illustrates the option ISCALE = 0. When ISCALE = 1 in a NAMELIST PICT defining a partial plot, a "blow-up" of that section is obtained. ISCALE cannot be zero in the first NAMELIST PICT.

c. Partial Plots

To develop a partial plot, three methods of segregating elements exist: first, by the X, Y, and Z cutting planes; second, by node numbers, and, third, by element numbers. If a model has an area where the elements are relatively small, a "blow-up" may be desired. Choose a numbering scheme or coordinates to define the section to be segregated using one of the methods above. Example 3 (figure 14, Section III) uses X, Y, and Z cutting planes to define the partial plots. Example 4 (figure 17, Section III) uses element numbers to section the plots.

C. NODAL POINT (GEOMETRY) INFORMATION READ-IN

Nodal point data is read in by the GEOMn subroutines (GEOM1, GEOM2, and GEOM9, see figure 2). Since SAP IV and ADINA data decks are similar, SUBROUTINES GEOM1 and GEOM9 are also very similar. They are both constructed to read and generate data in exactly the same way as ADINA and SAP IV. All data not needed by PSAP1 is disregarded and the nodal point data is stored in array ZZZ (figure 3). After studying storage array ZZZ and GEOM1 (or GEOM9), a user moderately familiar with FORTRAN programming could easily construct a user supplied subroutine (GEOM2) to read the nodal point data in any desired format.

D. ELEMENT (CONNECTIVITY) INFORMATION READ-IN

After reading and storing the nodal point data, the element data must be read. The GEOMn subroutine will read the element control card (NPAR, references 1 and 2). SUBROUTINE ELTYPE (figure 2) calls the proper element subroutine to read the element data specified on the element control card. If several groups of elements are to be read, the process is repeated until all of the element groups have been read. Although the nodal point data is stored in array ZZZ, the element connectivity is read and stored on device 10 (disk).

E. DISPLACEMENT DATA READ-IN FOR DISPLACEMENT POSTPROCESSING

Displacement data may be read in (figure 1) by SUBROUTINE DATA9 (KDATA=9) or SUBROUTINES DATA1 or DATA5 (user supplied,

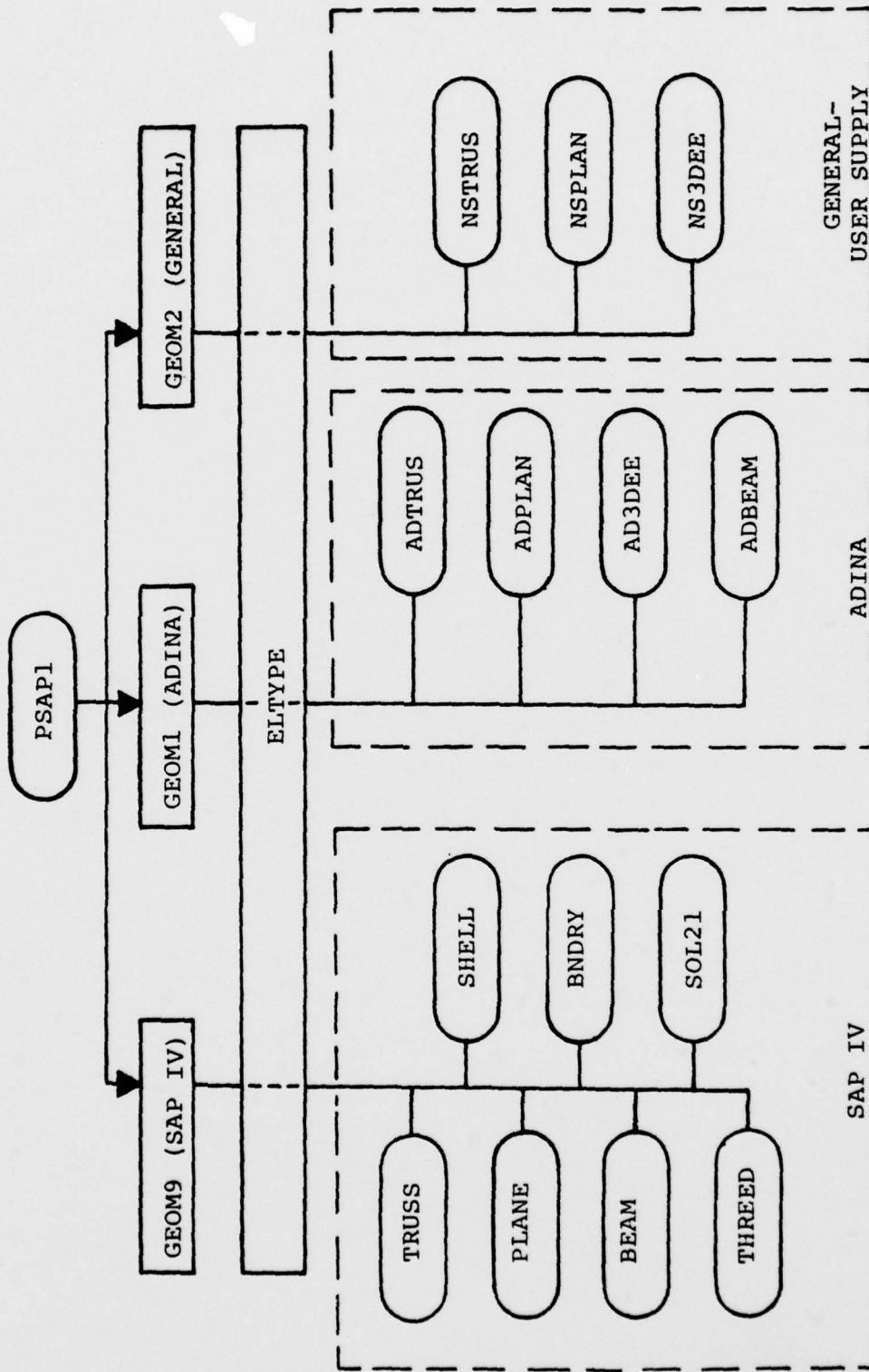


Figure 2. Flow chart for reading geometry and element connectivity data.

ZZZ (1)	ZZZ (N+1)	ZZZ (2N+1)	ZZZ (3N+1)	ZZZ (4N+1)	ZZZ (5N+1)	ZZZ (6N+1)
ZZZ (2)	ZZZ (N+2)	ZZZ (2N+2)	ZZZ (3N+2)	ZZZ (4N+2)	ZZZ (5N+2)	ZZZ (6N+2)
"	"	"	"	"	"	"
"	"	"	"	"	"	"
ZZZ (N)	ZZZ (2N)	ZZZ (3N)	ZZZ (4N)	ZZZ (5N)	ZZZ (6N)	ZZZ (7N)

(a)

NUMPT (1)	XPT (1)	YPT (1)	ZPT (1)	UPT (1)	VPT (1)	WPT (1)
NUMPT (2)	XPT (2)	YPT (2)	ZPT (2)	UPT (2)	VPT (2)	WPT (2)
"	"	"	"	"	"	"
"	"	"	"	"	"	"
NUMPT (N)	XPT (N)	YPT (N)	ZPT (N)	UPT (N)	VPT (N)	WPT (N)

(b)

1	X1	Y1	Z1	U1	V1	W1
2	X2	Y2	Z2	U2	V2	W2
"	"	"	"	"	"	"
"	"	"	"	"	"	"
N	XN	YN	ZN	UN	VN	WN

(c)

Figure 3. Nodal point and displacement storage arrays.
 N = The number of nodes. (a) Array in subroutine PSAP1.
 (b) Arrays in subroutines called by PSAP1. (c) Nodal
 coordinates and displacements in (a) and (b).

KDATA = 1 or 5). When read, displacement data is stored in the last three columns of array ZZZ (figure 3). PSAP1 can postprocess displacements for both ADINA and SAP IV. The difficulty comes in obtaining a punched deck of cards. Reference 6 gives a description of how to obtain a deck of cards for SAP IV in a format acceptable to SUBROUTINE DATA9. ADINA has no such provision. However, when preprocessing, the displacement data will be omitted (NUDISP=NVDISP=NWDISP=0), and this step will be by-passed.

F. PLOTTING LOGIC

SUBROUTINE PLOTX (figure 4) is the main plotting routine. Since the nodal point data is stored in array ZZZ and the connectivity is stored on device 10, it is a simple matter to read the connectivity from device 10 (one element at a time), and connect the nodes as they are defined in references 1 and 2. For example, NEND = number of nodes defining the connectivity of a single element, NUMEL = the element number and NODE(NEND) is the array containing the connectivity. Device 10 contains this information successively for each element. The statement

```
READ(10) NEND,NUMEL,(NODE(I),I=1,NEND)
```

will read the element connectivity to be plotted. The 8-20 node brick (ADINA and SAP IV) and the 4-8 node plane elements (ADINA) may have 3 points defining each edge. If the midpoint node is defined, then isoparametric

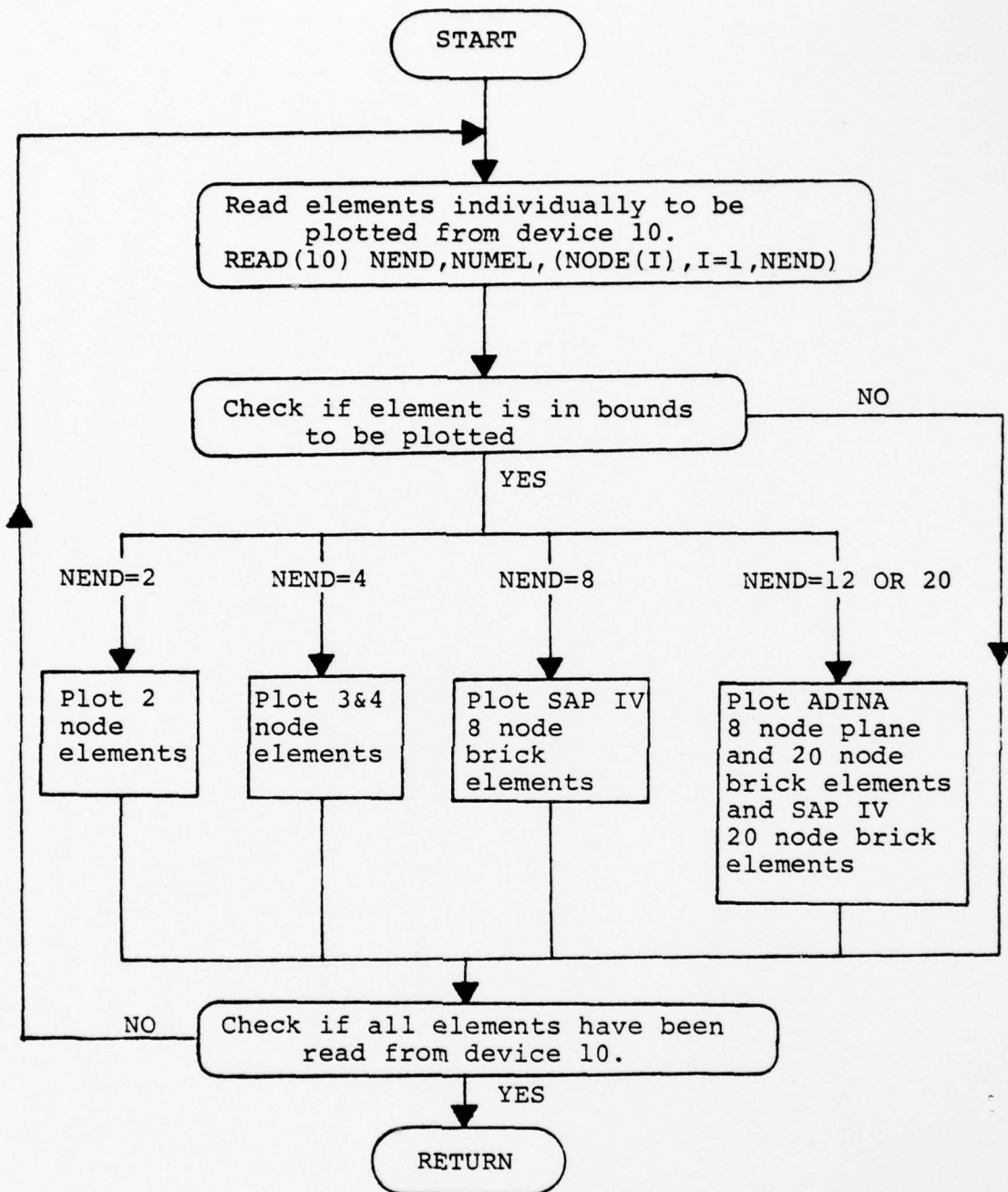


Figure 4. Flow chart for PSAP1 plotting subroutine, PLOTX. Array NODE contains the connectivity of the element being plotted. NEND = the number of nodes defining the connectivity of a single element. NUMEL = the element number.

shape functions (reference 3) are used to interpolate along the three-node edges. These shape functions are identical to those used by ADINA and SAP IV, so the geometry represented graphically is identical to the problem solved in ADINA and SAP IV. SUBROUTINE CURVE does the interpolation with the following equations:

$$X_o = N1 * X_{01} + N2 * X_{02} + N3 * X_{03}$$

$$Y_o = N1 * Y_{01} + N2 * Y_{02} + N3 * Y_{03}$$

$$N1 = S * (S - 1.0) / 2.0$$

$$N2 = - (S + 1.0) * (S - 1.0)$$

$$N3 = S * (S + 1.0) / 2.0$$

$$-1.0 \leq S \leq 1.0$$

PSAP1 uses the NPS plotting package [reference 5] subroutines.

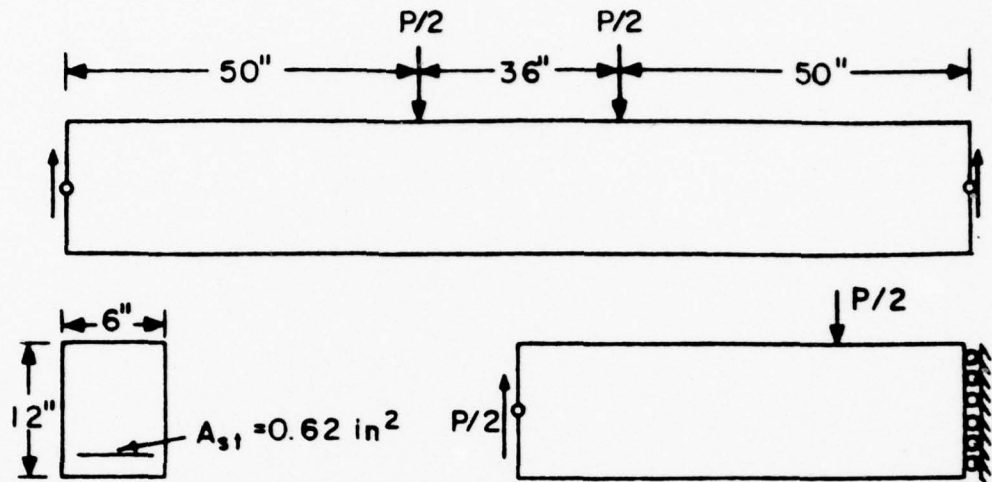
III. PSAP1 SAMPLE PROBLEMS

The following examples have been chosen to illustrate some of the most useful options of PSAP1. Prior to attempting to use PSAP1, the user should have the problem defined and the cards punched in the format of references 1 and 2. Appendix A of this thesis gives a complete description of deck preparation for PSAP1 here at NPS. This section should prove helpful in the understanding and interpretation of the options presented in Appendix A.

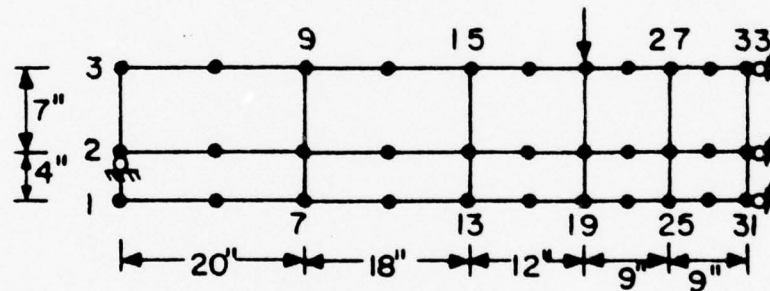
A. ADINA EXAMPLES

1. Reinforced Concrete Beam (example 1, figure 5)

This example was chosen because it illustrates the importance of the exploded plot when more than one element group is used. It is taken from the ADINA manual [reference 1]. Figure 6 is a listing of the data cards as they are prepared for ADINA. Figure 7 indicates how that deck would be modified for PSAP1. Note: load cards are removed, NAMELIST OPTION and NAMELIST PICT are added, and the title to be plotted on the graph is added in figure 7. Otherwise, figures 6 and 7 are the same. Figure 8, part (a), illustrates an undistorted ($KDISP = 0$) PSAP1 plot with the nodes numbered ($NOTAT = 1$). Figure 8, part (b), shows the same mesh in an exploded form ($KDISP = 2$) with the elements numbered ($NOTAT = 2$). Note how the truss elements are visible in the exploded plot.



BEAM DIMENSIONS



FINITE ELEMENT IDEALIZATION

MATERIAL PROPERTIES:

- $\sigma_c = 3740$ psi
- $\sigma_t = 458$ psi
- $\sigma_{y\text{steel}} = 44000$ psi
- $E_{o\text{concrete}} = 6100$ ksi
- $\nu = 0.2$
- $E_{\text{steel}} = 30000$ ksi
- $E'_{\text{steel}} = 300$ ksi

Figure 5. Example 1, ADINA truss and 8 node plane elements, Reinforced Concrete Beam (Given on page 84, reference 2).


```

1 10 1 0
0.620 .000733863 0.300.
30000. 44.0 1 3 0.
1 4 1 3 0.
10 28 31 1 0.
*****
2 10 1 0
1.000217164
6100. 0.20
0.458 -3.74
1 6 1 1 0
8 2 1 1 0
5 6 1 1 0
32 26 25 1 0
6 6 1 1 0
9 3 2 1 0
10 27 1 1 0
33 27 26 1 1 0
*****
1 9 0.0 1.0
0.4.0 8.0 5.0
8.0 13.5
21 3 1 -0.5
*****
TRUSS ELEMENT INPUT 3 1 3
1 3 0.
1 0 0.
2D CONTINUUM ELEMENT INPUT 5 1 0 0 0
3 3
0.0005 0.50
0 4 0 6.
0 28 0 6.
0 5 0 6.
0 29 0 6.
APPLIED LOAD DATA 4.8 3.0 6.4
9.5 3.2 7.0 12.5
6.0 2.0
8.0 6.0
11.0

```

Figure 6. Example 1, ADINA input deck listing, page 2 of 2.
 ***** Not part of input deck.

```

***** PSAPI TITLE TO BE PLOTTED ON GRAPH
NONLINEAR ANALYSIS OF A REINFORCED CONCRETE BEAM (ADINA EX)
*****
      NAMELIST OPTION
&OPTION
KGECM=1,
NNDESI=33,
YSPACE=10.0,
&END

```

```

***** TITLE CARD ADINA EXAMPLE
NONLINEAR ANALYSIS OF A REINFORCED CONCRETE BEAM

```

```

33100111 0 2 1 2 1 0 1 0
                MASTER CONTROL CARDS
                BLANK CARD
                BLANK CARD
                BLANK CARD

```

25

```

BLANK CARD
BLANK CARD
BLANK CARD

```

```

1 4 7 13 19 25 31 2 5 8 14 20 26 32 3 9 15 21 27 33
0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
*****
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
NODAL POINT DATA
0. 0. 10. 20. 38. 50. 59. 68. 0. 10. 20. 38. 50. 59. 68. 0.
3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3
4. 4. 4. 4. 4. 4. 11. 11. 11. 11. 11.

```

```

*****
LOAD CONTROL CARD

```

20

**** Not part of input deck.

Figure 7. Example 1, PSAPI input deck listing, page 1 of 2.

```

*****
1
1 10 1 0
0.620 .000733963 0.300.
30000. 44.0 1 0.
1 28 31 1 0.
10
*****
2 10 1 0
1.000217164
6100. 0.20
0.458 -3.74
1 8 6 2 1 1 1
5 2 6 6 1 1 1
32 26 25 1 2 1 1
9 3 3 2 1 1 1
10 6 27 26
33
*****
2
1 7 6
1 5 6
1 1 29 6
31 1 6 6
8 6 6
1 1 6 30
32
*****
3
1 3 0.
1 0 0.
*****
6
2D CONTINUUM ELEMENT INPUT
0 5 1 0 0 0
0.0005 0.50
0 4 0 6.
0 28 0 6.
0 5 0 6.
0 29 0 6.
*****
APPLIED LOAD DATA IS REMOVED
NAMELIST PICT (UNDEFORMED STRUCTURE)
*****
NAMELIST PICT (EXPLODED PLOT)
*****
DMAGS=0.7,
KDISP=2,
KDOF=0,
NOTAT=2,
PLOTSZ=9.0,
&END

```

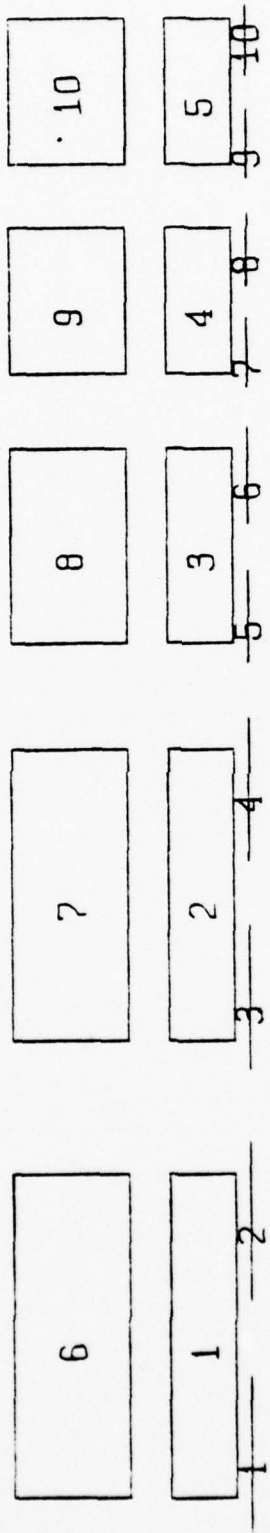
**** Not part of input deck

Figure 7. Example 1, PSAPI input deck listing, page 2 of 2.

3	6	9	12	15	18	21	24	27	30	33
2	5	8	11	14	17	20	23	26	29	32
1	4	7	10	13	16	19	22	25	28	31

(a) Undeformed structure (KDISP=0), nodes numbered (NOTAT=1).

Figure 8. Example 1, PSAP1 output graphs, page 1 of 2.



(b) Exploded plot (KDISP=2), elements numbered (NOTAT=2).

Figure 8. Continued, page 2 of 2.

2. Flat Plate With Hole (example 2, figure 9)

This is a well known problem with which one can calculate the stress concentration on a hole in a plate under axial tension. Figure 10 is a listing of the PSAP1 deck set-up. The mesh is composed of ADINA variable 4-8 node plane elements. Notice on figure 11, parts (a) and (b), how the interpolating shape functions round off the 3-node edges. Part (a) has the nodes numbered (NOTAT = 1). Part (b) has the elements numbered (NOTAT = 2), and illustrates the use of the symmetry option (KSYMXY=KSYMxz=1). The symmetry option enables one to obtain a picture of the complete plate even though the model only consisted of a quarter plate with proper boundary conditions.

B. SAP IV EXAMPLES

1. SAP IV Truss Problem (example 3, figure 12)

Figure 13 is a listing of the PSAP1 data deck. Figure 14 indicates how multiple partial plots can be used to obtain a better representation of the model. Part (a) of figure 14 is the complete model. Part (b) is the left half (XXMAX = 50'), and part (c) is the right half (XXMIN = 50', XXMAX = 1.0E20'). Notice also that for Parts (b) and (c), ISCALE = 0, which means succeeding plots have the same scale as the first. Had ISCALE equaled 1 (blow-up), then the width of the half view would have been the same as the complete model. Figure 14 size is limited by the NAMELIST PICT variable PLOTSZ.

Figure 9. Example 2, flat plate with a hole in tension. ADINA 4-8 node plane element.

