CONTRACT REPORT ARBRL-CR-00472

AUTOMATIC INPUT DATA GENERATION FOR ADINA AND NASTRAN STRUCTURAL COMPUTER CODES

Prepared by

University of Illinois
Department of Aeronautical and
Astronautical Engineering
Urbana, Illinois 61801

October 1981

US ARMY ARMAMENT RESEARCH AND DEVELOPMENT COMMAND
BALLISTIC RESEARCH LABORATORY
ABERDEEN PROVING GROUND, MARYLAND

Destroy this report when it is no longer needed.
Do not return it to the originator.

Secondary distribution of this report by originating or sponsoring activity is prohibited.

Additional copies of this report may be obtained from the National Technical Information Service, U.S. Department of Commerce, Springfield, Virginia 22151.

The findings in this report are not to be construed as an official Department of the Army position, unless so designated by other authorized documents.

The use of trade names or manufacturers' names in this report does not constitute indorsement of any commercial product.

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE *(When Date Entered)*

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER CONTRACT REPORT ARBRL-CR-00472	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE *(and Subtitle)* AUTOMATIC INPUT DATA GENERATION FOR ADINA AND NASTRAN STRUCTURAL COMPUTER CODES		5. TYPE OF REPORT & PERIOD COVERED Contract Report JAN 1980 To FEB 1981 6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) ADAM R. ZAK		8. CONTRACT OR GRANT NUMBER(s) DAAK11-78-C-0116
9. PERFORMING ORGANIZATION NAME AND ADDRESS UNIV. OF ILLINOIS DEPT. OF AERONAUTICAL AND ASTRONAUTICAL ENGINEERING URBANA, IL 61801		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS US Army Armament Research & Development Command US Army Ballistic Research Laboratory (DRDAR-BL) Aberdeen Proving Ground, MD 21005		12. REPORT DATE October 1981 13. NUMBER OF PAGES 106
14. MONITORING AGENCY NAME & ADDRESS*(if different from Controlling Office)*		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)*

Finite Element
Structural Analysis
Mesh Generation
Adina
Nastran

20. ABSTRACT *(Continue on reverse side if necessary and identify by block number)*

An existing structural finite element code SANX was modified to generate two new programs, ADIGEN and NASGEN. The SANX program is a three-dimensional code designed for approximate structural analysis of cylindrical configurations with deviations from axial symmetry. The input procedure for SANX is relatively simple and is similar to the procedure of two-dimensional axi-symmetric codes. The function of codes ADIGEN and NASGEN is to use the input data for SANX and internally generate input data for general three-dimensional codes ADINA and NASTRAN respectively. Numerical examples are given which illustrate the use of

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE *(When Data Entered)*

the programs ADIGEN and NASGEN and the resulting saving of effort in developing data for ADINA and NASGEN.

TABLE OF CONTENTS

LIST OF ILLUSTRATIONS

I. INTRODUCTION

The objective of the investigation was to utilize the SANX structural finite element computer program to generate the input data for ADINA and NASTRAN computer codes. The SANX program[1] is a structural finite element code which has been developed for the purpose of approximate three-dimensional analysis of non-axisymmetric configurations in cylindrical coordinates. The ADINA[2] and the NASTRAN[3] codes are general finite element codes capable of handling structural analysis of multi-dimensional problems. However, the input procedure for these codes requires a great deal of time on the part of the user. On the other hand, the input procedure for the approximate three-dimensional program SANX is relatively simple since it is similar to the procedure for two-dimensional, axisymmetric codes such as SAAS.[4] After the input data for SANX is created then the program itself generates a great deal of input information internally which contains almost all the information necessary for either ADINA or NASTRAN programs except this information is not in the correct form for these codes.

The result of the present investigation is a development of a modified version of SANX which does not perform stress analysis, but rather uses its own internal finite element information to generate the input data for ADINA or NASTRAN codes. Actually, two different versions of this program have been developed, one for ADINA and one for NASTRAN. Developing two different versions was dictated by the fact that the input procedures for ADINA and NASTRAN codes are of completely different form and, consequently, it is much more convenient to generate the input separately for each program.