Thickness = 1 inch

Young's modulus = 30.0×10^6 psi

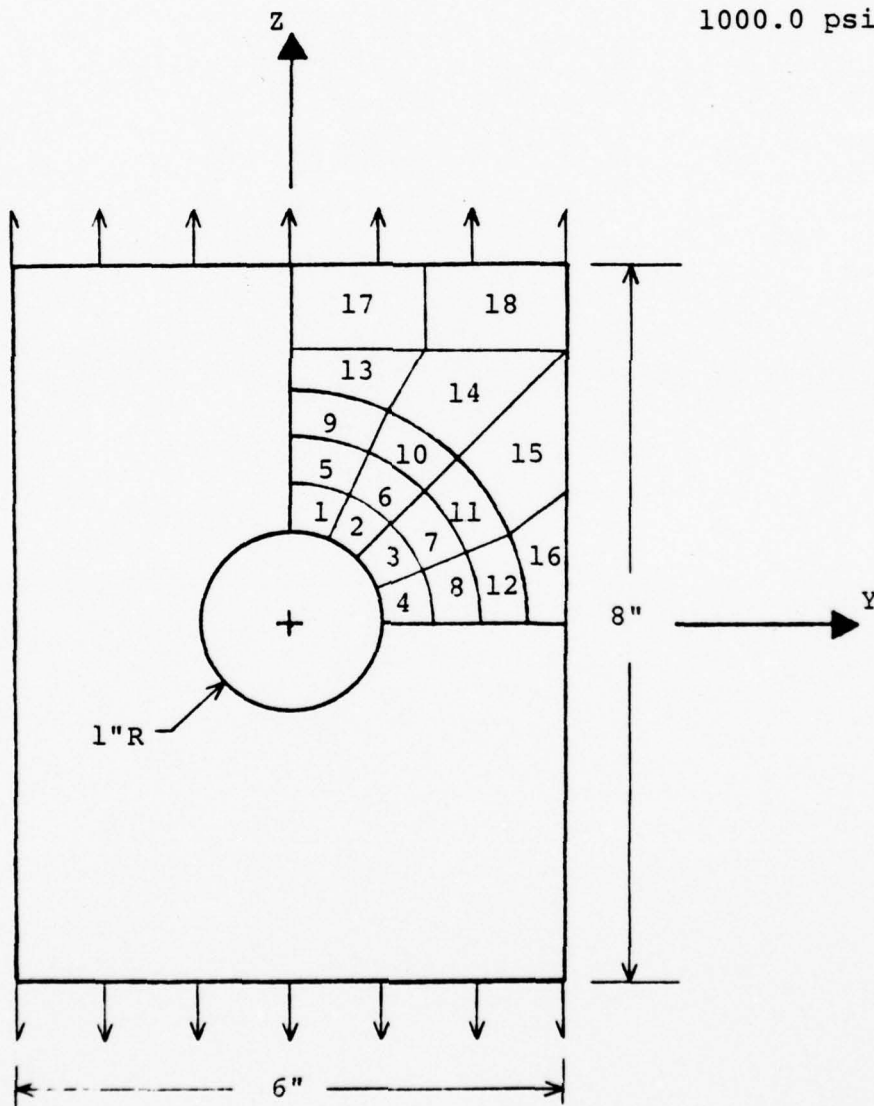
Poisson's ratio = .333

Total elements = 18

Total nodes = 44

Distributed Load =

1000.0 psi



KIBLER AE FLAT PLATE TESTING GEOM1 AND ADPLAN 18 ELEMENTS PSAPI TITLE CARD TO BE PLOTTED ON GRAPH

OPTION
KGEOM=1
NNDST=44
YSPACE=5.0
\$END
NAMELIST OPTION

KIBLER AE FLAT PLATE PROBLEM IN ADINA 8 NODE PLANE WITH DATA GENERATION

44100111 1 0 1 0 0
0 0 1.0 0.0 1 0 0
0 0 0 0
1

BLANK CARD
BLANK CARD

	NODAL	POINT DATA	
X 37	1	0.0	90.0
X 29	1	0.0	90.0
X 3	1	0.0	78.75
X 30	1	0.0	78.75
X 4	1	0.0	67.50
X 31	1	0.0	67.50
X 5	1	0.0	56.25
X 32	1	0.0	56.25
X 6	1	0.0	45.0
X 33	1	0.0	45.0
X 7	1	0.0	33.75
X 34	1	0.0	33.75
X 8	1	0.0	22.50
X 35	1	0.0	22.50
X 9	1	0.0	11.25
X 36	1	0.0	11.25

Figure 10. Example 2, PSAPI input deck listing, page 1 of 3.

***** Not part of input deck.

38	1	0	0	1	1	0.0	1.5	3.0
39	1	0	0	1	1	0.0	3.0	3.0
40	1	0	0	1	1	0.0	3.0	1.5
41	1	0	1	1	1	0.0	3.0	0.0
42	1	0	1	1	1	0.0	0.0	4.0
43	1	0	0	1	1	0.0	1.5	4.0
44	1	0	0	1	1	0.0	3.0	4.0

3	1	2	0	****	LOAD CONTROL CARD			
0	1			****	INITIAL CONDITIONS			

2	18	0	0	****	6	2D CONTINUUM ELEMENT	INPUT	1	1	0	0	0
1	30.0F06	0	0	2								
1	1	1	1	1	1	1	1	1	1	1	1	1
4	7	9	6	2	2	0	11	0	1.0	0	0	0
5	10	12	6	11	8	0	17	0	1.0	0	0	0
8	16	18	6	17	2	0	20	0	1.0	0	0	0
15	19	21	6	25	11	0	26	0	1.0	0	0	0
16	18	27	6	17	17	0	29	0	1.0	0	0	0
19	21	30	6	20	2	0	35	0	1.0	0	0	0
12	25	27	6	34	26	0		0	1.0	0	0	0
13	28	36	6	1	29	0		0	1.0	0	0	0
14	30	38	6	37	31	0		0	1.0	0	0	0
15	32	39	6	38	33	0		0	1.0	0	0	0
16	34	40	6	39	35	0		0	1.0	0	0	0
17	36	41	6	40		0		0	1.0	0	0	0
17	37	41	6	41		0		0	1.0	0	0	0
18	38	43	6	42		0		0	1.0	0	0	0
18	39	44	6	43		0		0	1.0	0	0	0
				44								

Figure 10. Example 2, PSAP1 input deck listing, page 2 of 3.

**** Not part of input deck.

```
***** APPLIED LOAD DATA IS REMOVED
***** NAMELIST PICT (ACTUAL STRUCTURE)
```

```
&PICT
KHCRZ=2,
KVERT=3,
NDIAT=1,
PLOTSZ=7.6,
ISCALE=1,
KODE=2,
&END
```

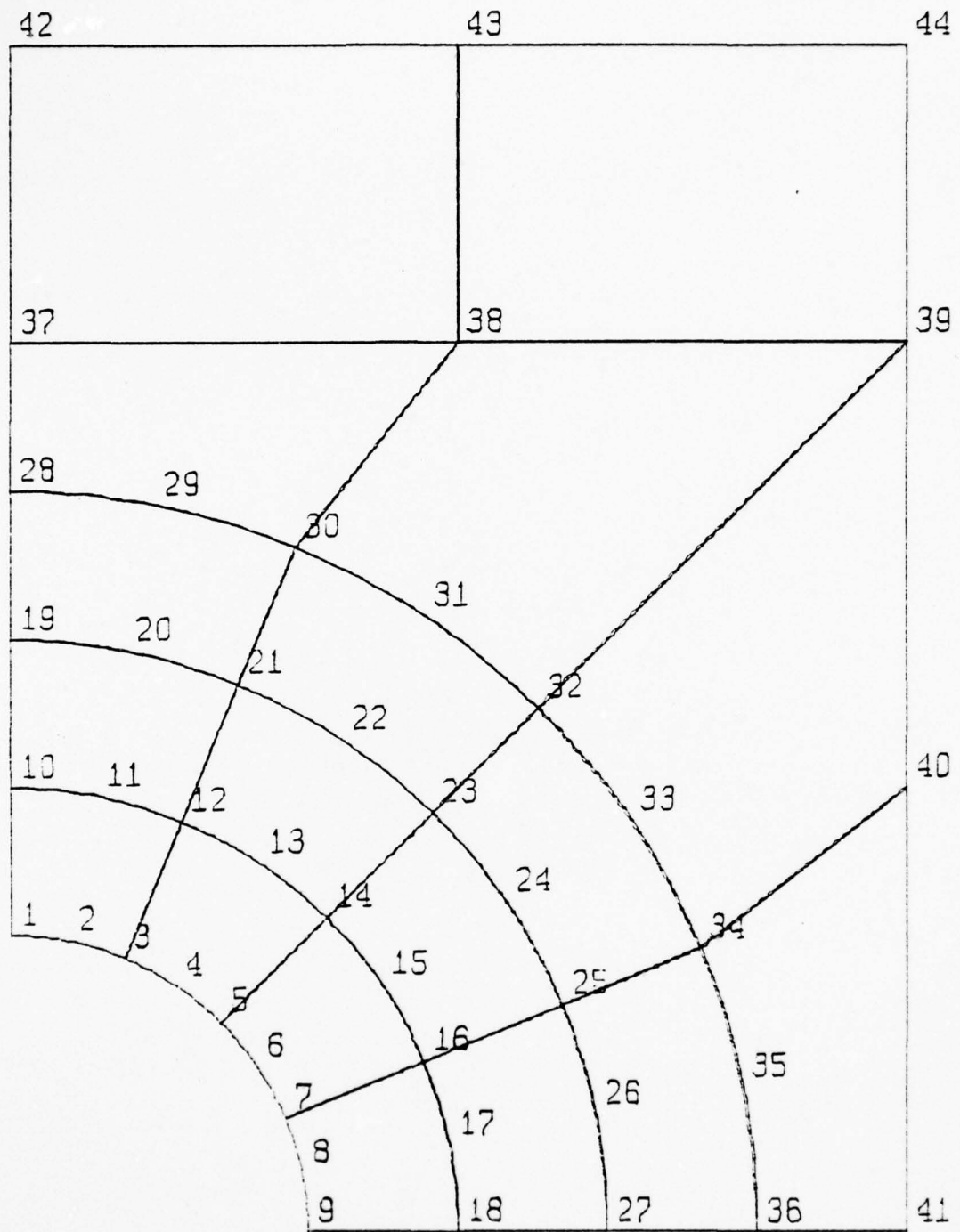
```
***** NAMELIST PICT (SYMMETRIC REPRESENTATION)
```

```
&PICT
PLOTSZ=8.0
NDIAT=2,
KSYMXX=1,
KSYMXY=1,
KCODE=0,
&END
```

Figure 10. Example 2, PSAP1 input deck listing, page 3 of 3.

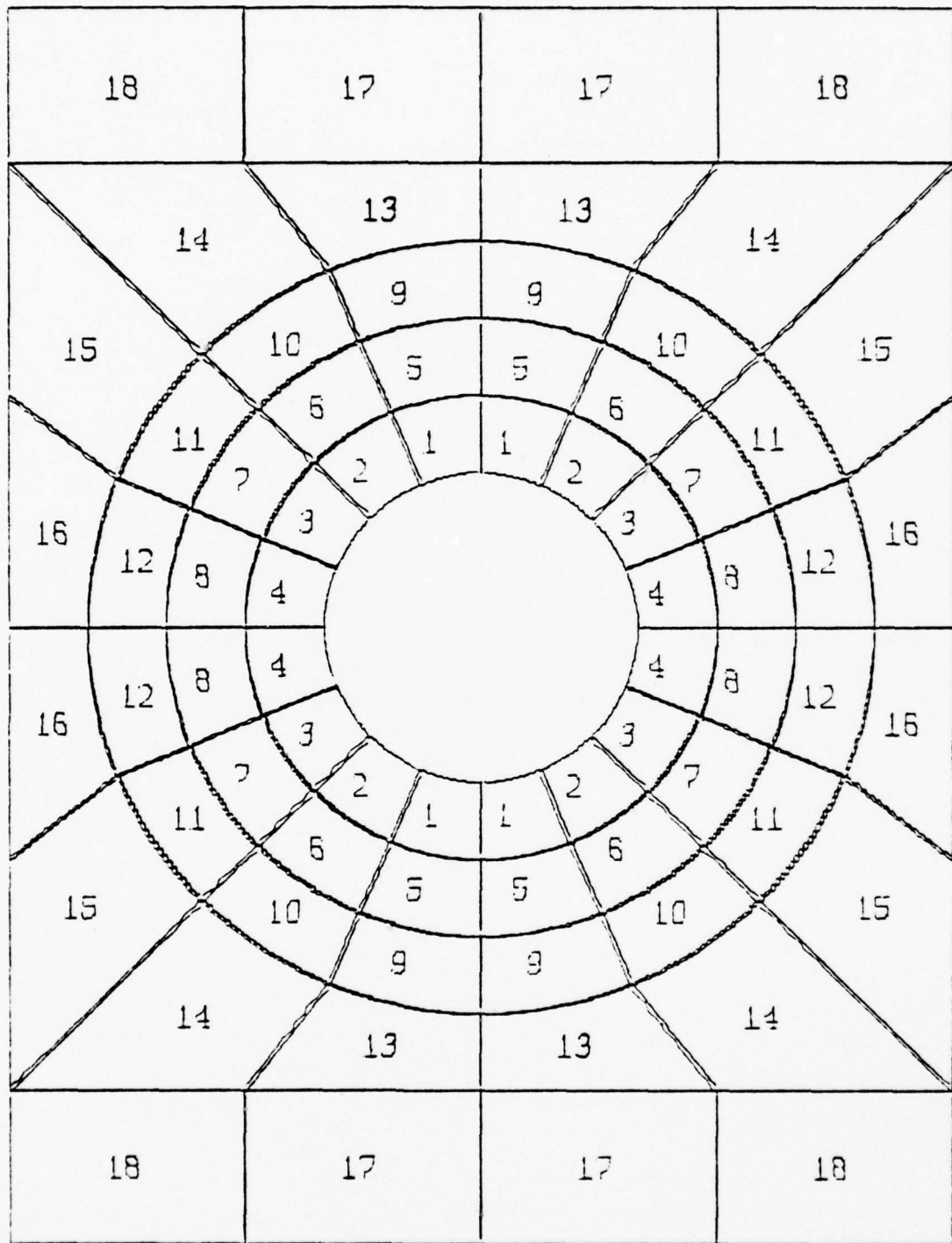
***** Not part of input deck.

Figure 11. Example 2, PSAP1 output graphs, page 1 of 2.



(a) Actual mesh, nodes numbered.

Figure 11. Continued, page 2 of 2.



(b) Symmetric representation, elements numbered.

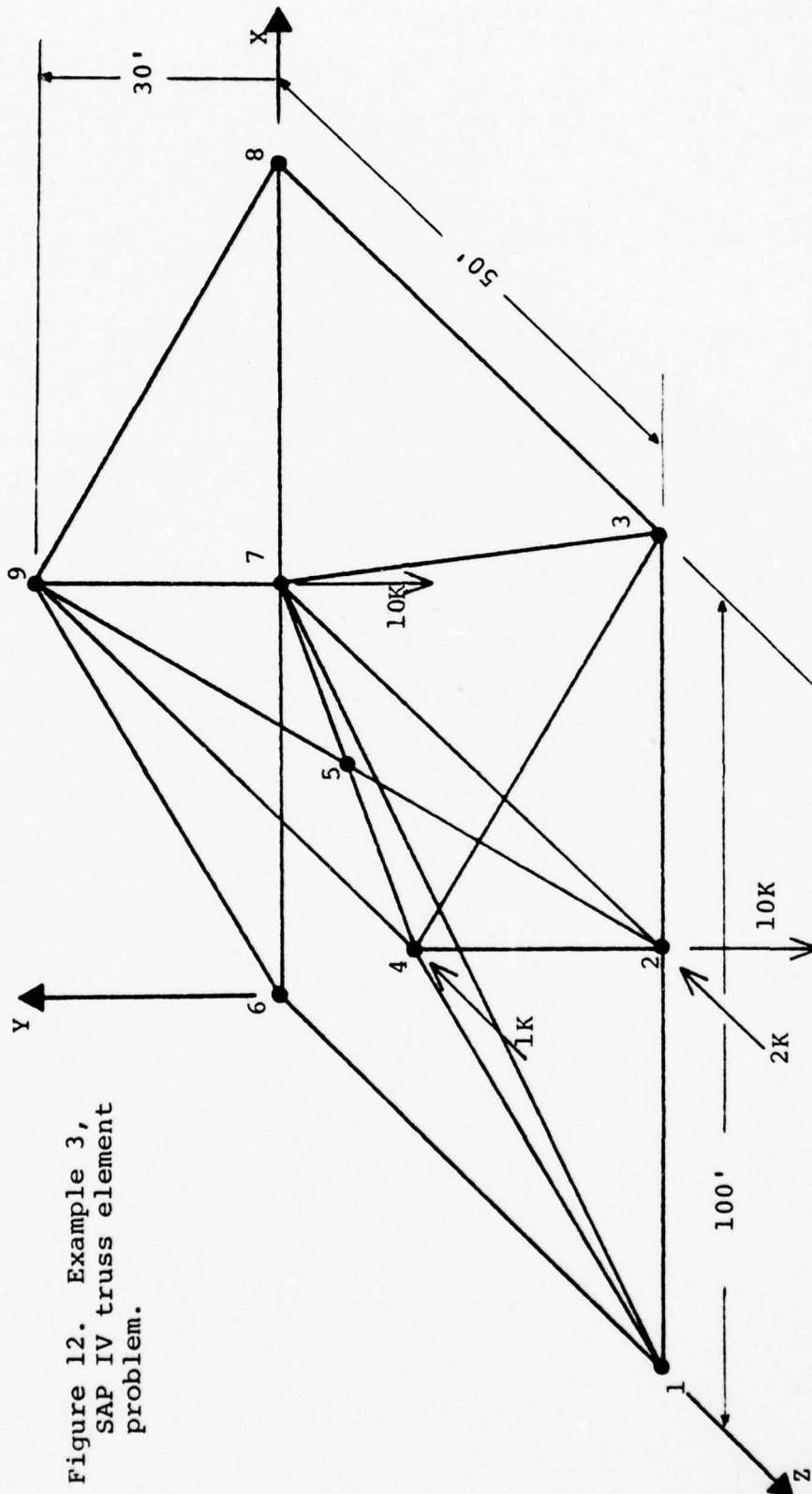


Figure 12. Example 3, SAP IV truss element problem.

Young's modulus = 4.32×10^9 LB/SQFT
 Cross-sectional area = 7.0×10^{-2} SQFT
 Weight density = 4.89×10^2 LB/SQFT

BOUNDARY CONDITIONS

NODE	X	Y	Z
1	fixed	fixed	free
3	free	fixed	free
6	fixed	fixed	free
8	free	fixed	free

All others are free.


```

10 70.0
11 70.0
12 70.0
13 70.0
14 70.0
15 70.0
16 70.0
17 70.0
18 70.0
19 70.0
20 70.0

```

LOAD,LOAD CASE MULTIPLIER AND DYNAMIC ANALYSIS
CARDS REMOVED


```

1 1
2 1
3 1
4 1
5 1
6 1
7 1
8 1
9 1

```

NAMELIST PICT FOR TCTAL MESH

```

&PICT
KHCR7=1,
KVERT=2,
PSI=-20.0,
PHI=25.0,
THETA=-40.0,
PLCTSZ=4.5,
NGTAT=1
KCODE=1,
&ENC

```

NAMELIST PICT (LEFT HAND SIDE)

```

&PICT
XXMAX=50.0,
ISCALE=C,
&ENC

```

NAMELIST PICT (RIGHT HAND SIDE)

```

&PICT
XXMAX=1.0E2J,
XXMIN=50.0,
KCODE=C,
&ENC

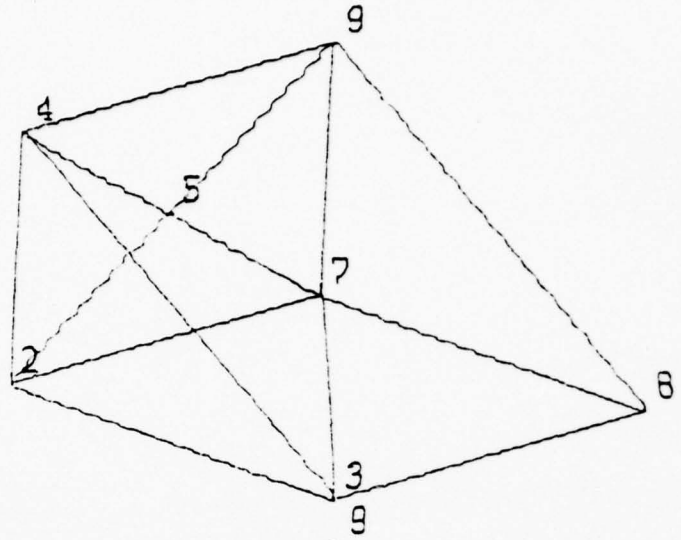
```

Figure 13. Example 3, PSAP1 input deck listing, page 2 of 2.

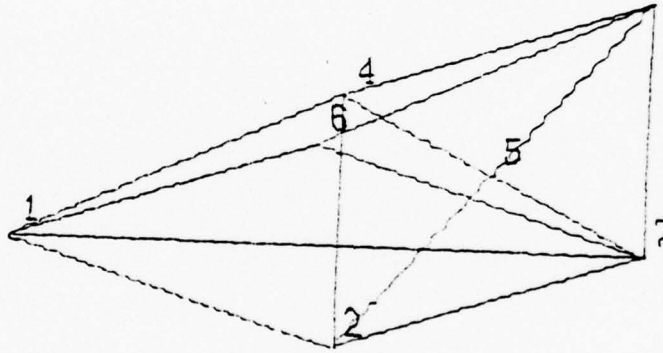
**** Not part of input deck.

Figure 14. Example 3, PSAP1 output graphs.

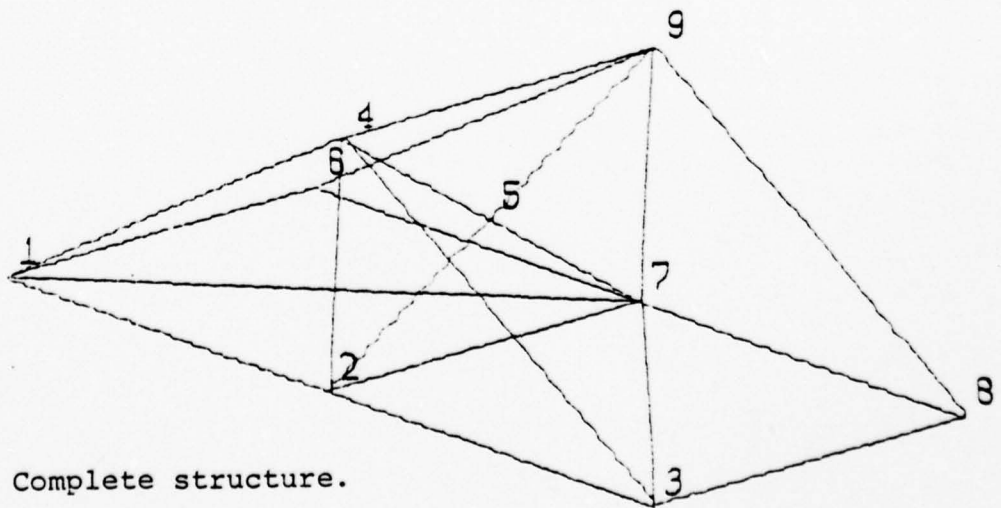
(c) Right hand side.



(b) Left hand side.

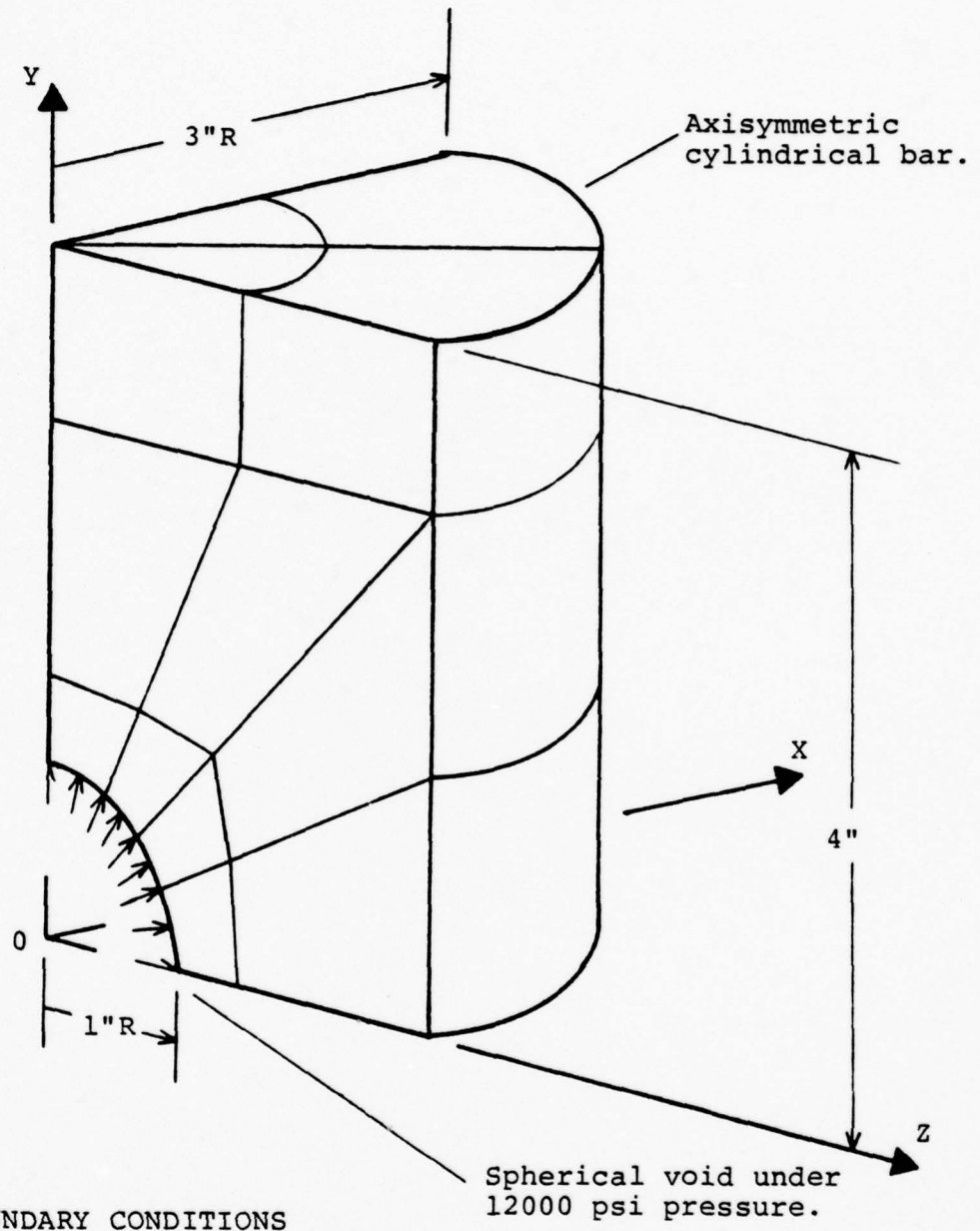


(a) Complete structure.



2. Cylindrical Bar With Spherical Hole (example 4, figure 15)

This mesh could have several uses. Two might be to calculate stress concentrations if the bar is under axial load or to calculate loading if the void is under pressure (i.e., dispersed nuclear fuel pellet). Figure 16 is a listing of the PSAP1 data deck. Figure 17, part (a), is a representation of the complete model. Parts (b), (c), (d), and (e) of figure 17 are partial plots of the total structure using options of the undeformed structure (KDISP = 0) with node numbering (NOTAT = 1), and the exploded plot (KDISP = 2) with element numbering (NOTAT = 2).



BOUNDARY CONDITIONS

- X fixed on YZ plane ($X = 0$)
- Y fixed on XZ plane ($Y = 0$)
- Z fixed on XY plane ($Z = 0$)

Young's modulus = 30.0×10^6 psi

Poisson's ratio = .333

Figure 15. Example 4, six inch diameter cylindrical bar with a two inch diameter spherical void on the center line under pressure.

100	92	94	102	54	46	48	56	97	93	98	101	51	47	52	55
74	71	72	75	0	0	0	0	0	0	0	0	0	0	0	0
108	20	102	110	62	54	56	64	105	101	106	109	59	55	60	63
77	74	75	78	0	0	0	0	0	0	0	0	0	0	0	0
9	14	0	1	84	80	80	86	114	115	118	118	81	82	85	85
117	113	113	119	0	0	0	0	0	0	0	0	0	0	0	0
10	16	0	1	0	0	0	0	0	0	0	0	0	0	0	0
125	117	119	127	92	84	86	94	122	118	123	126	89	85	90	93
11	14	0	1	0	0	0	0	0	0	0	0	0	0	0	0
40	34	34	42	7	1	1	9	36	37	41	41	3	4	4	8
12	16	0	1	0	0	0	0	0	0	0	0	0	0	0	0
48	40	42	50	15	7	9	17	44	41	45	49	11	8	12	16
13	16	0	1	0	0	0	0	0	0	0	0	0	0	0	0
56	48	50	58	23	15	17	25	52	49	53	57	19	16	20	24
14	16	0	1	0	0	0	0	0	0	0	0	0	0	0	0
64	56	58	66	31	23	25	33	60	57	61	65	27	24	28	32
15	18	0	1	0	0	0	0	0	0	0	0	0	0	0	0
86	80	80	88	40	34	34	42	82	83	87	87	36	37	41	41
69	67	67	70	0	0	0	0	0	0	0	0	0	0	0	0
16	20	0	1	0	0	0	0	0	0	0	0	0	0	0	0
54	86	88	96	48	40	42	50	90	87	91	95	44	41	45	49
72	69	70	73	0	0	0	0	0	0	0	0	0	0	0	0
17	20	56	104	0	0	0	0	0	0	0	0	0	0	0	0
102	54	56	76	56	48	50	58	98	95	99	103	52	49	53	57
75	72	73	76	0	0	0	0	0	0	0	0	0	0	0	0
118	20	104	112	64	56	58	66	106	103	107	111	60	57	61	65
110	75	76	79	0	0	0	0	0	0	0	0	0	0	0	0
178	14	0	1	0	0	0	0	0	0	0	0	0	0	0	0
115	113	113	121	86	80	80	88	115	116	120	120	82	83	87	87
20	16	0	1	0	0	0	0	0	0	0	0	0	0	0	0
127	119	121	129	94	86	88	96	123	120	124	128	90	87	91	95

Figure 16. Continued, page 5 of 6. ***** Not part of input deck.

***** NAMELIST PICT FOR TCTAL STRUCTURE

```
EPICIT=2,  
KHCRZ=3,  
KVERT=3,  
PHI=10.C,0,  
THETA=10.0,  
PSI=45.0,  
ISCALE=2,  
XCRGN=C.4,  
PSCALE=0.5714,  
KODE=1,  
&ENC
```

***** NAMELIST PICT, ELEMENTS 1-10, NODES NUMBERED

```
EPICIT=2,  
ISCALE=C,  
NCTAT=1,  
XLFT=C.1,  
NELMAX=10,  
&ENC
```

***** NAMELIST PICT, ELEMENTS 1-10, EXPLODED PLOT

```
EPICIT=2,  
NOTAT=2,  
KDISP=2,  
DMAGS=C.5,  
XLFT=0.151,  
&ENC
```

***** NAMELIST PICT, ELEMENTS 11-20, NODES NUMBERED

```
EPICIT=1,  
NOTAT=C.1,  
XLFT=C.1,  
KDISP=C,  
NELMIN=11,  
NELMAX=20,  
&ENC
```

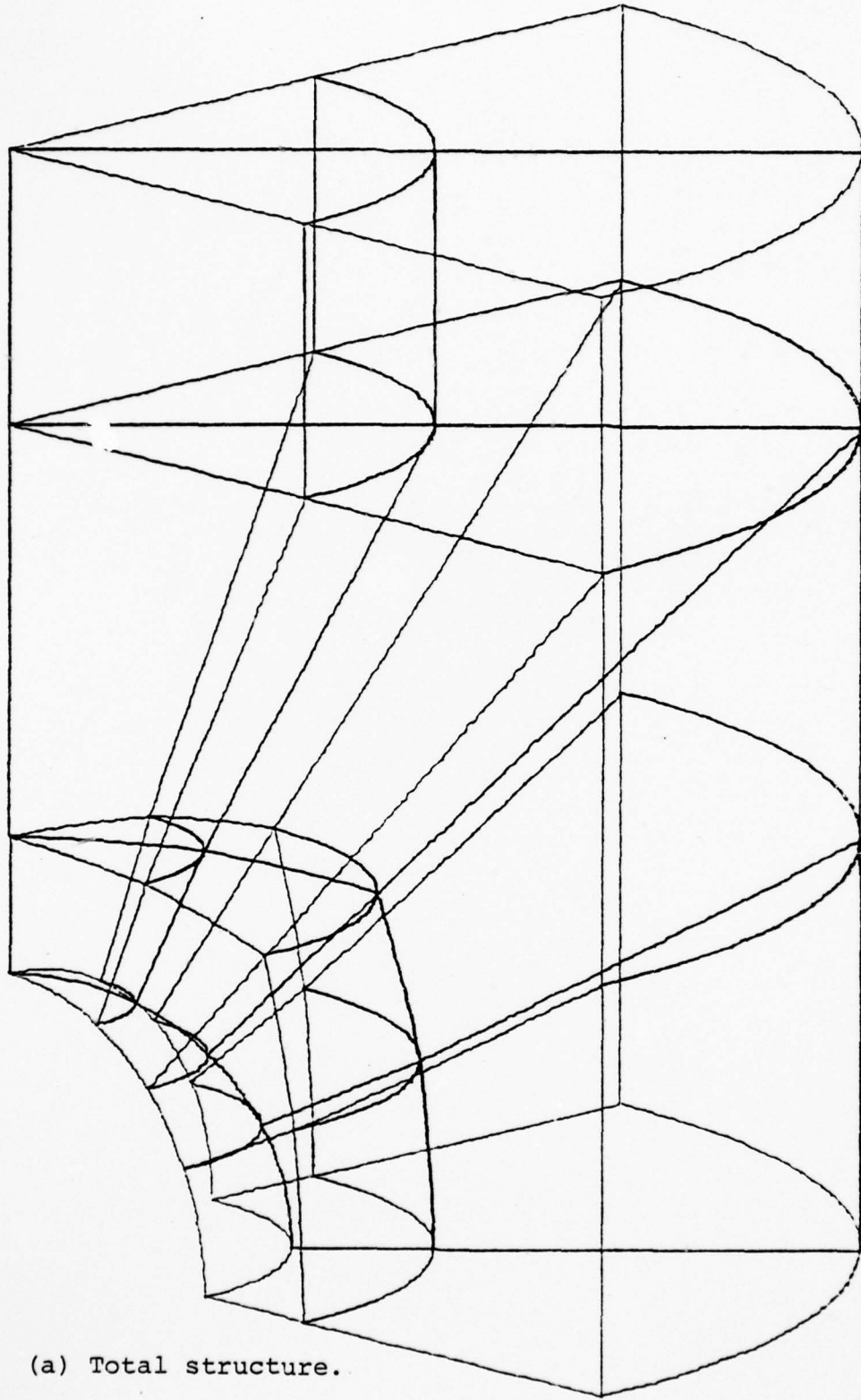
***** NAMELIST PICT, ELEMENTS 11-20, EXPLODED PLOT

```
EPICIT=2,  
NOTAT=2,  
KDISP=2,  
DMAGS=C.5,  
XLFT=0.151,  
KODE=0,  
&ENC
```

Figure 16. Continued, page 6 of 6.

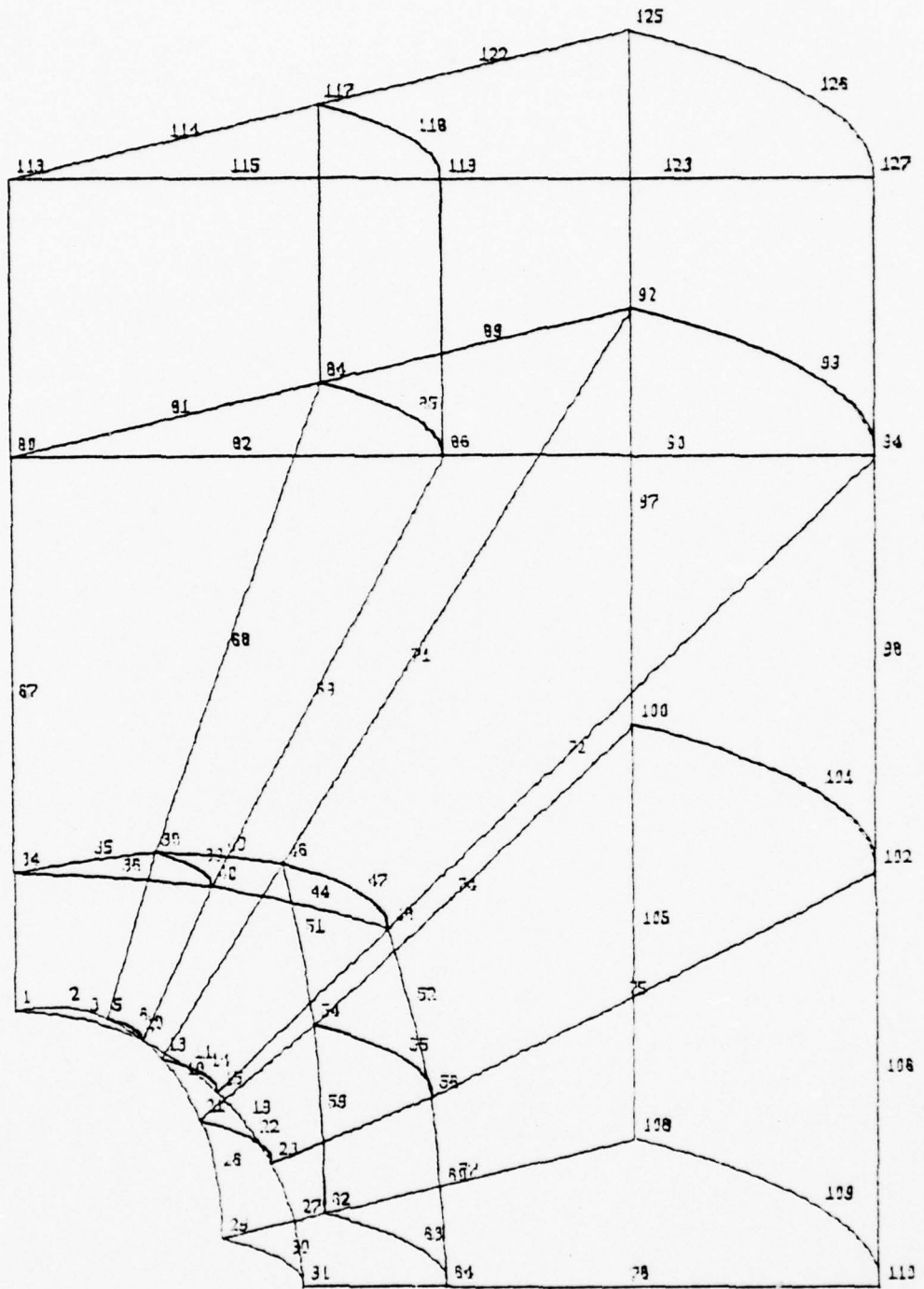
***** Not part of input deck.

Figure 17. Example 4, PSAP1 output graphs, page 1 of 5.



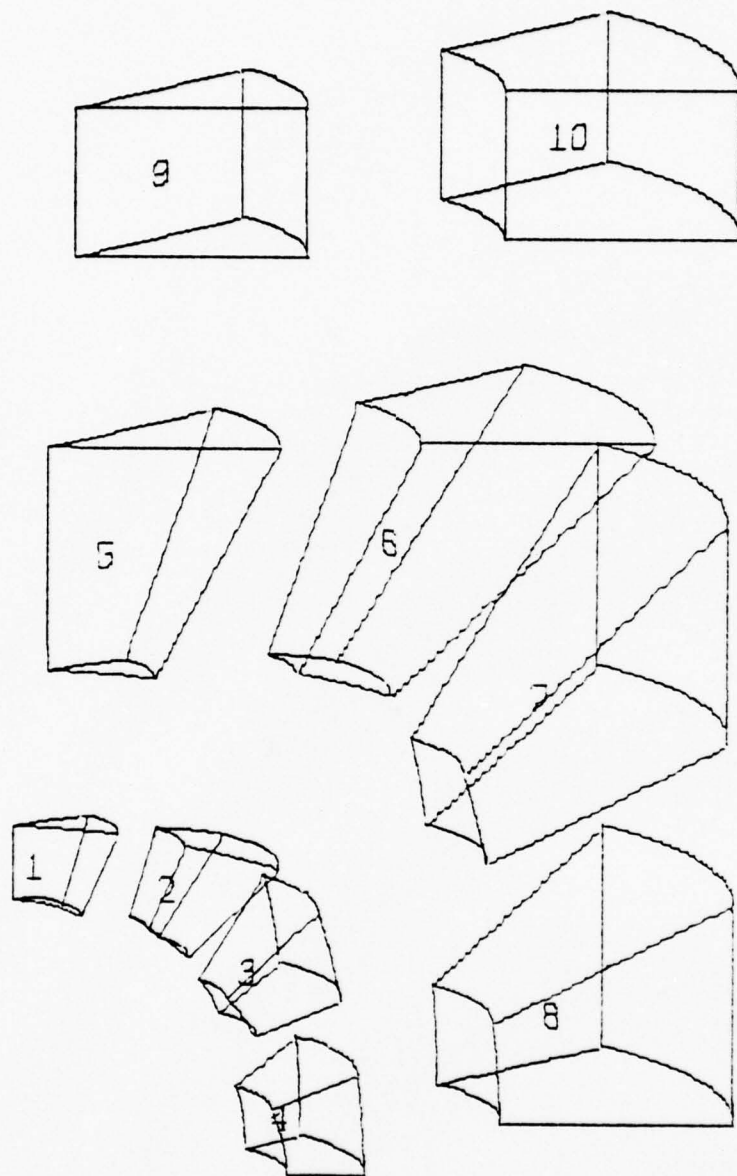
(a) Total structure.

Figure 17. Continued, page 2 of 5.



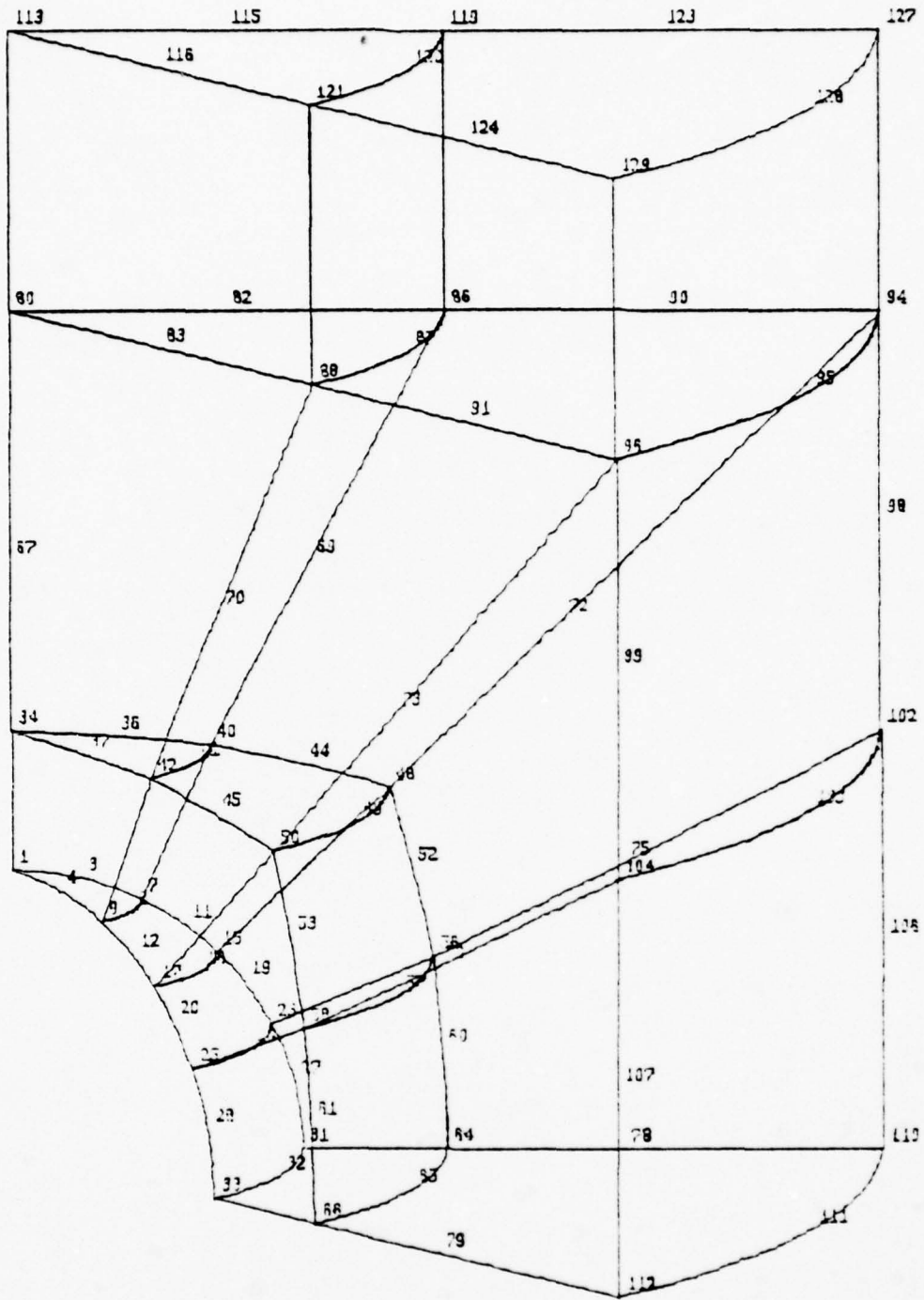
(b) Elements 1-10, nodes numbered.

Figure 17. Continued, page 3 of 5.



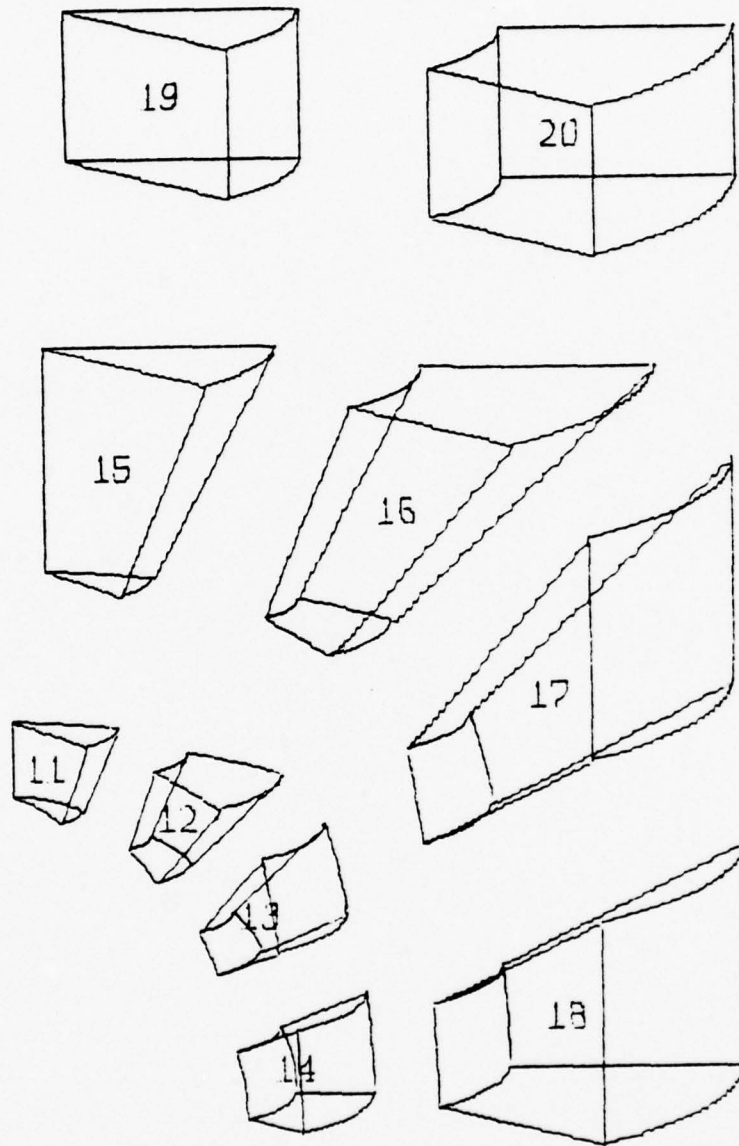
(c) Elements 1-10, elements numbered, exploded plot.

Figure 17. Continued, page 4 of 5.



(d) Elements 11-20, nodes numbered.

Figure 17. Continued, page 5 of 5.



(e) Elements 11-20, elements numbered, exploded plot.

IV. CONCLUSIONS AND RECOMMENDATIONS

With a little study of PSAP1 and figure 2, one realizes how simple it would be to expand PSAP1 to include virtually an unlimited number of geometry data input formats. Simply supply the GEOMn subroutines, element subroutines and modify SUBROUTINE ELTYPE if necessary. Another interesting project would be to incorporate PSAP1 into ADINA and SAP IV so that a mesh plot could be obtained in the data check mode only. Reference 4 contains routines for plotting stress contours which could be incorporated. Hidden line logic is available. It would be difficult but could be incorporated. Hidden line plots are pretty, but they are not very practical when preprocessing because in preprocessing it is advantageous to see all of the nodes.

APPENDIX A

PSAP1 USERS MANUAL

I. NAMELISTS

It may be useful for the user to review NAMELISTs in any good FORTRAN manual. However, a short description of the NAMELIST input is given here. Only columns 2 through 80 of each card will be read. The computer expects to find a special delimiter symbol in column 2 of the first card followed by the NAMELIST name. The delimiter symbol for the IBM 360-370 series is the ampersand & ; other processors use the dollar sign \$. Following the first card comes the list of variables and their assigned values. Some important points about the variables are:

- A. Variables and their values need not appear in any special order.
- B. Predefined variables need not appear in the list if it is not necessary to change the value.
- C. A comma should follow each assigned value.
- D. It is recommended that each variable appear on a separate card to facilitate change.

The last card following the NAMELIST variables contains, beginning in column 2, the delimiter sign followed by the word END. For example, &OPTION (read NAMELIST OPTION) contains 12 variables. All are initially assigned default

values. But, suppose three of those values (NNDEST=200, KGEOM=9, YSPACE=2.0) do not apply to a specific problem.

The &OPTION would be read as follows:

```
  &OPTION  
  _NNDEST=300,  
  _KGEOM=1,  
  _YSPACE=5.0,  
  _&END
```

Blank space in column 1.

II. INPUT PROCEDURES

The following sequence of cards is necessary to use PSAP1.