1. Zak, A. R., Craddock, J. N. and Drysdale, W. H., "Approximate Finite-Element Method of Stress Analysis of Non-Axisymmetric Configurations," International Journal of Computers and Structures, Vol. 9, pp. 201-206, 1978.

2. Bathe, K. J., "ADINA-A Finite Element Program for Automatic Dynamic Incremental Nonlinear Analysis," MIT Acoustics and Vibration Laboratory, Mechanical Engineering Department, Report 82448-1, 1977, Cambridge, Massachusetts.

3. McCormick, C. W., "MSC/NASTRAN Basic Training Manual," MSR-46, 15 July 1977, (MacNeal-Schwendler Corporation, 7442 N. Figueroa Street, Los Angeles, CA 90041).

4. Sawyer, S. G., "BRLESC Finite Element Program for Axisymmetric, Plane Strain, and Plane Stress, Orthotropic Solids with Temperature-Dependent Material Properties," BRL Report No. 1539 (March 1971). AD #727702

II. MODIFIED VERSIONS OF SANX

Starting with the original SANX program, two modified versions, one for ADINA and one for NASTRAN, were developed. The two programs are called ADIGEN and NASGEN, and the flow charts for these codes are shown in Figures 1 and 2, respectively. It may be noted that the two flow charts are similar except ADIGEN contains the Subroutine ADINA and NASGEN has a Subroutine NASTRN in the same place.

The first step in generating these programs was to truncate the SANX code to remove the portion which does the stress analysis and retain the parts which generate nodal, element, and loading information. Referring to Figures 1 and 2, it can be seen that this includes all the subroutines in these flow charts except ADINA and NASTRN. The latter two are the only new subroutines in ADIGEN and NASGEN, respectively. After truncating the SANX code, the only changes which were made to the modified program, besides adding the extra subroutine to each version, were in the MAIN program. The remaining subroutines left from SANX have no changes. Descriptions of ADIGEN and NASGEN programs are given in the following sections.

III. PROGRAM ADIGEN

The program ADIGEN is designed to generate three-dimensional data for ADINA program. The listing for the ADIGEN program is given in Appendix A. In the MAIN portion of the program ADIGEN, there are two basic changes from SANX. The first change is the addition of six temporary files on which the generated data for ADINA is stored. The reason for six tapes is that SANX does not generate all the nodal data, element data, etc., sequentially, but rather various portions of this data are generated by looping through each segment of the structure. However, the ADINA input data has to have the data in the specific order. The procedure is, therefore, to store various components of data on separate files and then combine these in a Procedure File which also runs the program.

The six temporary files and their functions are as follows:

TAPE2, ADNPD; nodal point data

TAPE3, ADELD; element data

TAPE4, ADMAX; material axes orientation

TAPE7, ADNPX; extra nodal information necessary for material axes orientation

TAPE8, ADMPC; material property cards

TAPE9, ADECC; element control card

The second basic change in the MAIN is the loading information of the temporary files ADMPC and ADECC. There are also some minor changes such as addition of common statement TOTAL and dimensional statement which contains two new quantities which are later set in the program.

The Subroutine ADINA is a new subroutine and essentially creates the remaining files ADNPD, ADELD, ADMAX, and ADNPX.

After the ADIGEN program is executed, then the above temporary files are assembled into one file called the ADI which represents the input data for ADINA program. The file ADI is obtained by assembling the temporary files in the following order:

ADNPD

ADNPX

ADECC

ADMPC

ADMAX

ADELD

The above order is dictated by the input requirements of the ADINA program. In order to illustrate how this procedure works the program ADIGEN was applied to a test example and the results will be presented and discussed later in this report.