NOTES IBM CARDS

COLUMNS

123456789-----

- (1) // [Standard green job card]
// EXEC FORTCLGP,REGION.GO=150K
//FORT.SYSIN DD *

- (2) C MAIN PROGRAM
DIMENSION ZZZ(NZ),DISPD(5,3,NON)
CALL PSAP1(ZZZ,NZ,DISPD,NON)
STOP
END

- (3) /*
//LINK.USDD DD UNIT=3330,VOL=SER=DISK02,
// DISP=SHR,DSN=S1153.PSAP1
//LINK.SYSIN DD *
INCLUDE USDD(LOADM)
ENTRY MAIN

(4) /*
//GO.FT10F001 DD UNIT=SYSDA,
// SPACE=(CYL,(3,1)),
// DCB=(RECFM=VS,BULKSIZE=3520)

(5) //GO.SYSIN DD *

(6) [PSAP1 title card as it appears on plot]

(7) &OPTION
[&OPTION variables to be initially set or changed]
&END

(8) [ADINA or SAP IV geometry data. Title
card through element data - remove load cards]

(9) [Case identification card - Omit if IDCASE = 0]
[Displacement data cards - Omit for preprocessing
only]

(10) &PICT
[&PICT variables to be initially set or changed]
&END

(11) [Additional data as defined by last value of
KODE in NAMELIST PICT - Omit if last
value of KODE = 0]

(12) /*

Notes:

- (1) Standard basic deck set up as described in chapter 3 of reference 8.

```
// [Standard green job card]
// EXEC FORTCLGP,REGION.GO=150K
//FORT.SYSIN DD *
```

- (2) Main program.

```
DIMENSION ZZZ(NZ),DISPD(5,3,NON)
CALL PSAP1(ZZZ,NZ,DISPD,NON)
STOP
END
```

The main program has two functions: to allocate fast storage space, and to call PSAP1. NON must be greater than the number of nodes. NZ must be greater than 4*NON(7*NON if displacement data cards are to be input. i.e., NUDISP, NVDISP, or NWDISP = 1).

- (3) Using load module library.

```
/*
//LINK.USDD DD UNIT=3330,VOL=SER=DISK02,
// DISP=SHR,DSN=S1153.PSAP1
//LINK.SYSIN DD *
INCLUDE USDD(LOADM)
ENTRY MAIN
```

PSAP1 should be precompiled and stored in the machine in a load module. Load modules are a type of user library described in reference 7 and chapter 3 of reference 8. Since PSAP1 requires over one minute to compile, precompiling results in a large time

savings. These cards are subject to change, and the most current version is determined by the user maintaining this library. If PSAP1 is not on a load module, these control cards may be replaced by subroutine PSAP1 (which consists of a box and a half of cards).

(4) Allocation of storage.

```
/*  
//GO.FT10F001 DD UNIT=SYSDA,  
// SPACE=(CYL,(3,1)),  
// DCB=(RECFM=VS,BULKSIZE=3520)
```

PSAP1 uses a slow storage device to store the element connectivity. These cards allocate 3 cylinders as described in chapter 3 of reference 8.

(5) Deck set up card.

```
//GO.SYSIN DD *
```

This is a standard deck set up card described in chapter 3 of reference 8.

(6) PSAP1 title card.

PSAP1 title card is the title that will appear on the graph. Make sure a user identification is on this card. It consists of 80 alphanumeric characters. The first 40 characters will form the first title line. The last 40 will form the second line.

(7) NAMELIST OPTION.

(Note: start in second column)

```
&OPTION  
[&OPTION variables to be initially set or changed]  
&END
```

VARIABLE-DEFAULT

DESCRIPTION

NNDEST-200	Must be equal to the number of nodes.
NUDISP-0	0 - X direction displacements not input. 1 - X direction displacements input.
NVDISP-0	0 - Y direction displacements not input. 1 - Y direction displacements input.
NWDISP-0	0 - Z direction displacements not input. 1 - Z direction displacements input.

(Note: unless displacement data is to be input,
allow NUDISP, NVDISP, and NWDISP to default.)

KGEOM-9	Specifies the geometry input format. 1- Subroutine GEOM1 reads in ADINA data deck geometry and connectivity. 2- Subroutine GEOM2 may be supplied by user along with subroutines NSPLAN, NS3DEE and NSTRUS to read nodal data and connectivity in any format. 9- Subroutine GEOM9 reads in SAP IV data deck geometry and connectivity.
KDATA-9	Specifies the subroutine and corresponding method of input for displacement data. 1- Subroutine DATA1, a user supplied subroutine.

2- Subroutine DATA2, a user supplied subroutine.

9- Subroutine DATA9, reads a punched output displacement deck from execution of SAP IV as presented in reference 6. (Note: unless displacement data is to be plotted, allow to default.)

NVALUS-0 Not incorporated, allow to default.

IRESEQ-1 Not incorporated, allow to default.

KPLOT-1 Not incorporated, allow to default.

YSPACE-2.0 Space between plots in the Y direction in inches when successive plots are plotted (i.e., $KODE \neq 0$). The graph title is plotted both on the top and at the bottom of each set of graphs controlled by a given NAMELIST OPTION. The space between the title and the plot is $YSPACE/2.0$.

PSIZE-9.0 Paper size in the X direction, in inches. Used in scaling of the plots to insure this dimension is not exceeded. However, when manual scaling ($ISCALE = 2$; see NAMELIST PICT) this protection is not available, and it is possible to exceed the paper width.

IDCASE-0 0- No identification card precedes the deck of displacement values.

 1- Identification card precedes the deck of displacement values.

(8) SAP IV or ADINA data cards.

Here insert the geometry (node coordinates) and element connectivity. This includes the title card through the element data cards; the load cards are removed. For a SAP IV data deck, NAMELIST OPTION variable KGEOM = 9. For an ADINA data deck, KGEOM = 1.

From ADINA deck remove:

1. Applied loads data.
2. Frequency and mode shape calculations data.

From SAP IV deck remove:

1. Concentrated load mass data.
2. Element load multipliers.
3. Dynamic analysis cards.

Otherwise, these cards are exactly the same as the deck prepared for SAP IV or ADINA. PSAP1 is not limited to SAP IV and ADINA. The user may specify any unique format (i.e., KGEOM = 2). In this case, SUBROUTINE GEOM2 must be supplied by the user.

(9) Case ID card and displacement data cards.

PSAP1 is intended to be used essentially for preprocessing. If it is desired to use the postprocessing option, the user is referred to reference 6. Otherwise, omit the case ID card and displacement data.

(10) and (11) NAMELIST PICT.

(Note: start in second column)

```
&PICT  
[&PICT variables to be initially set or changed]  
&END
```

<u>VARIABLE-DEFAULT</u>	<u>DESCRIPTION</u>
KHORZ-1	Integer designating the horizontal axes of the viewing plane. 1 = X_0 . 2 = Y_0 . 3 = Z_0 . (See figure 18.)
KVERT-2	Integer designating the vertical axes of the viewing plane. 1 = X_0 . 2 = Y_0 . 3 = Z_0 . (See figure 18.)
PHI-0.0	Angular rotation of the model about its X axis in degrees (performed 3rd, see figure 18).
THETA-0.0	Angular rotation of the model about its Y axis in degrees (performed 2nd, see figure 18).
PSI-0.0	Angular rotation of the model about its Z axis in degrees (performed 1st, see figure 18).
NEWFR-1	1- Frame change before plotting. 2- No frame change before plotting. (Normally allow to default. A frame change resets the Y origin past the previous plot by YSPACE given in NAMELIST OPTION and resets the X origin at 0.0)
ISCALE-1	0- No scale change. Use the same scale as the previous plot. Useful in an assembly graph where it is desired to examine a mesh in sections

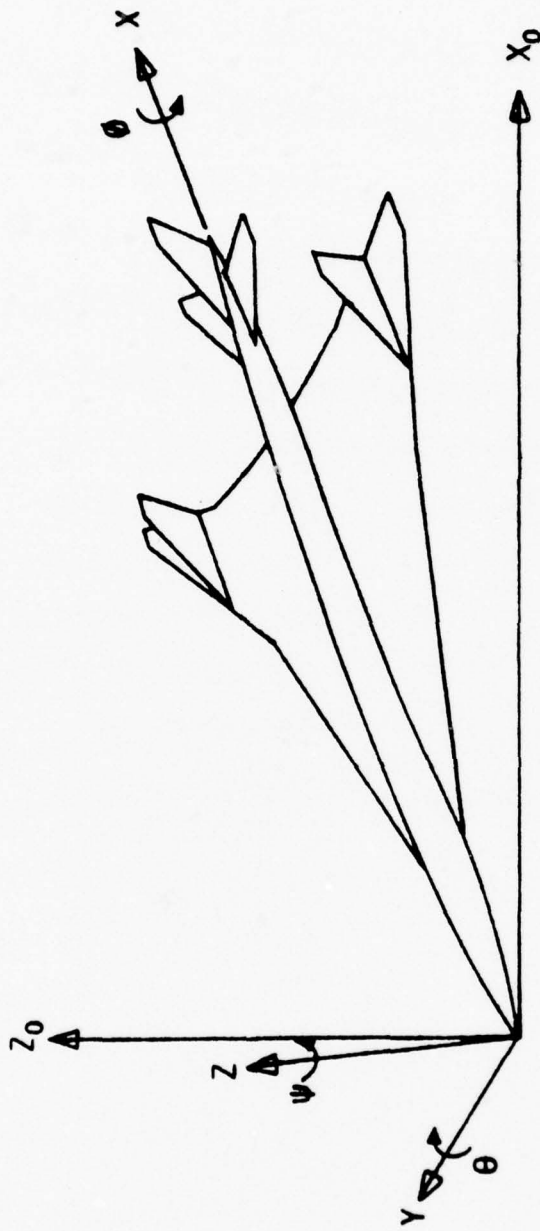


Figure 18. Coordinate systems and Euler angles (rotations) for an oblique orthographic projection shown in X-Z viewing plane. (Taken from reference 4, page 127.)

without losing perspective. See example 3 in Section III of this thesis. ISCALE cannot be zero in the first NAMELIST PICT.

1- Automatic scaling of plot and computation of proper origin location.

2- User specified origin and scaling.

PLOTSZ-10.0 Maximum dimension desired on completed plot prior to rotation. After rotation it is possible for the maximum dimension in the projected plane to exceed PLOTSZ. The maximum size of the projection is unlimited on the vertical axis but is limited by the paper width (PSIZE in NAMELIST OPTION) on the horizontal axis. If rotation of the model causes the projection to go off the paper, it is rescaled prior to plotting. PLOTSZ is used in scaling only if ISCALE = 1).

XORGN-0.0 X location of the plot origin.
(Used only if ISCALE = 2.)

YORGN-0.0 Y location of the plot origin.
(Used only if ISCALE = 2.)

PSCALE-1.0 Model size reduction factor.
PSCALE is equal to the actual model size divided by the desired plot size. It is used only if ISCALE = 2. (Note that when ISCALE = 2 is used, no rescaling occurs if rotation causes the projection to exceed the paper width.)

NOTAT-0

- 0- No numbering on plots.
- 1- Numbering of grid points.
- 2- Numbering of elements.

XLHT-0.14

Height in inches of the integers specified by NOTAT. It should be a multiple of .07. If XLHT is not a multiple of .07, it will be rounded to the nearest multiple. XLHT has a maximum of .49 and a minimum of .07.

KDISP-0

- 0- Plot of undeformed structure.
- 1- Plot of deformed structure.
- 2- Exploded plot.
- 3- Displacement represented by vectors.

KDISP = 1 or 3 represents a form of postprocessing and displacement data must be input in (9). If postprocessing is desired, refer to reference 6.

IDMAG-2

- 1- Direct magnification of displacement data by DMAGS.
- 2- Scaling of displacement data to a maximum value of DMAGS.

DMAGS-1.0

Magnification of displacements (if KDISP = 1 or 3). Reduction factor of elements (if KDISP = 2).

KSYMXY-0

- 1- Symmetry about X-Y plane.

KSYMZX-0 1- Symmetry about X-Z plane.

KSYMZY-0 1- Symmetry about Y-Z plane.

 A plate quadrant with KSYMZX and
KSYMZY equal to 1 would yield a complete plate.
See example 2 in Section III in this thesis.

Note: To develop a partial plot, three methods of
segregating elements exist: First, by X, Y and Z
cutting planes; second, by node numbers; and third,
by element numbers. The next ten variables are used
to separate elements into partial plots.

XXMAX,YYMAX,ZZMAX-1.0E20 Local cutting planes
XXMIN,YYMIN,ZZMIN-(-1.0E20) Parallel to the principal
 planes.

NDMAX-9999999 Maximum gridpoint identification
 number to be included in the plot.

NDMIN-0 Minimum gridpoint identification
 number to be included in the plot.

NELMAX-9999999 Maximum element number to be
 included in the plot.

NELMIN-0 Minimum element number to be
 included in the plot.

KODE-0

Specifies the control option after the plot is completed.

0- Last plot, exit from program.

1- Read another NAMELIST PICT.

2- Read a new set of displacement data. (Postprocessing only.) For KODE = 2, displacement data must be followed by another NAMELIST PICT.

3- Read a complete new set of input data starting with a title card.

For KODE = 1, 2 or 3, additional sections of the deck must be repeated. The deck must end with a NAMELIST PICT having a value of KODE = 0.

Note: A most important point to remember when generating a sequence of plots is that once a parameter has been defined, it retains that value until it is reassigned. For example, if PLOTSZ is assigned a value of 8.0 for the first of a series of plots, and it is not redefined in any subsequent NAMELIST PICT; the value of PLOTSZ will be retained as originally specified. However, when KODE = 3 and a new title, NAMELIST OPTION and problem data are read, all variables in NAMELIST OPTION and NAMELIST PICT are assigned their default values. The

problem starts over in this case. Refer to the flow chart in figure 1 and study the path for different values of KODE.

(12) Delimiter card.

/*

Delimiter card is defined in chapter 3 of reference 8.

III. SPECIAL FEATURES OF PSAP1

A. POSTPROCESSING

Reference 6 contains information and examples on the use of the postprocessor. SAP IV has the capability to punch displacement data cards in an acceptable format for PSAP1 (specifically subroutine DATA9). As of this writing, ADINA does not have this capability, and data cards would have to be punched manually by the user.

B. PARTIAL DATA

Reference 6, on page 30, establishes a procedure by which it is possible to input only a portion of the finite element model for a data check. This feature is valuable in a case where several different people are preparing different parts of a large data base for a problem and desire to check individually their inputs graphically for accuracy. All nodal coordinates for the entire model may be input, or only those that specifically define the portion of the finite element model to be plotted. In either case, all nodal coordinates that define the elements to be plotted must be specified. To use this feature the "element control cards" (described in references 1 and 2) must be modified. For example, if only the connectivity for elements 15 through 50 of a problem are available, the changes below would be made.

1. All SAP IV Elements

All SAP IV element control card changes would have similar changes because columns 6-10 contain the total number of group elements. Columns 66-70 are not used. Thus to plot only elements 15 through 50, make the following changes:

a. Enter the upper bound (i.e., 50) in columns 6-10.

b. Enter the lower bound (i.e., 15) in columns 66-70.

2. ADINA Truss, 2D and 3D Elements

ADINA element control cards for the truss, 2D continuum and the 3D continuum elements would be changed as follows:

a. Enter the upper bound (i.e., 50) in columns 5-8.

b. Enter the lower bound (i.e., 15) in columns 53-56.

3. ADINA Beam Element

ADINA element control cards for the beam element would be changed as follows:

a. Enter the upper bound (i.e., 50) in columns 5-8.

b. Enter the lower bound (i.e., 15) in columns 65-68.

APPENDIX B

SUBROUTINE PSAP1 LISTING

```

00000010
00000020
00000030
00000040
00000050
00000060
00000070
00000080
00000090
00000100
00000110
00000120
00000130
00000140
00000150
00000160
00000170
00000180
00000190
00000200
00000210
00000220
00000230
00000240
00000250
00000260
00000270
00000280
00000290
00000300
00000310
00000320
00000330
00000340
00000350
00000360
00000370
00000380

MAIN PROGRAM
DIMENSION ZZZ(1400),DISPD(5,3,200)
CALL PSAP1(ZZZ,1400,DISPD,200)
STOP
END
PSAP1
SUBROUTINE PSAP1 DOCUMENTATION
DESCRIPTION OF INPUT DATA CARDS
TITLE CARD -- 80 ALPHANUMERIC CHARACTERS OF GRAPH TITLE INFORMATION
TO BE PRINTED ABOVE AND BELOW THE GRAPH. THE FIRST 40
CHARACTERS WILL FORM THE FIRST TITLE LINE. THE LAST 40
THE SECOND LINE.
NAMELIST OPTION - CONTAINS VALUES TO VERIFY STORAGE IN BLANK
COMMON AND CONTROL VALUES NEEDED BY THE PROGRAM.
THE FOLLOWING VALUES ARE INCLUDED---
NNDST = ESTIMATE NUMBER OF GRID POINTS TO BE USED. VALUE MUST
BE GREATER THAN OR EQUAL TO THE ACTUAL NUMBER OF GRID
POINTS.
** DEFAULT = 200 **
NUDISP = 0 FOR NO DISPLACEMENT DATA IN X-DIRECTION.
** DEFAULT = 1 FOR DATA INCLUDING DISPLACEMENTS IN X-DIRECTION.
NVDISP = 0 FOR NO DISPLACEMENT DATA IN Y-DIRECTION.
** DEFAULT = 1 FOR DATA INCLUDING DISPLACEMENTS IN Y-DIRECTION.
NWDISP = 0 FOR NO DISPLACEMENT DATA IN Z-DIRECTION.
** DEFAULT = 1 FOR DATA INCLUDING DISPLACEMENTS IN Z-DIRECTION.
KGEOM SPECIFIES SUBROUTINE AND CORRESPONDING METHOD OF INPUT
FOR MODEL GEOMETRY.
KGEOM = 1 FOR USER SUPPLIED SUBROUTINE - GEOM1

```

```

000000390
000000400
000000410
000000420
000000430
000000440
000000450
000000460
000000470
000000480
000000490
000000500
000000510
000000520
000000530
000000540
000000550
000000560
000000570
000000580
000000590
000000600
000000610
000000620
000000630
000000640
000000650
000000660
000000670
000000680
000000690
000000700
000000710
000000720
000000730
000000740
000000750
000000760
000000770
000000780
000000790
000000800
000000810
000000820
000000830
000000840
000000850
000000860

GEOM1 DEVELOPED TO READ ADINA GEOMETRY DATA -- MAR 77
= 2 FOR USER SUPPLIED SUBROUTINE - GEOM2
= 9 FOR SAP IV DATA DECK INPUT SUBROUTINE - GEOM9.
** GEOM9 READS SAP IV GEOMETRY DATA - MODIFIED MAR 77
** DEFAULT = 9 **
KDATA SPECIFIES SUBROUTINE AND CORRESPONDING METHOD OF INPUT
FOR DISPLACEMENT DATA.
KDATA = 1 FOR SUBROUTINE DATA TO READ IN DISPLACEMENT DATA
-- SUPPLIED BY THE USER.
= 5 FOR SUBROUTINE DATA5 TO READ IN DISPLACEMENT DATA
BY THE USER.
= 9 FOR SUBROUTINE DATA9 TO READ SAP IV DATA.
** DEFAULT = 9 **
NVALUS - NOT USED AT NPS ----- ALLOW DEFAULT

** DEFAULT = 0 **
IRESEQ - NOT USED AT NPS ----- ALLOW TO DEFAULT
** DEFAULT = 1 ** TYPE OF OUTPUT DEVICE TO BE USED.
KPLLOT SPECIFIES THE TYPE OF OUTPUT DEVICE TO BE USED.
KPLLOT = 1 FOR CALCOMP.
= 2 FOR LANGLEY RESEARCH CENTER USE ONLY
= 3 FOR LRC USE ONLY
= 4 FOR LRC USE ONLY
** DEFAULT = 1 **
YSPACE = SPACE BETWEEN PLOTS IN Y DIRECTION (INCHES) WHEN
MULTIPLE PLOTS ARE PRODUCED. YSPACE/2.0 IS SPACE
BETWEEN TITLE BLOCK AND PLOT.
** DEFAULT = 2.0 **
PSIZE = PAPER SIZE IN X DIRECTION, USED IN SCALING OF
PLOTS TO INSURE THIS DIMENSION IS NOT EXCEEDED.
** DEFAULT = 5.0 **
IDCASE = 0 FOR NO TITLE CARD PRECEDING
DECKS OF DISPLACEMENT VALUES.
= 1 FOR TITLE CARD PRECEDING
DECKS OF DISPLACEMENT VALUES.
** DEFAULT = 0 **

MODEL GEOMETRY IS NOW INPUT IN ONE OF THE FOLLOWING FORMS,
DEPENDING ON THE VALUE OF KGEOM SPECIFIED IN NAMELIST OPTION.

USE IF KGEOM = 1
CALL SUBROUTINE GEOM1 WHICH READS ADINA GEOMETRY DATA
USE IF KGEOM = 2
CALL SUBROUTINE GEOM2 WHICH IS PREPARED BY THE USER TO

```

0000087C
 00000880
 CC0C089C
 00000900
 00000910
 C0000920
 00000930
 00000940
 CC000950
 00000960
 00000970
 C000098C
 00000990
 00001000
 C00C1010
 00001020
 00001030
 00001040
 00001050
 CC00106C
 00001070
 00001080
 C0001090
 00001100
 00001110
 00001120
 00001130
 00001140
 00001150
 00001160
 C00C117C
 00001180
 C0001190
 00001200
 00001210
 C0001220
 00001230
 00001240
 C0001250
 00001260
 00001270
 00001280
 00001290
 C0001300
 00001310
 00001320
 CC0C133C
 00001340

READ GECOMETRY DATA.

USE IF KGEOM = 9

CALL SUBROUTINE GECM9 WHICH READS SAP IV GEOMETRY DATA.

CASE IDENTIFICATION CARD.

THIS CARD IS OMITTED IF IDCASE=0 IS SPECIFIED IN EOPTION
 IF PRESENT, THIS CARD CONTAINS ANY DESIRED ALPHANUMERIC
 INFORMATION IN COLS. 1-80 WILL NOT APPEAR ON PLOT BUT WILL
 APPEAR ON PRINTOUT ABOVE DISPLACEMENT DATA

DATA TO BE PLOTTED IS NOW INPUT IN ONE OF THE FOLLOWING FORMS
 DEPENDING ON THE VALUE OF KDATA SPECIFIED IN NAMELIST OPTION.

USE IF KDATA = 1

CALL SUBROUTINE DATA1 WHICH IS PREPARED BY THE USER

USE IF KDATA = 5

CALL SUBROUTINE DATA5 WHICH IS PREPARED BY THE USER

USE IF KDATA = 9

CALL SUBROUTINE DATA9 WHICH READS SAP IV DISPLACEMENT DATA.
 A DISPLACEMENT DATA DECK CAN BE PREPARED FOR ADINA IN A
 FORMAT COMPATIBLE WITH DATA9.

NAMELIST PICT - CONTAINS VALUES NEEDED TO GENERATE PLOTS.

THE FOLLOWING VALUES ARE INCLUDED----

KHORZ = INTEGER DESIGNATING HORIZONTAL AXIS OF VIEWING PLANE,
 WHERE 1=X, 2=Y, 3=Z.
 ** DEFAULT = 1 **

KVERT = INTEGER DESIGNATING VERTICAL AXIS OF VIEWING PLANE,

CC

```

00001350
00001360
00001370
00001380
00001390
00001400
00001410
00001420
00001430
00001440
00001450
00001460
00001470
00001480
00001490
00001500
00001510
00001520
00001530
00001540
00001550
00001560
00001570
00001580
00001590
00001600
00001610
00001620
00001630
00001640
00001650
00001660
00001670
00001680
00001690
00001700
00001710
00001720
00001730
00001740
00001750
00001760
00001770
00001780
00001790
00001800
00001810
00001820

WHERE 1=X, 2=Y, 3=Z.
** DEFAULT = 2 **
** ANGULAR ROTATION OF MODEL ABOUT ITS X-AXIS, IN DEGREES
(PHI) (MUST BE TAKEN THIRD).
** DEFAULT = 0.0 **
** ANGULAR ROTATION OF MODEL ABOUT ITS Y-AXIS, IN DEGREES
(THETA) (MUST BE TAKEN SECOND).
** DEFAULT = 0.0 **
** ANGULAR ROTATION OF MODEL ABOUT ITS Z-AXIS, IN DEGREES
(PHI) (MUST BE TAKEN FIRST).
** DEFAULT = 0.0 **
** CHANGE BEFORE PLOT IS MADE.
** FRAME CHANGE RESETS THE Y-ORIGIN PAST PREVIOUS PLOT
** (A FRAME CHANGE AND X ORIGIN AT 0.0)
** NEWFR = 1 FOR NO FRAME CHANGE BEFORE PLOTTING
** DEFAULT = 1 **
** INTERNAL ORIGIN LOCATION AND SCALING.
** ISCALE = 1 FOR USER SPECIFIED ORIGIN AND SCALING.
** ISCALE = 2 FOR NO SCALE CHANGE. (I.E. USE SAME SCALE AS PREVIOUS
(PLOT) THIS IS USEFUL IN AN ASSEMBLY GRAPH WHERE IT IS
NECESSARY TO EXAMINE A MESH IN SECTIONS WITHOUT LOSING
PERSPECTIVE.
** DEFAULT = 1 **
** DIMENSION DESIRED ON COMPLETED PLOT.
** PLOTSZ = MAXIMUM FOR SCALING IF ISCALE = 1)
** PLOTSZ = 1 FOR SCALING THE PLOT PRIOR TO ROTATION. IF ROTATION
CAUSES THE PLOT TO EXCEED PAPER WIDTH (P.SIZE), IT IS
RESCALED AND THE PLOT SIZE IS REDUCED ACCORDINGLY.
** DEFAULT = 10.0 **
** X-LOCATION OF PLOT ORIGIN (USED IF ISCALE = 2).
** XORGN = 0 **
** Y-LOCATION OF PLOT ORIGIN (USED IF ISCALE = 2).
** YORGN = 0 **
** MODEL SIZE REDUCTION FACTOR, PSCALE = ACTUAL MODEL
SIZE/DESIRED PLOT SIZE (USED IF ISCALE = 2).
** PSCALE = 1.0 **
** DEFAULT = 1.0 **
** NO NUMBERING ON PLOTS.
** NOTAT = 0 **
** NUMBERING OF GRID POINTS.
** NOTAT = 1 **
** NUMBERING OF ELEMENTS.
** NOTAT = 2 **
** HEIGHT OF INTEGERS SPECIFIED BY NCTAT, IN INCHES.
** XLHT = 0 **
** DEFAULT = C.15 **
** UNDEFORMED PLOT.
** KDISP = 0 **
** UNDEFORMED PLOT.
** KDISP = 1 **
** EXPLODED PLOT.
** KDISP = 2 **
** DISPLACEMENTS REPRESENTED BY VECTORS.
** KDISP = 3 **
** DEFAULT = C **
** DIRECT SCALING OF DATA BY DMAGS.
** IC MAG = 1 **

```



```

00002310
00002320
00002330
00002340
00002350
00002360
00002370
00002380
00002390
00002400
00002410
00002420
00002430
00002440
00002450
00002460
00002470
00002480
00002490
00002500
00002510
00002520
00002530
00002540
00002550
00002560
00002570
00002580
00002590
00002600
00002610
00002620
00002630
00002640
00002650
00002660
00002670
00002680
00002690
00002700
00002710
00002720
00002730
00002740
00002750
00002760
00002770
00002780

*****
THE SUBROUTINES USED IN THE ACTUAL CREATION OF PLOTS BY
THE CALCOMP MODEL 765 CAN BE FOUND IN NPS TECHNICAL NOTE
NUMBER 0211-03, "PLOTTING PACKAGE FOR NPS IBM 360/367".
*****
SUBROUTINE PSAP1 IS A MODIFICATION TO NAVAL POSTGRADUATE
SCHOOL THESIS BY LT. D. M. LOSH, DECEMBER 1976. MODIFICATION
INCLUDED SAP IV 8-21 NODE BRICK ELEMENTS, BOUNDARY ELEMENTS AND
ADINA TRUSS, PLANE, BRICK, BEAM ELEMENTS, AND OTHER MINOR
IMPROVEMENTS.
*****
MODIFIED BY ADRIAN E. KIBLER JR.
LT USN
NAVAL PCSTGRADUATE SCHOOL
MONTEREY, CA.
JAN - JUN 1977
*****
SUBROUTINE PSAP1 (ZZZ, NZ, DISPD, NON)
* * * * *
*** THIS IS THE MAIN SUBROUTINE WHICH CALLS OTHER SUBROUTINES
* * * * *
INTEGER NUMPT, XPT, YPT, ZPT, UPT, VPT, WPT
COMMON/CDATA/NTIME, NTL
COMMON/CONTRL/KGEOM, KDATA, KPLOT, KSYMXY, KSYMxz, KSYMZ, NGTAT, XLHT,
LKHZRZ, KVERT, PHI, THETA, PSI, NEWFR, ISCALE, PLOTSZ, XORGN, YORGN,
2PSCALE, KDISP, DMAG, KODE
COMMON/LIMITS/ XXMAX, YYMAX, ZZMAX, XXMIN, YYMIN, ZZMIN, NDMAX, NDMIN,
1NELMAX, NELMIN
COMMON/CORGN/ YPMAX, YSPACE, PSIZE
COMMON/GLOOP/ ILOOP
*****

```

```

COMMON/ABLK/ A(3,3)
COMMON/SAVEV/ DMAGS, IDMAG
COMMON/KOUNT/ NNODE, NKDEST, NUDISP, NVDISP, NWCISP
COMMON/VALUES/ NVALUS
COMMON/CASEID/ IDCASE
DIMENSION ZZ(NZ), DISPD(5,3,NN), ABCD1(10), ABCD2(10), ABCD3(10),
1ABCD4(10)
NAMELIST/PICT/ KHORZ, KVERT, PHI, THETA, PSI, NEWFR, ISCALE
1PLOTSZ, XORG, YORG, YORGN, PSCALE, NOTAT, KDISP, IDMAG, DMAGS, KODE,
2KSYMXY, KSYMZX, KSYMZY, XXMAX, YYMAX, ZZMAX, XXMIN,
3YYMIN, ZZMIN, NDMAX, NDMIN, NELMAX, NELMIN, XLHT
C *** TO ZERO NODE AND ELEMENT SUMMATION COUNTERS
C
C ILOOP = 0
C NNODE = 0
C YPMAX=0.0
C
C *** TO DEFINE THE ORIGIN AND OPEN PLOTTING DATA SETS
C
C CALL CALCMP
C CALL CALPLT(-10.0,0.0,-3)
C CALL CALPLT(1.0,6.0,-3)
C CONTINUE
C 500 REWIND 10
C 8 WRITE(6,8)
C FORMAT(IH1)
C
C *** TO READ TITLE CARD FOR RUN
C
C READ(5,9004,END=999) (ABCD1(I),I=1,10), (ABCD2(I),I=1,10)
C 9004 FORMAT(20A4)
C 9006 WRITE(6,9006) (ABCD1(I),I=1,10), (ABCD2(I),I=1,10)
C 9006 FORMAT (///,20X,20A4///)
C CALL INITIAL
C
C *** TO PLOT THE TITLE CARD AT THE BEGINING OF THE PLOT
C
C CALL CALPLT(0.3,1.62,3)
C CALL CALPLT(0.0,0.62,2)
C CALL CALPLT(0.0,0.0,2)
C CALL CALPLT(9.0,0.0,2)
C CALL NOTATE(0.8,0.41,0.21,ABCD1,0.0,40)
C CALL NOTATE(0.8,0.10,0.21,ABCD2,0.0,40)
C CALL CALPLT(0.0,1.62+YSPACE/2.0,-3)
C
C *** TO SET POINTERS FOR BLANK COMMON STORAGE ZZ
C *** (WITH INTEGER NAMES CF ARRAYS USED IN CALLED SUBROUTINES)

```

```

00002790
00002800
00002810
00002820
00002830
00002840
00002850
00002860
00002870
00002880
00002890
00002900
00002910
00002920
00002930
00002940
00002950
00002960
00002970
00002980
00002990
00003000
00003010
00003020
00003030
00003040
00003050
00003060
00003070
00003080
00003090
00003100
00003110
00003120
00003130
00003140
00003150
00003160
00003170
00003180
00003190
00003200
00003210
00003220
00003230
00003240
00003250
00003260

```

```

C
NUMPT = 1
XPT = NUMPT+NNDEST
YPT = XPT+NNDEST
ZPT = YPT+NNDEST
UPT = ZPT+NNDEST
IF (NUDISP.EQ.0) VPT = UPT+1
IF (NUDISP.NE.0) VPT = UPT+NNDEST
IF (NVDISP.EQ.0) WPT = VPT+1
IF (NVDISP.NE.0) WPT = VPT+NNDEST
IF (NWDISP.EQ.0) NEND = WPT+1-1
IF (NWDISP.NE.0) NEND = WPT+NNDEST-1
WRITE(6,15) NEND
15 FORMAT(//,20X,' BLANK COMMON STORAGE ZZZ REQUIRES AT LEAST ',I6,
1,' LOCATIONS FOR THIS CASE',//)
1 IF (KGEOM.EQ.1) CALL GEOM1
1 (ZZZ(NUMPT),ZZZ(XPT),ZZZ(ZPT),ZZZ(YPT),ZZZ(UPT),ZZZ(VPT),ZZZ(WPT))
1 IF (KGEOM.EQ.2) CALL GEOM2
1 (ZZZ(NUMPT),ZZZ(XPT),ZZZ(ZPT),ZZZ(YPT),ZZZ(UPT),ZZZ(VPT),ZZZ(WPT))
1 IF (KGEOM.EQ.9) CALL GEOM9
1 (ZZZ(NUMPT),ZZZ(XPT),ZZZ(ZPT),ZZZ(YPT),ZZZ(UPT),ZZZ(VPT),ZZZ(WPT))
CALL PNTOUT(1)
1 (ZZZ(NUMPT),ZZZ(XPT),ZZZ(ZPT),ZZZ(YPT),ZZZ(UPT),ZZZ(VPT),ZZZ(WPT))
6CC CONTINUE
IF (IDCASE.EQ.0) GO TO 650
READ(5,9004,END=999) (ABCD3(I),I=1,10),(ABCD4(I),I=1,10)
WRITE(6,9006) (ABCD3(I),I=1,10),(ABCD4(I),I=1,10)
650 CONTINUE
CALL ZEROC
1 (ZZZ(NUMPT),ZZZ(XPT),ZZZ(ZPT),ZZZ(YPT),ZZZ(UPT),ZZZ(VPT),ZZZ(WPT))
1 IF (KDATA.EQ.1) CALL DATA1
1 (ZZZ(NUMPT),ZZZ(XPT),ZZZ(ZPT),ZZZ(YPT),ZZZ(UPT),ZZZ(VPT),ZZZ(WPT))
1 IF (KDATA.EQ.5) CALL DATA5
1 (ZZZ(NUMPT),ZZZ(XPT),ZZZ(ZPT),ZZZ(YPT),ZZZ(UPT),ZZZ(VPT),ZZZ(WPT))
1 IF (KDATA.EQ.9) CALL DATA9
1 (ZZZ(NUMPT),ZZZ(XPT),ZZZ(ZPT),ZZZ(YPT),ZZZ(UPT),ZZZ(VPT),ZZZ(WPT))
2DISP,NCN)
IF (NUDISP.EQ.0.AND.NVDISP.EQ.0.AND.NWDISP.EQ.0) GO TO 700
CALL PNTOUT(2)
1 (ZZZ(NUMPT),ZZZ(XPT),ZZZ(ZPT),ZZZ(YPT),ZZZ(UPT),ZZZ(VPT),ZZZ(WPT))
CONTINUE
700 IF (KPLOT.EQ.4.AND.ILOOP.NE.0) GO TO 6000
WRITE(6,1000)
1000 FORMAT(//)
READ(5,PICT)
WRITE(6,PICT)
6000 CONTINUE
CALL DSCALE
00003270
00003280
00003290
00003300
00003310
00003320
00003330
00003340
00003350
00003360
00003370
00003380
00003390
00003400
00003410
00003420
00003430
00003440
00003450
00003460
00003470
00003480
00003490
00003500
00003510
00003520
00003530
00003540
00003550
00003560
00003570
00003580
00003590
00003600
00003610
00003620
00003630
00003640
00003650
00003660
00003670
00003680
00003690
00003700
00003710
00003720
00003730
00003740

```



```

C      *      *      *      *      *      *      *      *      *      *      *      *      *      *      *
C      *** TO SET UP VALUES FOR CONTROL PARAMETERS
C      *** CALLED BY PSAPI
C      *      *      *      *      *      *      *      *      *      *      *      *      *      *      *
C      COMMON/CDATA/NTIME,NTLC
C      COMMON/CTRL/KGEOM,KDATA,KPLOT,KSXYXZ,KSXYMZ,KSXYZ,NOTAT,XLHT,
C      1KHORZ,KVERT,PHI,THETA,PSI,NEWFR,ISCALE,PLOTSZ,XORGN,YORGN,
C      2PSCALE,KDISP,DMAG,KCODE
C      COMMON/LIMITS/ XXMAX,YYMAX,ZZMAX,XXMIN,YYMIN,ZZMIN,NDMAX,NDMIN,
C      1NELMAX,NELMIN
C      COMMON/CGRGN/ YPMAX,YSPACE,PSIZE
C      COMMON/SAVEV/ DMAGS,DMAG
C      COMMON/KOUNT/ NNODE,NNDEST,NUDISP,NVDISP,NWDISP
C      COMMON/SEQUCE/ IRESEQ
C      COMMON/VALUES/ NVALU
C      COMMON/CASEID/ IDCASE
C      NAMELIST/OPTION/ NNDEST,NUDISP,NVDISP,NWDISP,
C      1KGEOM,KDATA,NVALU,IRESEQ,KPLOT,YSACE,PSIZE,IDCASE
C      *** DESCRIPTION OF VALUES IN &OPTION GIVEN IN SUBROUTINE DOCMNT
C      *      *      *      *      *      *      *      *      *      *      *      *
C      *** TO SET DEFAULT VALUES FOR &OPTION
C      NNDEST = 200
C      NVDISP = 0
C      NWDISP = 0
C      KGEOM = 9
C      KDATA = 9
C      NTIME = 0
C      NVALU = 0
C      IRESEQ = 1
C      KPLOT = 1
C      YSPACE = 2.0
C      PSIZE = 5.0
C      IDCASE = 0
C      *** TO SET DEFAULT VALUES FOR &PICT
C      KHORZ = 1
C      KVERT = 2
C      00004250
C      00004260
C      00004270
C      00004280
C      00004290
C      00004300
C      00004310
C      00004320
C      00004330
C      00004340
C      00004350
C      00004360
C      00004370
C      00004380
C      00004390
C      0000440C
C      00004410
C      00004420
C      00004430
C      00004440
C      00004450
C      00004460
C      00004470
C      00004480
C      00004490
C      00004500
C      00004510
C      00004520
C      00004530
C      00004540
C      00004550
C      00004560
C      0000457C
C      00004580
C      00004590
C      00004600
C      00004610
C      0000462C
C      00004630
C      00004640
C      0000465C
C      00004660
C      00004670
C      00004680
C      00004690
C      00004700
C      0000471C
C      00004720

```

```

C00C473C
00004740
00004750
00004760
00004770
C000478C
00004790
00004800
C00C481C
00004820
00004830
C0004840
00004850
00004860
00004870
00004880
000C4890
00004900
00004910
C00C492C
00004930
00004940
00004950
00004960
C00C497C
00004980
00004990
C0005000
00005010
00005020
00005030
00005040
C0005050
00005060
00005070
C0005080
00005090
00005100
00005110
00005120
00005130
C00C514C
00005150
00005160
00005170
00005180
C00C519C
00005200

```

```

PHI = 0.0
THETA = 0.0
PSI = 0.0
NEWFR = 1
ISCALE = 10.0
XORGN = 0.0
YORGN = 0.0
PSCALE = 1.0
NOTAT = 0.1400
XLHT = 0
KDISP = 0
DMAG = 2
DMAGS = 1.0
KSYMXY = 0
KSYMxz = 0
KSYMZY = 0
XXMAX = 1.0E20
YYMAX = 1.0E20
ZZMAX = 1.0E20
XXMIN = -1.0E20
YYMIN = -1.0E20
ZZMIN = -1.0E20
NDMAX = 99999999
NDMIN = 0
NELMAX = 5559999
NELMIN = 0
CODE = 0
READ(5,OPTION,END=955)
WRITE(6,9010)
FORMAT(///)
WRITE(6,OPTION)
WRITE(6,9010)
RETURN
CALL ERROR(3)
END
SUBROUTINE BOUND(NUMPT,XPT,YPT,ZPT,UPT,VPT,WPT)
* * * * *
** TO DETERMINE MAXIMUM DIMENSIONAL LIMITS OF BODY FOR USE
** IN SCALING PLOTS
** CALLED BY PSAPI
* * * * *
COMMON/CONTRL/ KGEOM,KDATA,KPLOT,KSYMXY,KSYMxz,KSYMZY,NOTAT,XLHT,

```

901C

999

C C C C C C C C

```

CC005210
00005220
00005230
00005240
00005250
00005260
00005270
00005280
00005290
00005300
00005310
00005320
00005330
00005340
00005350
00005360
00005370
00005380
00005390
00005400
00005410
00005420
00005430
00005440
00005450
00005460
00005470
00005480
00005490
00005500
00005510
00005520
00005530
00005540
00005550
00005560
00005570
00005580
00005590
00005600
00005610
00005620
00005630
00005640
00005650
00005660
00005670
00005680

```

```

1 KHORZ, KVERT, PHI, THETA, PSI, NEWFR, ISCALE, PLOTSZ, XORGN, YORGN,
2 PSCALE, KDISP, DMAG, KODE
COMMON/LIMITS/ XXMAX, YYMAX, ZZMAX, XXMIN, YYMIN, ZZMIN, NDMAX, NDMIN,
1 NELMAX, NELMIN
COMMON/XYZLIM/ XYZMAX(3), XYZMIN(3)
COMMON/KOUNT/ NNODE, NNCEST, NUDISP, NVDISP, NWDISP
DIMENSION NUMPT(1), XPT(1), YPT(1), ZPT(1), VPT(1), WPT(1)
DIMENSION NODE(20)
DO 5 I=1,3
XYZMIN(I) = +1.0E20
XYZMAX(I) = -1.0E20
5 CONTINUE
REWIND 10
100 CONTINUE
END=1000) NEND, NUMEL, (NODE(I), I=1, NEND)
IF (NUMEL.LT.NELMIN.OR.NUMEL.GT.NELMAX) GO TO 100
DO 10 I=1, NEND
ND = NODE(I)
IF (NODE(I).EQ.0) GO TO 10
IF (NUMPT(ND).LT.NDMIN.CR.NUMPT(ND).GT.NDMAX) GO TO 100
CONTINUE
DO 20 I=1, NEND
IF (NODE(I).EQ.0) GC TC 20
ND = NODE(I)
IF (XPT(ND).GT.XXMAX) GO TO 20
IF (XPT(ND).LT.XXMIN) GO TO 20
IF (YPT(ND).GT.YYMAX) GO TO 20
IF (YPT(ND).LT.YYMIN) GO TO 20
IF (ZPT(ND).GT.ZZMAX) GO TO 20
IF (ZPT(ND).LT.ZZMIN) GO TO 20
IF (XPT(ND).GT.XYZMAX(1)) XYZMAX(1) = XPT(ND)
IF (XPT(ND).LT.XYZMIN(1)) XYZMIN(1) = XPT(ND)
IF (YPT(ND).GT.XYZMAX(2)) XYZMAX(2) = YPT(ND)
IF (YPT(ND).LT.XYZMIN(2)) XYZMIN(2) = YPT(ND)
IF (ZPT(ND).GT.XYZMAX(3)) XYZMAX(3) = ZPT(ND)
IF (ZPT(ND).LT.XYZMIN(3)) XYZMIN(3) = ZPT(ND)
20 CONTINUE
GO TO 100
1000 CONTINUE
DO 300 I=1,3
AND.KSYMZY.NE.1) GO TO 300
IF (I.EQ.1.AND.KSYMZY.NE.1) GO TO 300
IF (I.EQ.2.AND.KSYMZY.NE.1) GO TO 300
IF (I.EQ.3.AND.KSYMZY.NE.1) GO TO 300
XYZBIG = ABS(XYZMAX(I))
IF (ABS(XYZMIN(I)).GT.XYZBIG) XYZBIG = ABS(XYZMIN(I))
XYZMAX(I) = XYZBIG
XYZMIN(I) = -XYZBIG
300 CONTINUE

```

```

00005690
00005700
00005710
00005720
00005730
00005740
00005750
00005760
00005770
00005780
00005790
00005800
00005810
00005820
00005830
00005840
00005850
00005860
00005870
00005880
00005890
00005900
00005910
00005920
00005930
00005940
00005950
00005960
00005970
00005980
00005990
00006000
00006010
00006020
00006030
00006040
00006050
00006060
00006070
00006080
00006090
00006100
00006110
00006120
00006130
00006140
00006150
00006160

RETURN
END
SUBROUTINE ZEROD(NUMPT,XPT,YPT,ZPT,UPT,VPT,WPT)
* * * * *
*** INITIALIZES ALL DISPLACEMENTS TO ZERO.
*** CALLED BY PSAPI
* * * * *
COMMON/KOUNT/ NNODE, ANDEST, NUDISP, NVDISP, NWDISP
DIMENSION NUMPT(1), XPT(1), YPT(1), ZPT(1), UPT(1), VPT(1), WPT(1)
IF(NUDISP.EQ.0) GO TO 200
DO 150 I=1, NUDISP
  UPT(I) = 0.0
CONTINUE
150 CONTINUE
IF(NVDISP.EQ.0) GO TO 300
DO 250 I=1, NVDISP
  VPT(I) = 0.0
CONTINUE
250 CONTINUE
IF(NWDISP.EQ.0) GO TO 400
DO 350 I=1, NWDISP
  WPT(I) = 0.0
CONTINUE
350 CONTINUE
400 CONTINUE
RETURN
END
SUBROUTINE PNTOUT(IOUT, NUMPT, XPT, YPT, ZPT, UPT, VPT, WPT)
* * * * *
*** FOR PRINTED OUTPUT OF INFORMATION IN BLANK COMMON - ZZZ
*** CALLED BY PSAPI
* * * * *
COMMON/KOUNT/ NNODE, ANDEST, NUDISP, NVDISP, NWDISP
DIMENSION NUMPT(1), XPT(1), YPT(1), ZPT(1), UPT(1), VPT(1), WPT(1)
GO TO (1000,2000), IOUT
1000 CONTINUE
C ** FOR OUTPUT OF GEOMETRY INFORMATION
C
WRITE(6,16)

```

```

16 FORMAT(//,5X,'GRID FCINT INFORMATION',//)
17 WRITE(6,17)
18 FORMAT(5X,'RESEQUENCED',4X,'USER INPUT',/
15X,'GRID PCINT',5X,'GRID PCINT',/
25X,'NUMBER',9X,'NUMBER',13X,'X',14X,'Y',14X,'Z',//)
DO 30 I=1,NNODE
WRITE(6,18) I,NUMPT(I),XPT(I),YPT(I),ZPT(I)
18 FORMAT(2X,I10,5X,I10,3X,3E15.4)
30 CONTINUE
19 WRITE(6,19)
19 FORMAT(//,5X,'ELEMENT INFORMATION - WITH RESEQUENCED GRID POINTS
1,////)
WRITE(6,9008)
9008 FORMAT(1X,'RESEQUENCED',4X,'USER INPUT',25X,'GRID POINTS',/
11X,'ELEMENT',8X,'ELEMENT',/
21X,'NUMBER',9X,'NUMBER',7X,' 1 2 3 4 5 6 7
3 8 9 10 11 12 13 14 15 16 17 18 19 20',//)
REWIND 10
I=0
35 CONTINUE
I=I+1
READ(10,END=999) NEND,NUMEL,(NODE(J),J=1,NEND)
IF(NEND.EQ.12) GO TO 40
WRITE(6,9010) I,NUMEL,(NODE(J),J=1,NEND)
9010 FORMAT(1X,I4,11X,I4,9X,20I5)
GO TO 35
WRITE(6,9010) I,NUMEL,(NODE(J),J=1,4),(NODE(J),J=9,12)
GO TO 35
2000 CONTINUE
C *** FOR OUTPUT OF DISPLACEMENT DATA
C
C
WRITE(6,210)
210 FORMAT(//,5X,'DISPLACEMENTS TO BE PLOTTED',//)
WRITE(6,17)
DO 230 I=1,NNODE
U=0.0
IF(NVDISP.NE.0) U=LPT(I)
V=0.0
IF(NVDISP.NE.0) V=VPT(I)
W=0.0
IF(NWDISP.NE.0) W=WPT(I)
WRITE(6,18) I,NUMPT(I),U,V,W
230 CONTINUE
955 RETURN
END
SUBROUTINE PLOTX(NUMFT,XPT,YPT,ZPT,UPT,VPT,WPT)
C

```

```

0000617C
00006180
00006190
00006200
00006210
0000622C
00006230
00006240
00006250
00006260
00006270
00006280
00006290
00006300
00006310
00006320
00006330
00006340
00006350
00006360
00006370
00006380
00006390
00006400
00006410
00006420
00006430
0000644C
00006450
00006460
0000647C
00006480
00006490
00006500
00006510
00006520
00006530
00006540
00006550
00006560
00006570
00006580
00006590
00006600
00006610
00006620
00006630
00006640