IV. PROGRAM NASGEN

The program NASGEN is similar to ADIGEN in the sense that most of the changes are in the MAIN program and one new subroutine, NASTRN, has been added. The listing for the NASGEN is given in Appendix B. However, one big difference between ADIGEN and NASGEN is that the latter program contains only one additional file, TAPE1, NAS, as opposed to six for the ADIGEN. The file NAS contains the generated NASTRAN data. The reason why the NASTRAN data can be created with one file only, as opposed to six for ADINA, is that in the input procedure for NASTRAN, the data cards do not have to follow any specific order. As may be noted from the Subroutine NASTRN, each data card is started with a label which identifies the type of data on that card. The NASTRAN does its own sorting of cards.

Similarly to ADIGEN, the program NASGEN generates some of the data in the MAIN program and the rest in Subroutine NASTRN. In order to illustrate how the NASGEN program is used, a numerical test example will be given and discussed later in this report.

V. INPUT PROCEDURE FOR ADIGEN AND NASGEN PROGRAMS

The programs ADIGEN and NASGEN accept, directly, the input data for the elastic version of the SANX program. The procedure for developing the input data is described in Reference 1. In order to illustrate the procedure of using these programs, two numerical examples were executed and the corresponding input data for ADINA and NASTRAN was generated. The numerical examples are discussed below.

VI. NUMERICAL EXAMPLE - ADIGEN

The ADIGEN program was applied to a numerical example which corresponds to approximately 60° section of a non-axisymmetric cylinder. The actual configuration has no particular significance except that the data was available from previous investigations on the SANX program. The schematic of the physical configuration used in the numerical example is shown in Figure 3. The configuration is referred to a cylindrical coordinate system as required by the original SANX program.

For the purpose of the analysis the configuration is divided into four segments in the r-θ plane. This is illustrated in Figure 4. The first three segments are identical and the fourth is different. The first three segments are represented by segment Type No. 1 and the fourth by Type No. 2. The finite element models, in the r-z plane, for the two types of segments are illustrated in Figure 5 in the I-J grid configuration. The I-J grid is used in the SANX program and, therefore, in ADIGEN as the basis for numbering of the nodal points and the finite elements.

The above configuration was used in the ADIGEN program to generate the ADINA input file ADI. The listing for the ADI file is given in Appendix C. It is useful to identify various portions of this file which correspond to the various parts of the ADINA input data. In order to facilitate this discussion the start of each part of the data is identified in the right hand side by the original name of the temporary file as listed in the first part of this report.

Referring to Appendix C, the file ADI is divided into the following parts:

ADNPD Nodal Point Data

This data gives nodal point numbers and coordinates for the nodes which are later used to define finite elements. The coordinates are in the cylindrical coordinate system and, therefore, the letter X appears in front of each line. The scheme for generating numbers is as follows. Referring to Figure 4 it can be seen that the four segments

are separated by five planes. Each of these planes is in the r-z coordinate plane. The nodes in each of these planes are numbered starting from multiples of 200. For example, for the first plane the nodes start from 201, the second from 401, and the fifth from 1001. In any plane the nodes are numbered according to the I-J grid coordinate illustrated in Figure 3. The nodes are numbered in multiples of 25 depending on the value of J. For example, in the first plane, for J=1 the numbering starts from 201, for J=2 from 226, and so on. Using this scheme it is easy to identify the actual position of the node by its number relative to the r-z plane and the I-J grid.

ADNPX Extra Nodes for Defining Material Axes Orientation

For each finite element, which appears later in the ADI file, there are three extra nodes which define the material axes orientations. This is done by using number 7000, 8000, and 9000 and adding to them the element number. For example, for element 201 the three nodes are 7201, 8201, and 9201.

ADECC Element Control Card

This card is self explanatory according to the ADINA User's Manual.

ADMPC Material Property Cards

There are three materials used in the example and for each of these there are two material property cards as required by ADINA.

ADMAX Material Axes Orientation

For each element one card is generated which contains an element ID number, later repeated in the element data card, and the three nodes defining the material orientation according to the scheme required by ADINA.