```

```

C    *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *
C *** FOR GENERATING PLOTS.
C *** CALLED BY PSAPI
C    *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *
C    *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *
COMMON/CONTRL/ KGEOM, KDATA, KPLOT, KSYMXY, KSYMZX, KSYMZY, NOTAT, XLHT,
1KHORZ, KVERT, PHI, THETA, PSI, NEWFR, ISCALE, PLOTSZ, XORGN, YORGN,
2PSCALE, KDISP, DMAG, KODE
COMMON/LIMITS/ XXMAX, YYMAX, ZZMAX, XXMIN, YYMIN, ZZMIN, ND MAX, NDMIN,
1NELMAX, NELMIN
COMMON/XYZLIM/ XYZMAX(3), XYZMIN(3)
COMMON/CORGN/ YPMAX, YSPACE, PSIZE
COMMON/GLOOP/ ILOOP
COMMON/ABLK/ A(3,3)
COMMON/KOUNT/ NNODE, NNDEST, NUDISP, NVDISP, NWCISP
COMMON/PDELS/ DELX, DELY
COMMON/NUMPT(1), XPT(1), YPT(1), ZPT(1), UPT(1), VPT(1), WPT(1)
DIMENSION NODE(20), X(20), Y(20), Z(20), XDISP(20), YDISP(20),
DIMENSION NODE(20), XROT(20), YROT(20), XP(21), YP(21)
1ZDISP(20)
C    *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *
C *** TO MAKE ALL GRID POINT NUMBERS NEGATIVE
C    DO 50 I=1, NNODE
C    IF(NUMPT(I).GT.0) NUMPT(I) = -NUMPT(I)
C    CONTINUE
C    *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *
C *** TO MAKE FRAME CHANGE IF NEWFR = 1
C *** NO FRAME CHANGE CN FIRST PLOT AFTER NAMELIST OPTION
C    YMOVE=0.0
C    IF(ILOOP.EQ.0) GO TO 70
C    IF(NEWFR.EQ.1) YMOVE=YPMAX+YSPACE
C    CALL CALPLT(0.0, YMOVE, -3)
C    GO TO (710, 703, 710), KPLOT
C    CONTINUE
C    IF(NEWFR.EQ.1) CALL NFRAME
C    CONTINUE
C    IF(ISCALE.EQ.0) DELX=0.0
C    IF(ISCALE.NE.0) DELY=0.0
C    IF(ISCALE.EQ.1) CALL XYSCAL
C    CALL CALPLT(XORGN, YORGN, -3)
C    XSHIFT = 0.0
C    YSHIFT = 0.0
C    YPMAX = -1.0E20
C    *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *
00006650
00006660
00006670
00006680
00006690
00006700
00006710
00006720
00006730
00006740
00006750
00006760
00006770
00006780
00006790
00006800
00006810
00006820
00006830
00006840
00006850
00006860
00006870
00006880
00006890
00006900
00006910
00006920
00006930
00006940
00006950
00006960
00006970
00006980
00006990
00007000
00007010
00007020
00007030
00007040
00007050
00007060
00007070
00007080
00007090
00007100
00007110
00007120

```

```

C *** LOOPS TO ACCOUNT FOR SYMMETRY
C
ZSIGN = +1.0
DO 500 II=1,2
  IF(II.EQ.2.AND.KSYMXZ.NE.1) GO TO 500
  IF(II.EQ.2.AND.KSYMXZ.EQ.1) ZSIGN = -1.0
  YSIGN = +1.0
DO 510 JJ=1,2
  IF(JJ.EQ.2.AND.KSYMXY.NE.1) GO TO 510
  IF(JJ.EQ.2.AND.KSYMXY.EQ.1) YSIGN = -1.0
  XSIGN = +1.0
DO 520 KK=1,2
  IF(KK.EQ.2.AND.KSYMZY.NE.1) GO TO 520
  IF(KK.EQ.2.AND.KSYMZY.EQ.1) XSIGN = -1.0
C *** TO DETERMINE PROJECTED COORDINATES OF ELEMENTS
C
REWRITE 10
CONTINUE
READ(10,END=1000) NEND,NUMEL,(NODE(J),J=1,NEND)
IF(NUMEL.LT.NELMIN.OR.NUMEL.GT.NELMAX) GO TO 100
DO 10 I=1,NEND
  ND = NODE(I)
  IF(NODE(I).EQ.0) GO TO 10
C *** TO MAKE GRID POINT NUMBERS CONNECTED BY ELEMENTS POSITIVE
C
NUMPT(ND) = IABS(NUMPT(ND))
IF(NUMPT(ND).LT.NDMIN.OR.NUMPT(ND).GT.NDMAX) GO TO 100
10 CONTINUE
I = KHRZ
J = KVERT
DO 20 N=1,NEND
  IF(NODE(N).EQ.0) GO TO 20
  ND = NODE(N)
  IF(XPT(ND).GT.XXMAX) GO TO 100
  IF(XPT(ND).LT.XXMIN) GO TO 100
  IF(YPT(ND).GT.YYMAX) GO TO 100
  IF(YPT(ND).LT.YYMIN) GO TO 100
  IF(ZPT(ND).GT.ZZMAX) GO TO 100
  IF(ZPT(ND).LT.ZZMIN) GO TO 100
  XDISP(N) = 0.0
  YDISP(N) = 0.0
  ZDISP(N) = 0.0
  ZDISP(N) = C.0
  IF(KDISP.EQ.1.AND.NUCISP.NE.0) XDISP(N) = UPT(ND)
  IF(KDISP.EQ.1.AND.NVDISP.NE.0) YDISP(N) = VPT(ND)
  IF(KDISP.EQ.1.AND.NWDISP.NE.0) ZDISP(N) = WPT(ND)
  X(N) = XSIGN*(XPT(ND)+XDISP(N))*DMAG+XSHIFT)/PSCALE
  Y(N) = YSIGN*(YPT(ND)+YDISP(N))*DMAG+YSHIFT)/PSCALE
00007130
00007140
00007150
00007160
00007170
00007180
00007190
00007200
00007210
00007220
00007230
00007240
00007250
00007260
00007270
00007280
00007290
00007300
00007310
00007320
00007330
00007340
00007350
00007360
00007370
00007380
00007390
00007400
00007410
00007420
00007430
00007440
00007450
00007460
00007470
00007480
00007490
00007500
00007510
00007520
00007530
00007540
00007550
00007560
00007570
00007580
00007590
00007600

```



```

00007610
00007620
00007630
00007640
00007650
00007660
00007670
00007680
00007690
00007700
00007710
00007720
00007730
00007740
00007750
00007760
00007770
00007780
00007790
00007800
00007810
00007820
00007830
00007840
00007850
00007860
00007870
00007880
00007890
00007900
00007910
00007920
00007930
00007940
00007950
00007960
00007970
00007980
00007990
00008000
00008010
00008020
00008030
00008040
00008050
00008060
00008070
00008080

```

```

20 Z(N) = ZSIGN*(ZPT(ND)+ZDISP(N)*DMAG+ZSHIFT)/PSCALE
CONTINUE
IF(KDISP.EQ.2) CALL XFLOD(NEND,X,Y,Z,NODE)
XCENT = 0.0
YCENT = 0.0
FND=0.0
DO 25 N=1,NEND
IF(NODE(N).EQ.0) GO TO 25
XROT(N) = A(I,1)*X(N)+A(I,2)*Y(N)+A(I,3)*Z(N)
YROT(N) = A(J,1)*X(N)+A(J,2)*Y(N)+A(J,3)*Z(N)
IF(N.GT.8) GO TO 24
FND=FND+1.0
XCENT = XCENT+XROT(N)
YCENT = YCENT+YROT(N)
CONTINUE
24 XROT(N) = XROT(N)+DELX
YROT(N) = YROT(N)+DELY
IF(YROT(N).GT.YPMAX) YPMAX=YROT(N)
25 CONTINUE
IF(NOTAT.NE.2) GO TO 29
XCENT = XCENT/FND-(6.0/7.0)*XLHT
YCENT = YCENT/DELX
YCENT = YCENT+DELY
AL = NUMEL
SUBROUTINE NUMBER APPLIES ONLY TO CALCOMP
29 CONTINUE
IF(NOTAT.EQ.2) CALL NUMBER(XCENT,YCENT,XLHT,AL,0.0,-1)
C *** TO PLOT ELEMENTS
C
C IF(NEND.EQ.2) GO TO 280
C IF(NEND.EQ.4) GO TO 300
C IF(NEND.EQ.8) GO TO 320
C IF(NEND.EQ.12) GO TO 340
C CALL ERROR(4)
C
C ***TO PLOT 2 NODE ELEMENT
C
280 CONTINUE
CALL CALPLT(XROT(1),YROT(1),3)
CALL CALPLT(XROT(2),YROT(2),2)
GO TO 430
C
C *** TO PLOT 3 AND 4 NODE PLANE ELEMENT
C
300 CONTINUE

```

00008090
 00008100
 00008110
 00008120
 00008130
 00008140
 00008150
 00008160
 00008170
 00008180
 00008190
 00008200
 00008210
 00008220
 00008230
 00008240
 00008250
 00008260
 00008270
 00008280
 00008290
 00008300
 00008310
 00008320
 00008330
 00008340
 00008350
 00008360
 00008370
 00008380
 00008390
 00008400
 00008410
 00008420
 00008430
 00008440
 00008450
 00008460
 00008470
 00008480
 00008490
 00008500
 00008510
 00008520
 00008530
 00008540
 00008550

```

CALL CALPLT(XROT(1), YROT(1), 3)
DO 305 NP=2, NEND
CALL CALPLT(XROT(NP), YROT(NP), 2)
CONTINUE
305 CALL CALPLT(XROT(1), YROT(1), 2)
GO TO 430

C *** TO PLOT 8 NODE 3-D BRICK
C
C 320 CONTINUE
LP=1
DO 330 NP=2, 6, 4
  NP2=NP+2
  CALL CALPLT(XROT(LP), YROT(LP), 3)
  DO 325 MP=NP, NP2
    CALL CALPLT(XROT(MP), YROT(MP), 2)
    CONTINUE
  325 CALL CALPLT(XROT(LP), YROT(LP), 2)
  LP=LP+4
  CONTINUE
  330 DO 335 NP=1, 4
    NP4=NP+4
    CALL CALPLT(XROT(NP), YROT(NP), 3)
    CALL CALPLT(XROT(NP4), YROT(NP4), 2)
    CONTINUE
  335 GO TO 430

C *** TO PLOT VARIABLE 4-8 NODE PLANE AND 8-20 NOCE BRICK ELEMENTS
C *** NOTE SUBROUTINE LINE ONLY APPLIES TO THE CALCOMP PLOTTER
C
C 340 CONTINUE
LP=1
KP=8
DO 365 NP=2, 6, 4
  NP2=NP+2
  CALL CALPLT(XROT(LP), YROT(LP), 3)
  DO 345 MP=NP, NP2
    KP=KP+1
    N=2
    CALL WHERE(XP(1), YP(1))
    XP(2)=XROT(MP)
    YP(2)=YROT(MP)
    XP(3)=XROT(KP)
    YP(3)=YROT(KP)
    IF(NODE(KP).NE.0) CALL CURVE(XP, YP, N)
    CALL LINE(XP, YP, N, 1, 1)
    CONTINUE
  345 KP=KP+1

```

AD-A046 385

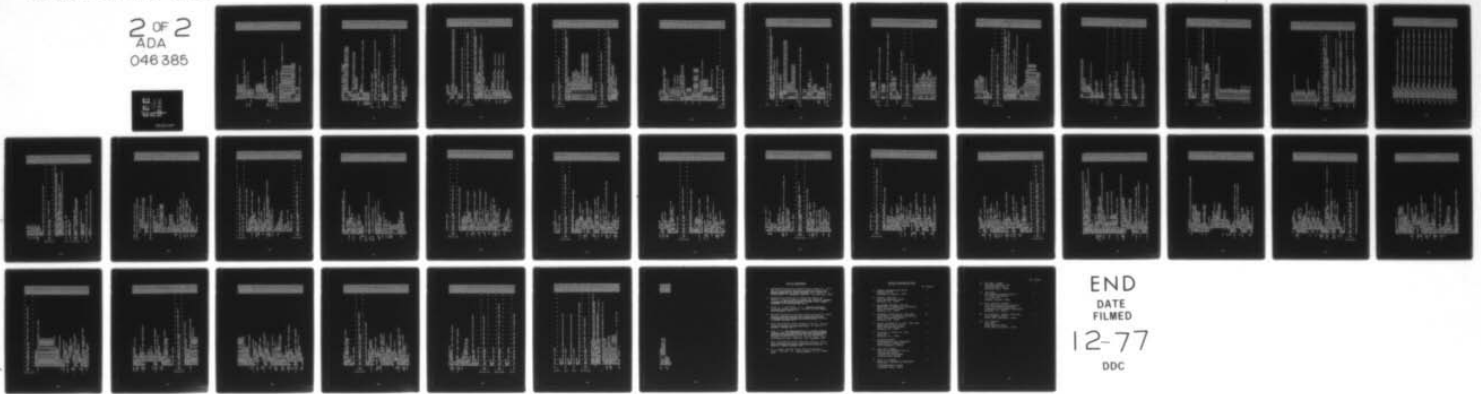
NAVAL POSTGRADUATE SCHOOL MONTEREY CALIF
A FINITE ELEMENT PREPROCESSOR FOR SAP IV AND ADINA.(U)
SEP 77 A E KIBLER

F/G 9/2

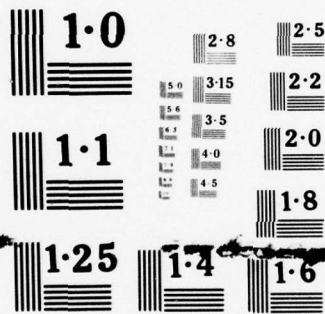
UNCLASSIFIED

NL

2 of 2
ADA
046 385



END
DATE
FILMED
12-77
DDC



NATIONAL BUREAU OF STANDARDS
MICROCOPY RESOLUTION TEST CHART

```

00008570
00008580
00008590
00008600
00008610
00008620
00008630
00008640
00008650
00008660
00008670
00008680
00008690
00008700
00008710
00008720
00008730
00008740
00008750
00008760
00008770
00008780
00008790
00008800
00008810
00008820
00008830
00008840
00008850
00008860
00008870
00008880
00008890
00008900
00008910
00008920
00008930
00008940
00008950
00008960
00008970
00008980
00008990
00009000
00009010
00009020
00009030
00009040

```

```

N=2
CALL WHERE(XP(1), YP(1))
XP(2)=XROT(LP)
YP(2)=YROT(LP)
XP(3)=XROT(KP)
YP(3)=YROT(KP)
IF(NODE(KP).NE.0) CALL CURVE(XP, YP, N)
CALL LINE(XP, YP, N, 1, 1)
LP=LP+4
IF(NEND.EQ.12) GO TO 430
CONTINUE
355 DO 390 NP=1,4
365 NP4=NP+4
KP=NP+16
N=2
XP(1)=XROT(NP)
YP(1)=YROT(NP)
XP(2)=XROT(NP4)
YP(2)=YROT(NP4)
XP(3)=XROT(KP)
YP(3)=YROT(KP)
IF(NODE(KP).NE.0) CALL CURVE(XP, YP, N)
CALL LINE(XP, YP, N, 1, 1)
CONTINUE
390 CONTINUE
430 GO TO 100
1000 CONTINUE
600 IF(KDISP.NE.3) GO TO 650
600 CONTINUE
C *** TO PLOT VECTORS AT GRID POINTS
C
DO 601 ND=1, NNODE
IF(NUMPT(ND).LE.0) GO TO 601
IF(NUMPT(ND).LT.NDMIN.OR.NUMPT(ND).GT.NDMAX) GO TO 601
IF(XPT(ND).GT.XYZMAX(1)) GO TO 601
IF(XPT(ND).LT.XYZMIN(1)) GO TO 601
IF(YPT(ND).GT.XYZMAX(2)) GO TO 601
IF(YPT(ND).LT.XYZMIN(2)) GO TO 601
IF(ZPT(ND).GT.XYZMAX(3)) GO TO 601
IF(ZPT(ND).LT.XYZMIN(3)) GO TO 601
X(1) = XSIGN*(XPT(ND)+XSHIFT)/PSCALE
Y(1) = YSIGN*(YPT(ND)+YSHIFT)/PSCALE
Z(1) = ZSIGN*(ZPT(ND)+ZSHIFT)/PSCALE
XDISP(1) = 0.0
YDISP(1) = 0.0
ZDISP(1) = 0.0
IF(NUDISP.NE.0) XDISP(1) = UPT(ND)

```



```

00009490
00009500
00009510
00009520
00009530
00009540
00009550
00009560
00009570
00009580
00009590
00009600
00009610
00009620
00009630
00009640
00009650
00009660
00009670
00009680
00009690
00009700
00009710
00009720
00009730
00009740
00009750
00009760
00009770
00009780
00009790
00009800
00009810
00009820
00009830
00009840
00009850
00009860
00009870
00009880
00009890
00009900
00009910
00009920
00009930
00009940
00009950
00009960

100      Y(I)=YP(I)
           CONTINUE
           DO 200 I=1,21
           YP(I)=Y(I)*R*(R-1.0)/2.0-Y(3)*(R+1.0)*(R-1.0)+Y(2)*R*(R+1.0)/2.0
           XP(I)=X(I)*R*(R-1.0)/2.0-X(3)*(R+1.0)*(R-1.0)+X(2)*R*(R+1.0)/2.0
           R=R+0.1
           CONTINUE
200      N=21
           RETURN
           END
           SUBROUTINE OSSCALE(NUMPT,XPT,YPT,ZPT,UPT,VPT,WPT)
           * * * * *
           *** THIS SUBROUTINE DETERMINES THE SCALE FACTOR FOR DISPLACEMENTS
           *** CALLED BY PSAPI
           * * * * *
COMMON/CONTRL/ KGEOM, KDATA, KPLOT, KSYMXY, KSYMZX, KSYMZY, NOTAT, XLHT,
1KHORZ, KVERT, PHI, THETA, PSI, NEWFR, ISCALE, PLOTSZ, XORGN, YORGN,
2PSCALE, KDISP, DMAG, KODE
COMMON/SAVEV/ DMAGS, DMAG
COMMON/KOUNT/ NNODE, NNODET, NUDISP, NVDISP, NWCISP
DIMENSION NUMPT(1),XPT(1),YPT(1),ZPT(1),UPT(1),VPT(1),WPT(1)
IF(KDISP.EQ.0.OR.KDISF.EQ.2) GO TO 10
GO TO (10,20), IDMAG
10 CONTINUE
DMAG = DMAGS
GO TO 30
20 CONTINUE
DMAX = 0.0
DO 100 I=1,NNODE
IF(NUDISP.EC.0) GO TC 500
IF(ABS(UPT(I)).GT.DMAX) DMAX = ABS(UPT(I))
CONTINUE
IF(NVDISP.EQ.0) GO TC 501
IF(ABS(VPT(I)).GT.DMAX) DMAX = ABS(VPT(I))
CONTINUE
IF(NWDISP.EQ.0) GO TC 502
IF(ABS(WPT(I)).GT.DMAX) DMAX = ABS(WPT(I))
CONTINUE
DMAG = DMAGS/DMAX
100 CONTINUE
30 CONTINUE
RETURN
END

```

```

00009570
00009980
00009990
00000010
00000020
00000030
00000040
00000050
00000060
00000070
00000080
00000090
00000100
00000110
00000120
00000130
00000140
00000150
00000160
00000170
00000180
00000190
00000200
00000210
00000220
00000230
00000240
00000250
00000260
00000270
00000280
00000290
00000300
00000310
00000320
00000330
00000340
00000350
00000360
00000370
00000380
00000390
00000400
00000410
00000420
00000430
00000440
00000450

SUBROUTINE RCTAT
* * * * *
*** SETS UP COEFFICIENTS OF ROTATION MATRIX
*** CALLED BY PSAPI
* * * * *
COMMON/CONTRL/ KGEOM, KDATA, KPLOT, KSYMXY, KSYMZX, KSYMZY, NOTAT, XLHT,
1KHORZ, KVERT, PHI, THETA, PSI, NEWFR, ISCALE, PLOTSZ, XORG, YORG,
2PSCALE, KDISP, DMAG, KCCE
COMMON/ABLK/ A(3, 3)
PI = 3.1415926536
SINPHI = SIN(PHI*PI/180.0)
COSPHI = COS(PHI*PI/180.0)
SINTHE = SIN(THETA*PI/180.0)
COSTHE = COS(THETA*PI/180.0)
SINPSI = SIN(PSI*PI/180.0)
COSPSI = COS(PSI*PI/180.0)
A(1,1) = COSTHE*COSPSI
A(1,2) = SINTHE*SINTHE*SINPHI-SINPHI*COSPHI
A(1,3) = SINTHE*COSPHI*SINPHI+SINPHI*SINPSI
A(2,1) = SINTHE*COSPHI
A(2,2) = SINPSI*COSPHI
A(2,3) = SINTHE*SINPHI*SINPSI+SINPSI*COSPHI
A(3,1) = -SINTHE
A(3,2) = COSTHE*SINPHI
A(3,3) = COSTHE*COSPHI
RETURN
END
SUBROUTINE XYSCAL
* * * * *
*** TO DETERMINE SCALE FACTOR FOR MODEL GEOMETRY.
*** CALLED BY PLOTX
* * * * *
COMMON/CONTRL/ KGEOM, KDATA, KPLOT, KSYMXY, KSYMZX, KSYMZY, NOTAT, XLHT,
1KHORZ, KVERT, PHI, THETA, PSI, NEWFR, ISCALE, PLOTSZ, XORG, YORG,
2PSCALE, KDISP, DMAG, KCCE
COMMON/XYZLIM/ XYZMAX(3), XYZMIN(3)
COMMON/CORGN/ YPMAX, YPSACE, PSIZE
COMMON/ABLK/ A(3, 3)
COMMON/PDELS/ DELX, DELY
I = KHORZ

```



```

C00C046C
00000470
00000480
00000490
00000500
00000510
00000520
00000530
00000540
00000550
00000560
00000570
00000580
00000590
00000600
00000610
00000620
00000630
00000640
00000650
00000660
00000670
00000680
00000690
00000700
00000710
00000720
00000730
00000740
00000750
00000760
00000770
00000780
00000790
00000800
00000810
00000820
00000830
00000840
00000850
00000860
00000870
00000880
00000890
00000900
00000910
00000920
00000930

```

```

J = KVERT
DMAX = 0.0
DO 5 N=1,3
  VDUM = ABS(XYZMAX(N)-XYZMIN(N))
  IF(VDUM.GT.DMAX) DMAX = VDUM
CONTINUE = DMAX/PLOTSZ
DO 10 L=1,2
DO 10 M=1,2
DO 10 N=1,2
  X = XYZMIN(1)
  IF(L.EC.2) X = XYZMAX(1)
  Y = XYZMIN(2)
  IF(M.EC.2) Y = XYZMAX(2)
  Z = XYZMIN(3)
  IF(N.EQ.2) Z = XYZMAX(3)
  XR0T = A(I,1)*X+A(I,2)*Y+A(I,3)*Z
  YR0T = A(J,1)*X+A(J,2)*Y+A(J,3)*Z
  IF(L*M*N.NE.1) GO TO 30
CONTINUE
XRMIN = XR0T
XRMAX = XR0T
YRMIN = YR0T
YRMAX = YR0T
30 CONTINUE
IF(XR0T.GT.XRMAX) XRMAX = XR0T
IF(XR0T.LT.XRMIN) XRMIN = XR0T
IF(YR0T.GT.YRMAX) YRMAX = YR0T
IF(YR0T.LT.YRMIN) YRMIN = YR0T
10 CONTINUE
XR=ABS(XRMAX-XRMIN)
IF(XR/PSCALE.GT.PSIZE) PSCALE=XR/PSIZE
XRMAX = XRMAX/PSCALE
XRMIN = YRMAX/PSCALE
YRMIN = XRMIN/PSCALE
YRMAX = YRMIN/PSCALE
DELY = -XRMIN
DELY = -YRMIN
XORGN = (PSIZE-XR/PSCALE)/2.0
RETURN
END
SUBROUTINE XPLOD(NEND,X,Y,Z,NODE)
C * * * * *
C *** FOR GENERATING EXPLODED PLOTS.
C *** CALLED BY PLOTX

```

```

C * * * * *
C C  COMMON/CONTRL/ KGEOM,KDATA,KPLOT,KSYMXY,KSYMxz,KSYMz,NOTAT,XLHT,
      IKHORZ,KVERT,PHI,THETA,PSI,NEWFR,ISCALE,PLOTSZ,XORGN,YORGN,
      2PSCALE,KDISP,DMAG,KODE
      DIMENSION X(20),Y(20),Z(20),NODE(20)
C * * * * *
C C  *** TO CALCULATE THE INCENTER OF TRIANGLES
      IF(NODE(4).EQ.0) NEND=3
      IF(NEND.NE.3) GO TO 20
      CONTINUE
10  A = SQRT((X(2)-X(3))**2+(Y(2)-Y(3))**2+(Z(2)-Z(3))**2)
      B = SQRT((X(1)-X(3))**2+(Y(1)-Y(3))**2+(Z(1)-Z(3))**2)
      C = SQRT((X(1)-X(2))**2+(Y(1)-Y(2))**2+(Z(1)-Z(2))**2)
      AC1 = A/(A+B+C)
      AC2 = B/(A+B+C)
      AC3 = C/(A+B+C)
      XOC = AC1*X(1)+AC2*X(2)+AC3*X(3)
      YOC = AC1*Y(1)+AC2*Y(2)+AC3*Y(3)
      ZOC = AC1*Z(1)+AC2*Z(2)+AC3*Z(3)
      GO TO 190
20  CONTINUE
C * * * * *
C C  *** TO CALCULATE THE CENTROID OF RODS, BARS, AND QUADS
      XOC = 0.0
      YOC = 0.0
      ZOC = 0.0
      FND=0.0
      DO 100 I=1,NEND
      IF(NODE(I).EQ.0) GO TO 100
      IF(I.GT.8) GO TO 101
      FND=FND+1.0
      XOC = XOC+X(I)
      YOC = YOC+Y(I)
      ZOC = ZOC+Z(I)
      CONTINUE
100 CONTINUE
101 XOC=XOC/FND
      YOC=YOC/FND
      ZOC=ZOC/FND
      CONTINUE
190 CONTINUE
C * * * * *
C C  *** TO REDUCE THE SIZE OF THE ELEMENT
      DO 200 I=1,NEND

```

```

00000940
00000950
00000960
00000970
00000980
00000990
00001000
00001010
00001020
00001030
00001040
00001050
00001060
00001070
00001080
00001090
00001100
00001110
00001120
00001130
00001140
00001150
00001160
00001170
00001180
00001190
00001200
00001210
00001220
00001230
00001240
00001250
00001260
00001270
00001280
00001290
00001300
00001310
00001320
00001330
00001340
00001350
00001360
00001370
00001380
00001390
00001400
00001410

```



```

XROT = A(IJ,1)*X+A(IJ,2)*Y+A(IJ,3)*Z
YROT = A(IJ,1)*X+A(IJ,2)*Y+A(IJ,3)*Z
XL = XROT+XLHT/2.0
YL = YROT+XLHT/2.0
XL = XL+DELY
YL = YL+DELY
AL = NUMPT(I)
CALL NUMBER(XL,YL,XLHT,AL,0.0,-1)
CONTINUE
500 RETURN
END
SUBROUTINE NFRAME
CALLED BY PLOTX
END
SUBROUTINE CCRT2
RETURN
END
C * * * * *
C * * * * * ADAPT FGR NPS SYSTEM
C * * * * * CALLED BY PSAPI
C * * * * *
C * * * * * SUBROUTINE CALCMP
C * * * * * COMMON/PLOTX/IBUFF(1024)
C * * * * * CALL PLOTS
C * * * * * RETURN
C * * * * * END
C * * * * *
C * * * * * ADAPT FOR NPS SYSTEM
C * * * * * CALLED BY PSAPI/PLOTX/GARROW/ERROR
C * * * * *
C * * * * * SUBROUTINE CALPLT(X,Y,IPEN)
C * * * * * CALL PLOT(X,Y,IPEN)
C * * * * * RETURN
C * * * * * END
C * * * * *
C * * * * * ADAPT FOR NPS SYSTEM
C * * * * * CALLED BY PSAPI
C * * * * *

```

```

00002380
00002390
00002400
00002410
00002420
00002430
00002440
00002450
00002460
00002470
00002480
00002490
00002500
00002510
00002520
00002530
00002540
00002550
00002560
00002570
00002580
00002590
00002600
00002610
00002620
00002630
00002640
00002650
00002660
00002670
00002680
00002690
00002700
00002710
00002720
00002730
00002740
00002750
00002760
00002770
00002780
00002790
00002800
00002810
00002820
00002830
00002840
00002850

```

```

C * * * * * SUBROUTINE NOTATE(X,Y,HT,BCD,THETA,N)
C * * * * * DIMENSION BCD(1)
C * * * * * CALL SYMBOL(X,Y,HT,BCD,THETA,N)
C * * * * * RETURN
C * * * * * END
C * * * * * SUBROUTINE ELTYPE(MTYPE,KGEOM)
C * * * * *
C * * * * * *** THIS SUBROUTINE CALLS OTHER ROUTINES TO READ ELEMENT CONNECTIVITY
C * * * * * *** MTYPE = ELEMENT TYPE
C * * * * * *** KGEOM = 1 ACINA ELEMENTS
C * * * * * *** KGEOM = 2 NONSAP ELEMENTS
C * * * * * *** KGEOM = 9 SAP IV ELEMENTS
C * * * * * *** CALLED BY GEOM1/GEOM2/GEOM9/
C * * * * *
C * * * * * IF (KGEOM.EQ.1) GO TO 20
C * * * * * IF (KGEOM.EQ.2) GO TO 40
C * * * * * GO TO (1,2,3,4,5,6,7,8,9,10,11,12), MTYPE
C * * * * *
C * * * * * 1 CALL TRUSS
C * * * * * 2 CALL BEAM
C * * * * * 3 CALL PLANE
C * * * * * 4 CALL PLANE
C * * * * * 5 CALL THREED
C * * * * * 6 CALL SHELL
C * * * * * 7 CALL BNDRY
C * * * * * 8 CALL SOL21
C * * * * * 9 CALL ERROR(1)
C * * * * * 10 CALL ERROR(1)
C * * * * * 11 CALL ERROR(1)
C * * * * * 12 CALL ERROR(1)
C * * * * *
00002860
00002870
00002880
00002890
00002900
00002910
00002920
00002930
00002940
00002950
00002960
00002970
00002980
00002990
00003000
00003010
00003020
00003030
00003040
00003050
00003060
00003070
00003080
00003090
00003100
00003110
00003120
00003130
00003140
00003150
00003160
00003170
00003180
00003190
00003200
00003210
00003220
00003230
00003240
00003250
00003260
00003270
00003280
00003290
00003300
00003310
00003320
00003330

```

```

00003340
00003350
00003360
00003370
00003380
00003390
00003400
00003410
00003420
00003430
00003440
00003450
00003460
00003470
00003480
00003490
00003500
00003510
00003520
00003530
00003540
00003550
00003560
00003570
00003580
00003590
00003600
00003610
00003620
00003630
00003640
00003650
00003660
00003670
00003680
00003690
00003700
00003710
00003720
00003730
00003740
00003750
00003760
00003770
00003780
00003790
00003800
00003810

20 GO TO 900
21 GO TO (21,22,23,24),MTYPE
22 CALL ADTRUS
23 CALL ADPLAN
24 CALL AD3DEE
40 CONTINUE (41,42,43),MTYPE
41 CALL NSTRUS
42 CALL NSPLAN
43 CALL NS3DEE
900 RETURN
END
SUBROUTINE FRROR(N)
* * * * *
*** THIS SUBROUTINE TERMINATES THE PROGRAM DUE TO ERROR IN INPUT.
*** ERROR ALSO ZEROS AND ADVANCES THE CALCOMP PLOTTER
*** CALLED BY ELTYPE/PSAPI/INITIAL/PLOTX/THREED/SOL21/ADTRUS/ADPLAN/
*** AD3DEE/ADBEAM/NSTRUS/NSPLAN/NS3DEE/GEOM2/
* * * * *
COMMON/CORGN/ YPMAX, YSPACE, PSIZE
CALL CALPLT(0.0, YPMAX+6.0, -3)
GO TO(1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20),N
1 CONTINUE
WRITE(6,9001)
9001 FORMAT(///,1X,'TERMINATION OCCURRED IN SUBROUTINE ELTYPE ELEMENT
1 IN INPUT DATA CANNOT BE PLOTTED CHECK ELEMENT TYPES'///)
2 CONTINUE
WRITE(6,9002)
9002 FORMAT(///,1X,'ABNORMAL TERMINATION OCCURRED IN SUBROUTINE PSAPI C
1 CHECK VALUE OF KODE IN NAMELIST PICT'///)
3 CONTINUE
WRITE(6,9003)
9003 FORMAT(///,1X,'ABNORMAL TERMINATION OCCURRED IN SUBROUTINE INITIAL
1- ATTEMPT TO READ NAMELIST OPTION'///)

```

```

4      GO TO 1000
      CONTINUE
9004  WRITE(6,9004)
      FORMAT(/,IX,'ABNORMAL TERMINATION OCCURRED IN PLOTX'///)
      GO TO 1000
5      CONTINUE
9005  WRITE(6,9005)
      FORMAT(/,,'ABNORMAL TERMINATION IN THREEED,ELEMENT CARD ERROR'///)
      GO TO 1000
6      CONTINUE
9006  WRITE(6,9006)
      FORMAT(/,,'ABNORMAL TERMINATION IN SOL21 ,ELEMENT CARD ERROR'///)
      GO TO 1000
7      CONTINUE
9007  WRITE(6,9007)
      FORMAT(/,,'ABNORMAL TERMINATION IN ADRUS,ELEMENT CARD ERROR'///)
      GO TO 1000
8      CONTINUE
9008  WRITE(6,9008)
      FORMAT(/,,'ABNORMAL TERMINATION IN ADPLAN,ELEMENT CARD ERROR'///)
      GO TO 1000
9      CONTINUE
9009  WRITE(6,9009)
      FORMAT(/,,'ABNORMAL TERMINATION IN AD3DEE,ELEMENT CARD ERROR'///)
      GO TO 1000
10     CONTINUE
9010  WRITE(6,9010)
      FORMAT(/,,'ABNORMAL TERMINATION IN ADBEAM,ELEMENT CARD ERROR'///)
      GO TO 1000
11     CONTINUE
9011  WRITE(6,9011)
      FORMAT(/,,'ABNORMAL TERMINATION IN NSTRUS,ELEMENT CARD ERROR'///)
      GO TO 1000
12     CONTINUE
9012  WRITE(6,9012)
      FORMAT(/,,'ABNORMAL TERMINATION IN NSPLAN,ELEMENT CARD ERROR'///)
      GO TO 1000
13     CONTINUE
9013  WRITE(6,9013)
      FORMAT(/,,'ABNORMAL TERMINATION IN NS3DEE,ELEMENT CARD ERROR'///)
      GO TO 1000
14     CONTINUE
9014  WRITE(6,9014)
      FORMAT(/,,'ABNORMAL TERMINATION NONSAP MESH CANNOT BE PLOTTED'///)
      GO TO 1000
15     CONTINUE
      GO TO 1000
16     CONTINUE

```

```

00003820
00003830
00003840
00003850
00003860
00003870
00003880
00003890
00003900
00003910
00003920
00003930
00003940
00003950
00003960
00003970
00003980
00003990
00004000
00004010
00004020
00004030
00004040
00004050
00004060
00004070
00004080
00004090
00004100
00004110
00004120
00004130
00004140
00004150
00004160
00004170
00004180
00004190
00004200
00004210
00004220
00004230
00004240
00004250
00004260
00004270
00004280
00004290

```



```

17 GO TO 1000
   CONTINUE
18 GO TO 1000
   CONTINUE
19 GO TO 1000
   CONTINUE
20 GO TO 1000
   CONTINUE
1000 CALL PSTOP
      RETURN
      END
SUBROUTINE GEOM9(NUMFT,XPT,YPT,ZPT,UPT,VPT,WPT)
  * * * * *
  *** GEOM9 READS SAP IV GEOMETRY DATA
  *** CALLED BY PSAPI
  * * * * *
  COMMON/CONTRL/ KGEOM,KDATA,KPLOT,KSYMXY,KSYMZX,KSYMZY,NOTAT,XLHT,
1KHORZ,KVERT,PHI,THETA,PSI,NEWFR,ISCALE,PLOTSZ,XORGN,YORGN,
2PSCALE,KDISP,DMAG,KOODE
  COMMON/KOUNT/ NNODE,NNDIST,NUDISP,NVDISP,NWDISP
  COMMON/GCONT/NUMNP,NPAR(20),NELTYP,NUMEL
  DIMENSION NUMPT(1),XPT(1),YPT(1),ZPT(1),UPT(1),VPT(1),WPT(1)
  DATA CTEST/'C' /
  * * * * *
  *** INSERT ROUTINE HERE
  * * * * *
  READ(5,120) HED
  FORMAT(12A6)
  * * * * *
  *** READ MASTER CONTROL CARD
  *** NUMNP = TOTAL NUMBER OF NODE POINTS
  *** NELTYP = NUMBER OF ELEMENT GROUPS
  * * * * *
  READ(5,200) NUMNP,NELTYP
  FORMAT(2I5)
  NNODE=NUMNP
  * * * * *
  *** READ OR GENERATE NODAL POINT DATA
  * * * * *
  NOLD=0
  10 READ(5,9006) CT,N,XPT(N),YPT(N),ZPT(N),

```

```

00004300
00004310
00004320
00004330
00004340
00004350
00004360
00004370
00004380
00004390
00004400
00004410
00004420
00004430
00004440
00004450
00004460
00004470
00004480
00004490
00004500
00004510
00004520
00004530
00004540
00004550
00004560
00004570
00004580
00004590
00004600
00004610
00004620
00004630
00004640
00004650
00004660
00004670
00004680
00004690
00004700
00004710
00004720
00004730
00004740
00004750
00004760
00004770

```

00004780
 C0004790
 C0004800
 C0004810
 C0004820
 C0004830
 C0004840
 C0004850
 C0004860
 C0004870
 C0004880
 C0004890
 C0004900
 C0004910
 C0004920
 C0004930
 C0004940
 C0004950
 C0004960
 C0004970
 C0004980
 C0004990
 C0005000
 C0005010
 C0005020
 C0005030
 C0005040
 C0005050
 C0005060
 C0005070
 C0005080
 C0005090
 C0005100
 C0005110
 C0005120
 C0005130
 C0005140
 C0005150
 C0005160
 C0005170
 C0005180
 C0005190
 C0005200
 C0005210
 C0005220
 C0005230
 C0005240
 C0005250

```

9006 FORMAT(A1,I4,30X,3F10.0,I5)
C
C **CHECK FOR CYLINDRICAL COORDINATES
C
  IF(CT.NE.CTEST) GO TO 20
  R=XPT(N)
  XPT(N)=R*SIN(ZPT(N)/57.2958)
  ZPT(N)=R*COS(ZPT(N)/57.2958)
20 CONTINUE
  NUMPT(N)=N
  IF (NOLD.EQ.0) GO TO 50
C
C*****CHECK IF GENERATION IS REQUIRED
C
  IF (KN.EQ.0) GO TO 50
  NUM=(N-NOLD)/KN
  NUMN=NUM-1
  IF (NUMN.LT.1) GO TO 50
  XNUM=NUM
  DX=(XPT(N)-XPT(NOLD))/XNUM
  DY=(YPT(N)-YPT(NOLD))/XNUM
  DZ=(ZPT(N)-ZPT(NOLD))/XNUM
  K=NOLD
  DO 30 J=1,NUMN
  KK=K
  K=K+KN
  XPT(K)=XPT(KK)+DX
  YPT(K)=YPT(KK)+DY
  ZPT(K)=ZPT(KK)+DZ
  NUMPT(K)=K
  CONTINUE
30 NOLD=N
50 IF (N.NE.NUMNP) GO TO 10
  NUMEL=0
C***** READ ELEMENT CONTROL CARDS
  DO 900 M=1,NELTYP
  READ(5,1001)END=999) (NPAR(I),I=1,14)
  FORMAT(14I5)
1001 WRITE(6,9010) (NPAR(I),I=1,14)
9010 FORMAT(//,10NPAR = ,20I5//)
  MTYPE=NPAR(1)
  CALL ELTYPE(MTYPE,KGEOM)
900 CONTINUE
  ENDFILE 10
999 RETURN
  END
SUBROUTINE TRUSS
C

```

```

C * * * * * READS SAP IV TRUSS ELEMENT CARDS (ELTYPE 1)
C *** CALLED BY ELTYPE * * * * *
C * * * * * COMMON/GCONT/NUMNP,NFAR(20),NELTYP,NUMEL
C * * * * * N2=2
C * * * * * NUME=NPARG(2)
C * * * * * NUMMAT=NPARG(3)
C * * * * * READ MATERIAL PROPERTY CARDS (DUMMY)
C * * * * * DO 10 I=1,NUMMAT
C * * * * * READ(5,10C1) DUMMY
C * * * * * FORMAT(10A8)
C * * * * * CONTINUE
C * * * * * READ ELEMENT LOAD MUL. (DUMMY1)
C * * * * * DO 20 I=1,4
C * * * * * READ(5,1001) DUMMY1
C * * * * * CONTINUE
C * * * * * IF(NPARG(14).EQ.0) NPARG(14) = 1
C * * * * * N = NPARG(14)
C * * * * * READ ELEMENT CONNECTION INFORMATION OR GENERATE
C * * * * * 100 READ(5,1004) M,II,JJ,MTYP,TEM,KK
C * * * * * 1004 FORMAT(4I5,F10.0,15)
C * * * * * IF(KK.EQ.0) KK=1
C * * * * * IF (M.NE.N) GO TO 200
C * * * * * I=II
C * * * * * J=JJ
C * * * * * KKK=KK
C * * * * * CONTINUE
C * * * * * NUMEL=NUMEL+1
C * * * * * WRITE(10) N2,N,I,J
C * * * * * IF(N.EQ.NUMEL) RETURN
C * * * * * N=N+1
C * * * * * I=I+KKK
C * * * * * J=J+KKK
C * * * * * IF(N.GT.M) GO TO 100
C * * * * * GO TO 120
C * * * * * END
C * * * * * SUBROUTINE PLANE
C * * * * *
C * * * * * READS SAP IV MEMBRANE ELEMENT CARDS (ELTYPE 3)
C * * * * * CALLED BY ELTYPE
C * * * * *

```

00005740
 00005750
 00005760
 00005770
 00005780
 00005790
 00005800
 00005810
 00005820
 00005830
 00005840
 00005850
 00005860
 00005870
 00005880
 00005890
 00005900
 00005910
 00005920
 00005930
 00005940
 00005950
 00005960
 00005970
 00005980
 00005990
 00006000
 00006010
 00006020
 00006030
 00006040
 00006050
 00006060
 00006070
 00006080
 00006090
 00006100
 00006110
 00006120
 00006130
 00006140
 00006150
 00006160
 00006170
 00006180
 00006190
 00006200
 00006210

```

C
  DIMENSION EMUL(4,5), IE(5), IX(4),
COMMON/GCONT/NUMNP, NPAR(20), NELTYP, NUMEL
N4=4
  NUME = NPAR(2)
  NUMMAT = NPAR(3)
  C **** READ MATERIAL PROPERTIES
  DO 60 M=1, NUMMAT
  1010 READ(5,1010) MAT,NT
  FORMAT(2I5)
  IF(NT.EQ.0) NT=1
  NTC=2*NT
  DO 50 K=1,NTC
  1005 READ(5,1005) DUMMY
  50 CONTINUE
  60 CONTINUE
  C**** READ ELEMENT LOAD FACTORS
  C
  1002 READ(5,1002) ((EMUL(I,J),J=1,5),I=1,4)
  FORMAT(5F10.0)
  C**** READ ELEMENT PROPERTIES
  C
  IF(NPAR(14).EQ.0) NPAR(14) = 1
  N=NPAR(14)-1
  130 READ(5,1003) M, (IF(I),I=1,4),KG
  1003 FORMAT(5I5,30X,I5)
  IF(KG.EQ.0) KG=1
  140 N=N+1
  IF(M.EQ.N) GO TO 145
  DO 142 I=1,4
  142 IX(I)=IX(I)+KG
  GO TO 150
  145 DO 148 I=1,4
  148 IX(I)=IE(I)
  150 CONTINUE
  I = IX(1)
  J = IX(2)
  K = IX(3)
  L = IX(4)
  NUMEL=NUMEL+1
  WRITE(10) N4,N,I,J,K,L
  310 IF(N.EQ.NUME) RETURN
  IF(N.EC.M) GO TO 130
  GO TO 140
  END

```

```

SUBROUTINE BEAM
* * * * *
*** READS SAP IV BEAM ELEMENT CARDS (ELTYPE 2)
*** CALLED BY ELTYPE
* * * * *
COMMON/GCONT/NUMNP,NPAR(20),NELTYP,NUMEL
N2=2
NUME=NPARG(2)
NUMEPC=NPARG(3)
NUMFEF=NPARG(4) * 2
NUMMAT=NPARG(5)
READ MATERIAL PROPERTY CARDS (DUMMY)
DO 10 I=1,NUMMAT
  READ(5,1001) DUMMY
  FORMAT(10A8)
CONTINUE
DO 20 J=1,NUMEPC
  READ(5,1001) DUMMY1
CONTINUE
READ ELEMENT LOAD MULTIPLIERS(DUMMY2)
DO 30 K=1,3
  READ(5,1001) DUMMY2
CONTINUE
READ FIXED-END FORCE CARDS(DUMMY3)
DO 40 L=1,NUMFEF
  READ(5,1001) DUMMY3
CONTINUE
IF(NPAR(14).EQ.0) NPAR(14) = 1
N=NPARG(14)
READ ELEMENT CONNECTION INFO
FORMAT(3I5,47X,I8)
IF (KK.EQ.0) KK=1
IF (M.NE.N) GO TO 200
  I = II
  J = JJ
  KKK = KK
CONTINUE
NUMEL = NUMEL+1
WRITE(10) N2,N,I,J
IF (N.EQ.NUMEL) RETURN
N = N + 1
  I = I + KKK

```

```

00006220
00006230
00006240
00006250
00006260
00006270
00006280
00006290
00006300
00006310
00006320
00006330
00006340
00006350
00006360
00006370
00006380
00006390
00006400
00006410
00006420
00006430
00006440
00006450
00006460
00006470
00006480
00006490
00006500
00006510
00006520
00006530
00006540
00006550
00006560
00006570
00006580
00006590
00006600
00006610
00006620
00006630
00006640
00006650
00006660
00006670
00006680
00006690

```



```

156 CONTINUE
C *** GO TO 162
GENERATION REQUIRED
160 DO 161 I=1,8
NP(I)=NP(I)+IINC
CONTINUE
161 CONTINUE
NUMEL=NUMEL+1
WRITE(10) N8,NEL,(NP(I),I=1,8)
IF(NEL.EQ.NUMEL) RETURN
IF(NEL.EQ.INEL) GC TC 130
GO TO 140
END
SUBROUTINE SHELL
C *** * * * * * * * * * * * * * * * * * * * * * * *
C *** READS SAP IV SHELL ELEMENT CARDS (ELTYPE 6)
*** CALLED BY ELTYPE
C *** * * * * * * * * * * * * * * * * * * * * * * *
DIMENSION IY(7),IX(4)
COMMON/GCONT/NUMNP,NPAR(20),NELTYP,NUMEL
N4=4
ISTOP=0
NUME = NPAR(2)
NUMMAT = NPAR(3)
NMAT = 2*NUMMAT
C *** READ MATERIAL PROPERTIES (DUMMY)
DO 10 N=1,NMAT
READ(5,1000) DUMMY
C 1000 FORMAT(10A8)
10 CONTINUE
C *** READ ELEMENT LOAD FACTORS (DUMMY1)
DO 20 K=1,5
READ(5,1000) DUMMY1
CONTINUE
20 IF(NPAR(14).EQ.0) NPAR(14) = 1
NN = NPAR(14)-1
100 READ(5,1001) MM,IY
1001 FORMAT(8I5)
110 NN = NN + 1
IF (MM -NN) 440,50,6C
50 DO 45 I=1,7
45 IX(I) = IY(I)
INCL = IY(7)
00007180
00007190
00007200
00007210
00007220
00007230
00007240
00007250
00007260
00007270
00007280
00007290
00007300
00007310
00007320
00007330
00007340
00007350
00007360
00007370
00007380
00007390
00007400
00007410
00007420
00007430
00007440
00007450
00007460
00007470
00007480
00007490
00007500
00007510
00007520
00007530
00007540
00007550
00007560
00007570
00007580
00007590
00007600
00007610
00007620
00007630
00007640
00007650

```

```

IF (INCL.EQ.0) INCL=1
GO TO 70
DO 65 I=1,4
IX(I)=IX(I) + INCL
70 CONTINUE
I=IX(1)
J=IX(2)
K=IX(3)
L=IX(4)
NUMEL = NUMEL +1
WRITE(10) N4,NN,I,J,K,L
GO TO 500
440 WRITE(6,2005) MM
2005 FORMAT(19HOCARD FOR ELEMT(,I5,14H) IS IN ERROR.,IX)
500 IF(MN.LT.MM) GO TO 110
IF(NN.EQ.NUME) RETURN
IF(ISTOP.EQ.1) STOP
GO TO 100
END
SUBROUTINE BNDRY
C * * * * *
C *** READS SAP IV BOUNDARY ELEMENT CARDS (ELTYPE 7)
C *** BOUNDARY ELEMENTS ARE NOT PLOTTED
C *** CALLED BY ELTYPE
C * * * * *
COMMON/GCONT/NUMNP,NPAR(20),NELTYP,NUMEL
NUME=NPAR(2)
READ LOAD CASE CARD (DUMMY)
1002 READ(5,1002) DUMMY
FORMAT(10A8)
C *** READ BOUNDARY ELEMENT CARDS
N=0
100 READ(5,1004) M,I1,KK
N=N+1
IF(N.GE.NUME) RETURN
IF(KK.GT.0) GO TO 200
GO TO 100
200 N=N+(M2-M)/KK
IF(N.GE.NUME) RETURN
GO TO 100
1004 FORMAT(2I5,25X,I5)
END
00007660
00007670
00007680
00007690
00007700
00007710
00007720
00007730
00007740
00007750
00007760
00007770
00007780
00007790
00007800
00007810
00007820
00007830
00007840
00007850
00007860
00007870
00007880
00007890
00007900
00007910
00007920
00007930
00007940
00007950
00007960
00007970
00007980
00007990
00008000
00008010
00008020
00008030
00008040
00008050
00008060
00008070
00008080
00008090
00008100
00008110
00008120
00008130