ADELD Element Data

For each element there are two cards generated. The first card defines the material information, according to ID number from ADMAX data, and the second card contains eight nodal points for the three-dimensional finite element. It may be noted that the elements are numbered sequentially. This is done by starting with segment No. 1 and using the scheme from SANX in which the elements are numbered first in the I direction followed by variation in the J direction.

VII. NUMERICAL EXAMPLE - NASGEN

The NASGEN program has been applied to a numerical example which corresponds to a 90° section of a non-axisymmetric cylinder. The geometry of the cylinder used in this numerical example is shown in Figure 6. The configuration is referred to a (r, z, θ) cylindrical coordinate system.

For analysis purposes the configuration was divided into five segments in the r-θ plane. The division into the segments is shown in Figure 7. The first and fifth segments are identical, the second, third, and fourth are different but identical. The first and fifth segments are represented by segment Type No. 1, whereas the second, third, and fourth segments are represented as Type No. 2. The finite element models in the r-z plane for the two different types of segments are illustrated in Figure 8 in the I-J grid configuration. The loading on the cylinder is assumed to be composed of an axial and radial pressure of 30×10^3 psi which is illustrated as acting on the different segments shown in Figure 8. The axial pressure is assumed to cause axial acceleration which has been calculated to match the external pressure load assuming aluminum material.

The NAS data generated for this example is shown in Appendix D. It is useful to identify the various types of data in this file by referring to Appendix D. The NAS data file contains the following generated input cards for the NASTRAN program:

GRID cards;

These cards define the cylindrical coordinates for the grid points. The scheme used to number the grids is as follows. Referring to Figure 7 it can be observed that the five segments are separated by six planes in the r-z plane. The grid points in each of these planes are numbered starting from multiples of 2000. For example, starting with the plane to the left of Segment No. 1, the first grid point will be 2001. Taking these planes in geometrical order, the first grid point in the plane between Segment No. 1 and No. 2 will be 4001, and so on. In addition to this scheme, in each plane the grid points are also numbered in the multiples of 25 depending on the value of J. For example, in the first plane, for J=1 the numbering starts with 2001, for J=2 the first grid is 2026, and so on. Using this scheme it is easy to identify the position of each grid point.

CHEXA cards;

These cards define the three-dimensional, isoparametric, finite elements. Each CHEXA has a continuation card identified by the NASTRAN procedure. Each pair of cards identifies the element number, material, and the eight connecting

grid points. The scheme of numbering the elements is according to the segment number. Starting with Segment No. 1 of Figure 7, the first element is numbered 2001, in the Segment No. 2 with 4001, and so on with multiples of 2000. In the segment the elements are numbered sequentially.

PLOAD cards;

These cards define the pressure load cards relative to four grid points.

MAT1 card;

This card defines the material properties.

CORD1C card;

This card, together with three additional GRID cards numbered 101, 102, and 103, defines the cylindrical coordinate orientation relative to the basic, Cartesian coordinate system.

GRAV card;

This defines the axial acceleration load.

PSOLID card;

This card controls the output parameters for the CHEXA elements.

VIII. CONCLUDING REMARKS

This report describes two programs, ADIGEN and NASGEN, which are designed for automatic generation of input data for the general purpose finite element structural codes ADINA and NASTRAN respectively. The two programs, ADIGEN and NASGEN, have been developed by modifying SANX structural program which was previously developed at the University of Illinois. Using these two programs, numerical examples were executed. The result of these examples is input data for ADINA and NASTRAN programs. The data for the NASTRAN program was checked-out by executing the NASTRAN program. Similar check was not performed for the ADINA program since this program is not available at the University of Illinois. In conclusion, it may be observed that the two new programs, ADIGEN and NASGEN, generate a great deal of input data for the ADINA and NASTRAN codes. In each case this data was generated from approximately 20 to 30 input cards required by the SANX code. Consequently, the automatic input data generation has a great potential of reducing the necessary effort in using the ADINA and NASTRAN programs.