```



```

1 KHORZ, KVERT, PHI, THETA, PSI, NEWFR, ISCALE, PLOTSZ, XORGN, YORGN,
2 PSCALE, KDISP, DMAG, KCCCE
COMMON /GCONT/ NNODE, NNCST, NUDISP, NVDISP, NWCISP
DIMENSION NUMPT(1), XPT(1), YPT(1), ZPT(1), UPT(1), VPT(1), WPT(1)
DIMENSION IDOF(6), ID(6), IDOLD(6)
1 DATA CTEST, 'X' //
NCARD=0
READ(5, 9000) DUMMY
FORMAT(20A4)
C *** READ MASTER CONTROL CARDS
C *** NUMNP = TOTAL NUMBER OF NODE POINTS
C *** NELTYP = NUMBER OF ELEMENT GROUPS
9001 READ(5, 9001) NUMNP, (IDOF(I), I=1, 6), NEGL, NEGNL, MODEX, NSTE
FORMAT(15, 6I1, 14, 3I5)
NELTYP=NEGL+NEGNL
NNODE=NUMNP
READ(5, 9002) IMASS, IDAMP, IMASSN, IDAMPN
FORMAT(4I5)
9002 READ(5, 9002) IEIG
READ(5, 9002) ISREF, NUMREF, IEQUIT, ITEMAX
READ(5, 9000) DUMMY
READ(5, 9000) DUMMY
READ(5, 9000) DUMMY
C *** READ OR GENERATE NODAL POINT DATA
NOLD=0
NEQ=0
10 READ(5, 9006) CT, N, (ID(I), I=1, 6), XPT(N), YPT(N), ZPT(N), KN
9006 FORMAT(A1, I4, I4, I4, I5, I3, F10.0, I5)
C *** CHECK FOR CYLINDRICAL COORDINATES
IF(CT.NE.CTEST) GO TO 12
DUM=ZPT(N)/57.2958
R=YPT(N)
YPT(N)=R*COS(ZPT(N)/57.2958)
ZPT(N)=R*SIN(ZPT(N)/57.2958)
12 CONTINUE
NUMPT(N)=N
IF(NOLD.EQ.0) GO TO 50
FOR GENERATION OF FIXED BOUNDARY CONDITIONS
C *** DO 15 I=1, 6
15 IF(I.COLD(I).EQ.-1.ANC.ID(I).EQ.0) ID(I)=IDOLD(I)
CONTINUE
IF(KNOLD.EQ.0) GO TO 5C
NUM=(N-NOLD)/KNOLD
NUMN=NUM-1
IF(NUMN.LT.1) GO TO 50
C *** TO COUNT DOFS TO DETERMINE NUMBER OF IC CARDS

```

```

0000911C
0000912C
0000913C
0000914C
0000915C
0000916C
0000917C
0000918C
0000919C
0000920C
0000921C
0000922C
0000923C
0000924C
0000925C
0000926C
0000927C
0000928C
0000929C
0000930C
0000931C
0000932C
0000933C
0000934C
0000935C
0000936C
0000937C
0000938C
0000939C
0000940C
0000941C
0000942C
0000943C
0000944C
0000945C
0000946C
0000947C
0000948C
0000949C
0000950C
0000951C
0000952C
0000953C
0000954C
0000955C
0000956C
0000957C
0000958C

```

00009590
 00009600
 00009610
 00009620
 00009630
 00009640
 00009650
 00009660
 00009670
 00009680
 00009690
 00009700
 00009710
 00009720
 00009730
 00009740
 00009750
 00009760
 00009770
 00009780
 00009790
 00009800
 00009810
 00009820
 00009830
 00009840
 00009850
 00009860
 00009870
 00009880
 00009890
 00009900
 00009910
 00009920
 00009930
 00009940
 00009950
 00009960
 00009970
 00009980
 00009990
 00000010
 00000020
 00000030
 00000040
 00000050
 00000060
 00000070

```

DO 20 I=1,6
  IF(IDOF(I).EQ.0.AND.IDOLD(I).EQ.0) NEQ=NEQ+NUMN
  CONTINUE
  DX=(XPT(N)-XPT(NOLD))/NUM
  IF(CT.NE.CTEST) GO TC 21
  ROLD=YPT(NOLD)/COS(DUMOLD)
  RNEW=YPT(N)/COS(DUM)
  DR=(RNEW-ROLD)/NUM
  DT=(DUM-DUMOLD)/NUM
  GO TO 22
21 CONTINUE
  DY=(YPT(N)-YPT(NOLD))/NUM
  DZ=(ZPT(N)-ZPT(NOLD))/NUM
22 CONTINUE
  K=NOLD
  DO 30 J=1,NUMN
    KK=K
    K=K+KNOLD
    XPT(K)=XPT(KK)+DX
    IF(CT.NE.CTEST) GO TO 26
    ROLD=ROLD+DR
    DUMOLD=DUMOLD+DT
    YPT(K)=ROLD*COS(DUMOLD)
    ZPT(K)=ROLD*SIN(DUMOLD)
    GO TO 28
  CONTINUE
26 YPT(K)=YPT(KK)+DY
  ZPT(K)=ZPT(KK)+DZ
28 CONTINUE
  NUMPT(K)=K
30 CONTINUE
  NOLD=N
  KNOLD=KN
  DUMOLD=DUM
  TO COUNT DOFS TO DETERMINE NUMBER OF IC CARDS
  DO 55 I=1,6
    IF(IDOF(I).EQ.0.AND.ID(I).EQ.0) NEQ=NEQ+1
    IDOLD(I)=ID(I)
  CONTINUE
55 CONTINUE
  IF(N.NE.NUMNPI) GO TO 10
  READ LOAD CONTROL CARDS
  READ(5,9000) DUMMY
  DO 80 I=1,IMASSN
    IF(IMASSN.EQ.0) GO TO 81
    READ(5,9000) DUMMY
  CONTINUE
80 CONTINUE
81 IF(ICAMPN.EQ.0) GO TC 91
  
```

```

C ***
90 DO 90 I=1, IDAMPN
91 READ(5,9000) DUMMY
92 CONTINUE
C ***
93 READ(5,5002) ICON
94 IF(ICON.EQ.0) GO TO 100
95 CARDNR=NEQ/6.0
96 NCARD=INT(CARDNR)
97 TEST=CARDNR-NCARD
98 IF(TEST.GT.0.1) NCARD=NCARD+1
99 DO 95 I=1, NCARD
100 READ(5,9000) DUMMY
101 CONTINUE
102 IF(IMASS.EQ.0) GO TO 100
103 DO 96 I=1, NCARD
104 READ(5,9000) DUMMY
105 CONTINUE
106 DO 98 I=1, NCARD
107 READ(5,5000) DUMMY
108 CONTINUE
109 FORMAT(6E12.6)
110 CONTINUE
111 NUMEL=0
112 WRITE(6,9005) NEQ, NCARD
113 FORMAT(//), NEQ AND NCARD FOR IC IN GEOM1 = ',I5,10X,I5//')
114 READ ELEMENT CONTROL CARDS
115 DO 900 M=1, NELTYP
116 READ(5,9008,END=999)(NPAR(I), I=1,20)
117 WRITE(6,9010) (NPAR(I), I=1,20)
118 FORMAT(//),
119 MTYPE=NPAR(1)
120 CALL ELTYPE(MTYPE, KGEOM1)
121 CONTINUE
122 ENDFILE 10
123 RETURN
124 END
C ***
9005 SUBROUTINE ADRTRUS
9006 * * * * *
9007 *** THIS SUBROUTINE TO READ ADINA TRUSS DATA
9008 *** THIS ROUTINE CALLED BY ELTYPE
9009 * * * * *
9010 COMMON/GCONT/NUMNP, NPAR(20), NELTYP, NUMEL
9011 * * * * *
9012 * * * * *
9013 * * * * *
9014 * * * * *
9015 * * * * *
9016 * * * * *
9017 * * * * *
9018 * * * * *
9019 * * * * *
9020 * * * * *
9021 * * * * *
9022 * * * * *
9023 * * * * *
9024 * * * * *
9025 * * * * *
9026 * * * * *
9027 * * * * *
9028 * * * * *
9029 * * * * *
9030 * * * * *
9031 * * * * *
9032 * * * * *
9033 * * * * *
9034 * * * * *
9035 * * * * *
9036 * * * * *
9037 * * * * *
9038 * * * * *
9039 * * * * *
9040 * * * * *
9041 * * * * *
9042 * * * * *
9043 * * * * *
9044 * * * * *
9045 * * * * *
9046 * * * * *
9047 * * * * *
9048 * * * * *
9049 * * * * *
9050 * * * * *
9051 * * * * *
9052 * * * * *
9053 * * * * *
9054 * * * * *
9055 * * * * *

```

```

00000560
00000570
00000580
00000590
00000600
00000610
00000620
00000630
00000640
00000650
00000660
00000670
00000680
00000690
00000700
00000710
00000720
00000730
00000740
00000750
00000760
00000770
00000780
00000790
00000800
00000810
00000820
00000830
00000840
00000850
00000860
00000870
00000880
00000890
00000900
00000910
00000920
00000930
00000940
00000950
00000960
00000970
00000980
00000990
00001000
00001010
00001020
00001030

```

```

NUMMAT=NP(16)
N2=2
UMMAT=EQ.0) NUMMAT=1
IF(NPAR(15).EQ.1) NCARD=2
IF(NPAR(15).EQ.3) NCARD=3
IF(NPAR(15).NE.2) GO TO 20
CARDNR=NP(17)/8.0
NCARD=INT(CARDNR)
TEST=CARDNR-NCARD
IF(TEST.GT.0.1) NCARD=NCARD+1
NCARD=NCARD+2
CONTINUE
READ MATERIAL PROPERTIES
DO 50 J=1,NUMMAT
DO 45 I=1,NCARD
READ(5,9000) DUMMY
FORMAT(20A4)
CONTINUE
CONTINUE
READ OR GENERATE ELEMENT DATA CARDS
IF(NPAR(14).EQ.0) NPAR(14)=1
NEL=NP(14)-1
READ(5,9002) INEL,II,JJ,IINC
FORMAT(3I5,20X,I5)
IF(IINC.EQ.0) IINC=1
ML=INEL+1
ML=INEL-NEL
IF(ML) 150,155,160
CALL ERROR(7)
NO GENERATION OF NODE POINTS REQUIRED
I=II
J=JJ
GO TO 162
GENERATION OF NODE POINTS REQUIRED
I=I+KN
J=J+KN
CONTINUE
NUMEL=NUMEL+1
WRITE(10) N2,NEL,I,J RETURN
IF(NEL.EQ.NPAR(2)) RETURN
IF(NEL.LT.INEL) GO TO 140
KN=I INC
GO TO 130
END
SUBROUTINE ADPLAN
COMMON/GCONT/NUMNP,NPAR(20),NELTYP,NUMEL
DIMENSION NP(12),INP(8)

```



```

9002 FORMAT (15,15X,15)
9004 READ (5,5004)(INP(I),I=1,8)
140 FORMAT (8I5)
    ML=NEL+1
    IF (ML) 150,155,160
150 CALL ERROR(8)
C *** NO GENERATION OF NODE POINTS REQUIRED
155 DO 156 I=1,4
    I9=I+8
    NP(I)=INP(I)
    NP(I5)=0
    NP(I9)=INP(I5)
    CONTINUE
156 GO TO 162
C *** GENERATION OF NODE POINTS REQUIRED
160 DO 161 I=1,N12
    IF (NP(I).EQ.0) GO TO 161
    NP(I)=NP(I)+KN
    CONTINUE
161 CONTINUE
162 NUMEL=NUMEL+1
    WRITE(10) N12,NEL,(NP(I),I=1,N12)
    IF (NEL.EQ.NPAR(2)) RETURN
    IF (NEL.LT.INEL) GO TC 140
    KN=I*INC
    GO TO 130
END
SUBROUTINE AD3DEE
* * * * *
C *** THIS SUBROUTINE TO READ ADINA 3-D SOLID ELEMENT DATA
C *** THIS ROUTINE CALLED BY ELTYPE
* * * * *
COMMON/GCONT/NUMNP,NPAR(20),NELTYP,NUMEL
DIMENSION NP(20),INP(20)
NUMMAT=NPAR(16)
NSTRES=NPAR(13)
CALCULATE THE NUMBER OF MATERIAL CASE CARDS
IF (NPAR(15).EQ. 1) NCARD=1
IF (NPAR(15).EQ. 2) NCARD=2+NPAR(18)
IF (NPAR(15).EQ. 3) NCARD=4
IF (NPAR(15).EQ. 4) NCARD=4
IF (NPAR(15).EQ. 5) NCARD=2
* * * * *
00001520
00001530
00001540
00001550
00001560
00001570
00001580
00001590
00001600
00001620
00001630
00001640
00001650
00001660
00001670
00001680
00001690
00001700
00001710
00001720
00001730
00001740
00001750
00001760
00001770
00001780
00001790
00001800
00001810
00001820
00001830
00001840
00001850
00001860
00001870
00001880
00001890
00001900
00001910
00001920
00001930
00001940
00001950
00001960
00001970
00001980
00001990

```


00002000
 00002010
 00002020
 00002030
 00002040
 00002050
 00002060
 00002070
 00002080
 00002090
 00002100
 00002110
 00002120
 00002130
 00002140
 00002150
 00002160
 00002170
 00002180
 00002190
 00002200
 00002210
 00002220
 00002230
 00002240
 00002250
 00002260
 00002270
 00002280
 00002290
 00002300
 00002310
 00002320
 00002330
 00002340
 00002350
 00002360
 00002370
 00002380
 00002390
 00002400
 00002410
 00002420
 00002430
 00002440
 00002450
 00002460
 00002470

```

IF(NPAR(15).EQ. 8) NCARD=1
IF(NPAR(15).EQ. 9) NCARD=1
IF(NPAR(15).EQ.10) NCARD=6
IF(NPAR(15).EQ.11) NCARD=6
IF(NPAR(15).NE.12) GC TO 20
CARDNR=NPAR(17)/8.0
NCARD=INT(CARDNR)
TEST=CARDNR-NCARD
IF(TEST.GT.0.1) NCARD=NCARD+1
CONTINUE
20 N20=20
C *** READ MATERIAL PROPERTIES
DO 50 J=1,NUMMAT
READ(5,9000) DUMMY
9000 FORMAT(20A4)
DO 45 I=1,NCARD
READ(5,9000) DUMMY
CONTINUE
45
50
C *** READ STRESS OUTPUT TABLE CARDS
IF(NPAR(13).EQ.0) GC TO 61
DO 60 I=1,NSTRES
READ(5,9000) DUMMY
CONTINUE
60
61
CONTINUE
IF(NPAR(14).EQ.0) NPAR(14)=1
NEL=NPAR(14)-1
READ(5,9002) IMEL, IINC
FORMAT(15,30X,15)
IF(IINC.EQ.0) IINC=1
READ(5,9004) (INP(I),I=1,8)
READ(5,9004) (INP(I),I=9,N20)
FORMAT(12I5)
9004
140
NEL=NEL+1
ML=INEL-NEL
IF(ML) 150,155,160
150
C *** CALL ERROR(9)
155
DO 156 I=1,N20
NP(I)=INP(I)
CONTINUE
156
GO TO 162
C *** GENERATION OF NODE PCINTS REQUIRED
160
DO 161 I=1,N20
IF(NP(I).EQ.0) GC TO 161
NP(I)=NP(I)+KN
CONTINUE
161
CONTINUE
162
  
```

```

NUMEL=NUMEL+1
WRITE(10) N20, NEL, (NP(I), I=1, N20)
IF(NEL.EQ.NPAR(2)) RETURN
IF(NEL.LT.INEL) GO TO 140
KN=IINC
GO TO 130
END
SUBROUTINE ADBEAM
C * * * * *
C *** THIS SUBROUTINE TO READ ADINA 2NODE BEAM ELEMENTS
C *** THIS ROUTINE CALLED BY ELTYPE
C * * * * *
COMMON/GCONT/NUMNP, NPAR(20), NELTYP, NUMEL
N2=2
NUMMAT=NPAR(16)
IF(NUMMAT.EQ.0) NUMMAT=1
READ MATERIAL PROPERTIES
DO 50 J=1, NUMMAT
  DO 45 I=1, 2
    READ(5, 9000) DUMMY
    FORMAT(20A4)
    CONTINUE
  CONTINUE
  READ STRESS OUTPUT TABLE CARDS
  IF(NPAR(13).EQ.0) GO TO 81
  IF(NPAR(14).EQ.0) NPAR(14)=16
  NST=NPAR(13)
  CARDST=NPAR(14)/16.0
  NCDST=INT(CARDST)
  TEST=CARCST-NCDST
  IF(TEST.GT.0.01) NCDST=NCDST+1
  NST=NST*NCDST
  DO 80 I=1, NST
    READ(5, 5000) DUMMY
    CONTINUE
  CONTINUE
  CONTINUE OR GENERATE ELEMENT DATA CARDS
  IF(NPAR(17).EQ.0) NPAR(17)=1
  NEL=NPAR(17)-1
  READ(5, 9002) INEL, II, JJ, IINC
  FORMAT(3I5, 15X, I5)
  IF(IINC.EQ.0) IINC=1
  NEL=INEL+1
  ML=INEL-NEL

```

```

00002480
C0002490
C0002500
00002510
00002520
C0002530
C0002540
00002550
00002560
C0002570
C0002580
C0002590
00002600
C0002610
C0002620
00002630
00002640
00002650
00002660
00002670
00002680
00002690
00002700
00002710
00002720
C0002730
00002740
00002750
00002760
00002770
00002780
00002790
00002800
00002810
00002820
00002830
C0002840
00002850
00002860
00002870
00002880
00002890
00002900
00002910
C0002920
00002930
00002940
C0002950

```

```

150 IF(ML) 150,155,160
C *** NO GENERATION OF NODE POINTS REQUIRED
155 I=I
J=JJ
GO TO 162
C *** GENERATION OF NODE PCINTS REQUIRED
160 I=I+KN
J=J+KN
162 CONTINUE
NUMEL=NUMEL+1
WRITE(10) N2,NEL,I,J
IF(NEL.EQ.NPAR(2)) RETURN
IF(NEL.LT.INEL) GO TO 140
KN=IINC
GO TO 130
END
SUBROUTINE GEOM2(NUMFT,XPT,YPT,ZPT,UPT,VPT,WPT)
CALL ERROR(14)
RETURN
END
SUBROUTINE NSTRUS
* * * * *
*** THIS SUBROUTINE TO READ NON SAP TRUSS ELEMENTS
* * * * *
RETURN
END
SUBROUTINE NSPLAN
* * * * *
*** THIS SUBROUTINE TO READ NON SAP 2 D 8 NODE PLANE ELEMENTS
* * * * *
RETURN
END
SUBROUTINE NS3DEE
* * * * *
*** THIS SUBROUTINE TO READ NON SAP 3-D ELEMENT DATA
* * * * *

```

```

C *** CALLED BY ELTYPE
C * * * * *
C RETURN
C END
C SUBROUTINE DATA1(NUMPT,XPT,YPT,ZPT,UPT,VPT,WPT)
C *** CALLED BY PSAPI
C RETURN
C END
C SUBROUTINE DATA5(NUMPT,XPT,YPT,ZPT,UPT,VPT,WPT)
C *** CALLED BY PSAPI
C RETURN
C END
C SUBROUTINE DATA9(NUMPT,XPT,YPT,ZPT,UPT,VPT,WPT,DISPD,NON)
C * * * * *
C *** USER SUPPLIED DISPLACEMENT INPUT SUBROUTINE.
C *** CALLED BY PSAPI
C * * * * *
C COMMON/CDATA/NTIME,NTLC
C COMMON/CONTRL/KGEOM,KDATA,KPLOT,KSYMXY,KSYMXYZ,KSYMZ,NOTAT,XLHT,
C 1KHORZ,KVERT,PHI,THETA,PSI,NEWFR,ISCALE,PLOTSZ,XORGN,YORGN,
C 2PSCALE,KDISP,DMAG,KODE
C COMMON/KOUNT/NNODE,NNDEST,NVDISP,NWDISP
C DIMENSION NUMPT(1),XPT(1),YPT(1),ZPT(1),UPT(1),VPT(1),WPT(1)
C DIMENSION DISPD(5,3,NCN)
C * * * * *
C IF (NVDISP.EQ.0.AND.NWDISP.EQ.0.AND.NWDISP.EQ.0) GO TO 25
C IF (NTIME.NE.0) GO TO 100
C READ(5,1000) NTLC,SCALEF
C FORMAT(15,F10.0)
C IF (SCALEF.EQ.0) SCALEF=1.0
C 10 READ(5,2000) N,NLCAS,U,V,W
C 2000 FORMAT(2I4,3E12,5)
C DISPD(NLCAS,1,N) = U*SCALEF
C DISPD(NLCAS,2,N) = V*SCALEF
C DISPD(NLCAS,3,N) = W*SCALEF
C IF ( (NLCAS.EQ.NTLC).AND.(N.EQ.1) ) GO TO 100
C GO TO 10
C 100 NTIME = NTIME + 1

```

```

00003440
00003450
00003460
00003470
00003480
00003490
00003500
00003510
00003520
00003530
00003540
00003550
00003560
00003570
00003580
00003590
00003600
00003610
00003620
00003630
00003640
00003650
00003660
00003670
00003680
00003690
00003700
00003710
00003720
00003730
00003740
00003750
00003760
00003770
00003780
00003790
00003800
00003810
00003820
00003830
00003840
00003850
00003860
00003870
00003880
00003890
00003900
00003910

```

```
20C DO 20 I=1,NNODE
    UPT(I) = DISP(NTIME,1,I)
    VPT(I) = DISP(NTIME,2,I)
    WPT(I) = DISP(NTIME,3,I)
    20 CONTINUE
    25 CONTINUE
    RETURN
    END
```

```
00003920
00003930
00003940
00003950
00003960
00003970
00003980
00003990
```

LIST OF REFERENCES

1. Earthquake Engineering Research Center report No. 73-11, SAP IV A Structural Analysis Program for Static and Dynamic Response of Linear Systems, By K. Bathe, E. L. Wilson and F. E. Peterson, June 1973 (revised April 1974).
2. Massachusetts Institute of Technology Report No. 82448-1, A Finite Element Program for Automatic Dynamic Incremental Nonlinear Analysis (ADINA), by K. Bathe, September 1975 (revised May 1976).
3. Bathe, K. J. and Wilson, E. L., Numerical Methods in Finite Element Analysis, 1st ed., p. 125, 128-32, Prentice-Hall, 1976.
4. National Aeronautics and Space Administration TN D-7797, Digital Computer Programs for Generating Oblique Orthographic Projections and Contour Plots, by Gary L. Giles, January 1975.
5. Naval Postgraduate School Technical Note No. 0211-03, Plotting Package for NPS IBM-360/67, by Patricia C. Johnson, January 1974.
6. Losh, D. M., The Implementation of a Finite Element Computer Code and Associated Pre- and Postprocessor into AE 4101 and 4102 (Flight Vehicle Structural Analysis I and II), Master of Science Thesis, Naval Postgraduate School, Monterey, CA., December 1976.
7. Naval Postgraduate School Technical Note No. 0211-5, User Libraries and Source Code Editing Under OS, by Sharon D. Raney, January 1976.
8. W. R. Church Computer Center Naval Postgraduate School, Monterey, CA., User's Manual, 1st ed., March 1970.

INITIAL DISTRIBUTION LIST

	No. Copies
1. Defense Documentation Center Cameron Station Alexandria, Virginia 22314	2
2. Library, Code 0212 Naval Postgraduate School Monterey, CA. 93940	2
3. Department Chairman, Code 69 Department of Mechanical Engineering Naval Postgraduate School Monterey, CA. 93940	1
4. Professor Gilles Cantin, Code 69Ci Department of Mechanical Engineering Naval Postgraduate School Monterey, CA. 93940	12
5. Assoc. Professor R. E. Ball, Code 67Bp Department of Aeronautics Naval Postgraduate School Monterey, CA. 93940	2
6. LT Adrian E. Kibler Jr., USN Third Ave. Hastings, PA. 16646	1
7. Georges Verchery Departement de Genie Mécanique Université de Technologie 60200 Compiègne, France	2
8. Jean Louis Armand Institut de Recherche Pour la Construction Navale 3 Avenue de Grand Champ 78230 le Pecq, France	2
9. Prof. K. J. Bathe Mechanical Engineering Department M. I. T. 77 Massachusetts Avenue Cambridge, Mass. 02139	2

	No. Copies
10. William J. Dodge Oak Ridge National Lab. Building 9204-1 Box Y Oak Ridge Tenn. 36830	1
11. Jack Tree Air Research Manufacturing Co. 402 South 36th Street P. O. Box 5217 Phoenix, Arizona 85010	1
12. Prof. Edward L. Wilson Structural Engineering Division Civil Engineering Department University of California (Berkeley) Berkeley, CA. 94720	1
13. Dr. William J. Stronge (Code 603) Naval Weapons Center China Lake, California 93955	2
14. J.E. Serpanos Code 3162 Naval Weapons Center China Lake, California 93955	1