REFERENCES

1. Zak, A.R., Craddock, J.N. and Drysdale, W.H., "Approximate Finite-Element Method of Stress Analysis of Non-Axisymmetric Configurations," International Journal of Computers and Structures, Vol. 9, pp. 201-206, 1978.

2. Bathe, K.J., "ADINA-A Finite Element Program for Automatic Dynamic Incremental Nonlinear Analysis," MIT Acoustics and Vibration Laboratory, Mechanical Engineering Department, Report 82448-1, 1977, Cambridge, Massachusetts.

3. McCormick, C.W., "MSC/NASTRAN Basic Training Manual," MSR-46, 15 July 1977, (MacNeal-Schwendler Corporation, 7442 N. Figueroa Street, Los Angeles, CA 90041.)

4. Sawyer, S.G., "BRLESC Finite Element Program for Axisymmetric, Plane Strain, and Plane Stress, Orthotropic Solids with Temperature-Dependent Material Properties," BRL Report No. 1539 (March 1971). AD #727702

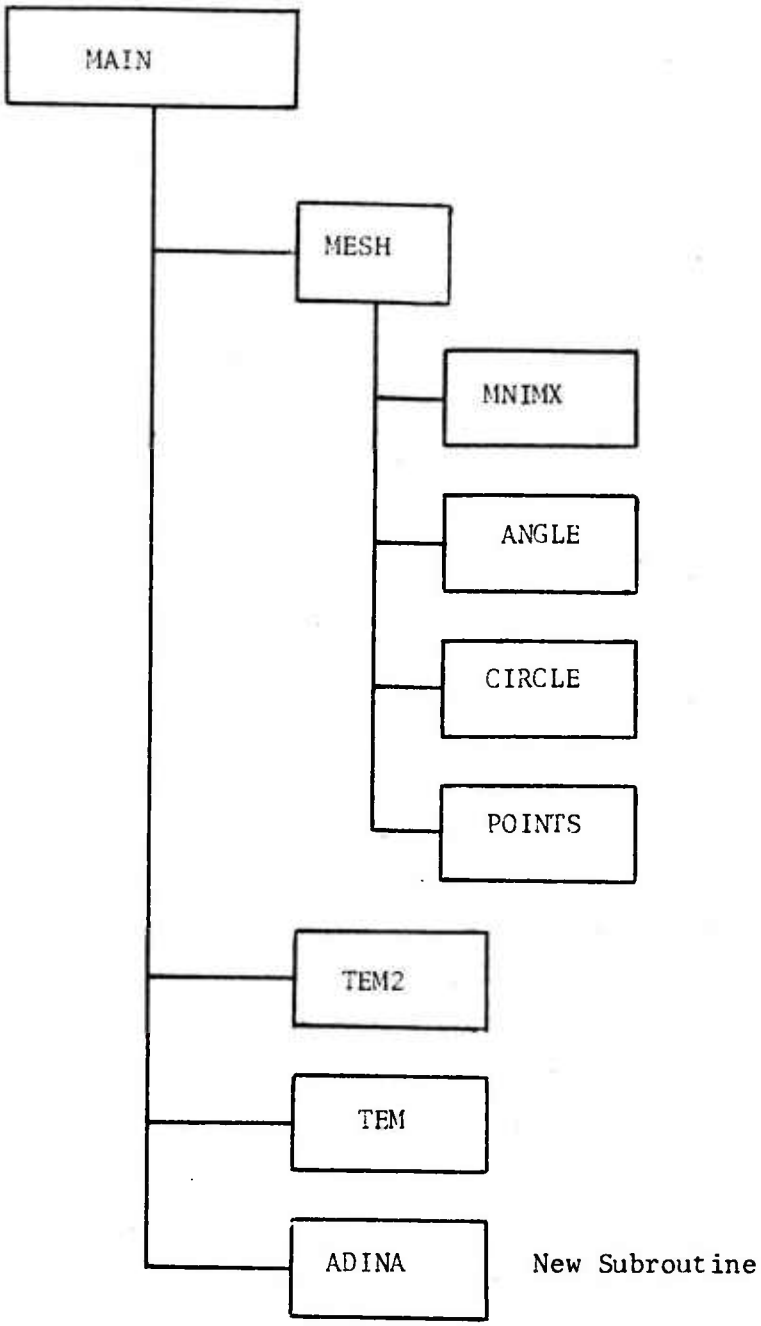

Figure 1. Flow Chart for ADIGEN Computer Program

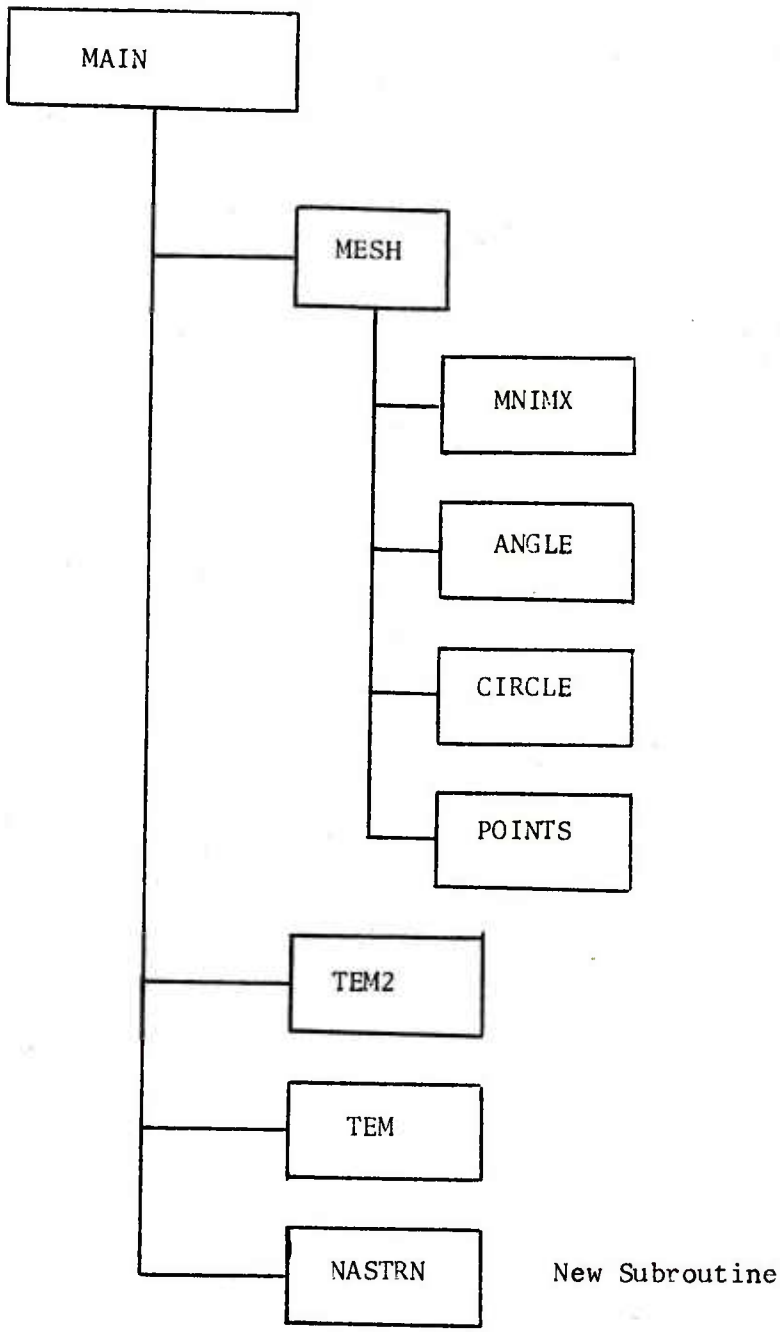

Figure 2. Flow Chart for NASGEN Computer Program

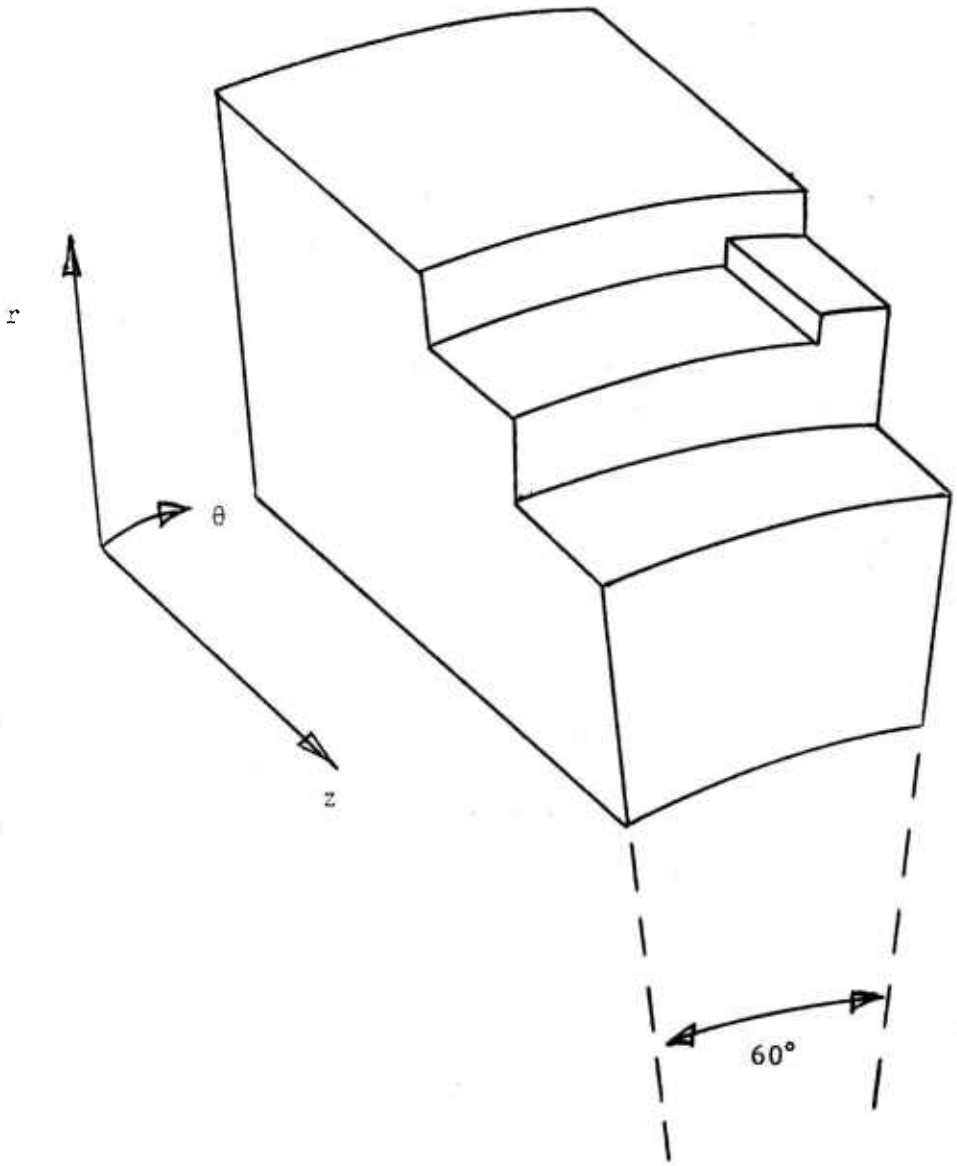

Figure 3. Cylindrical Configuration Used in the Numerical Example

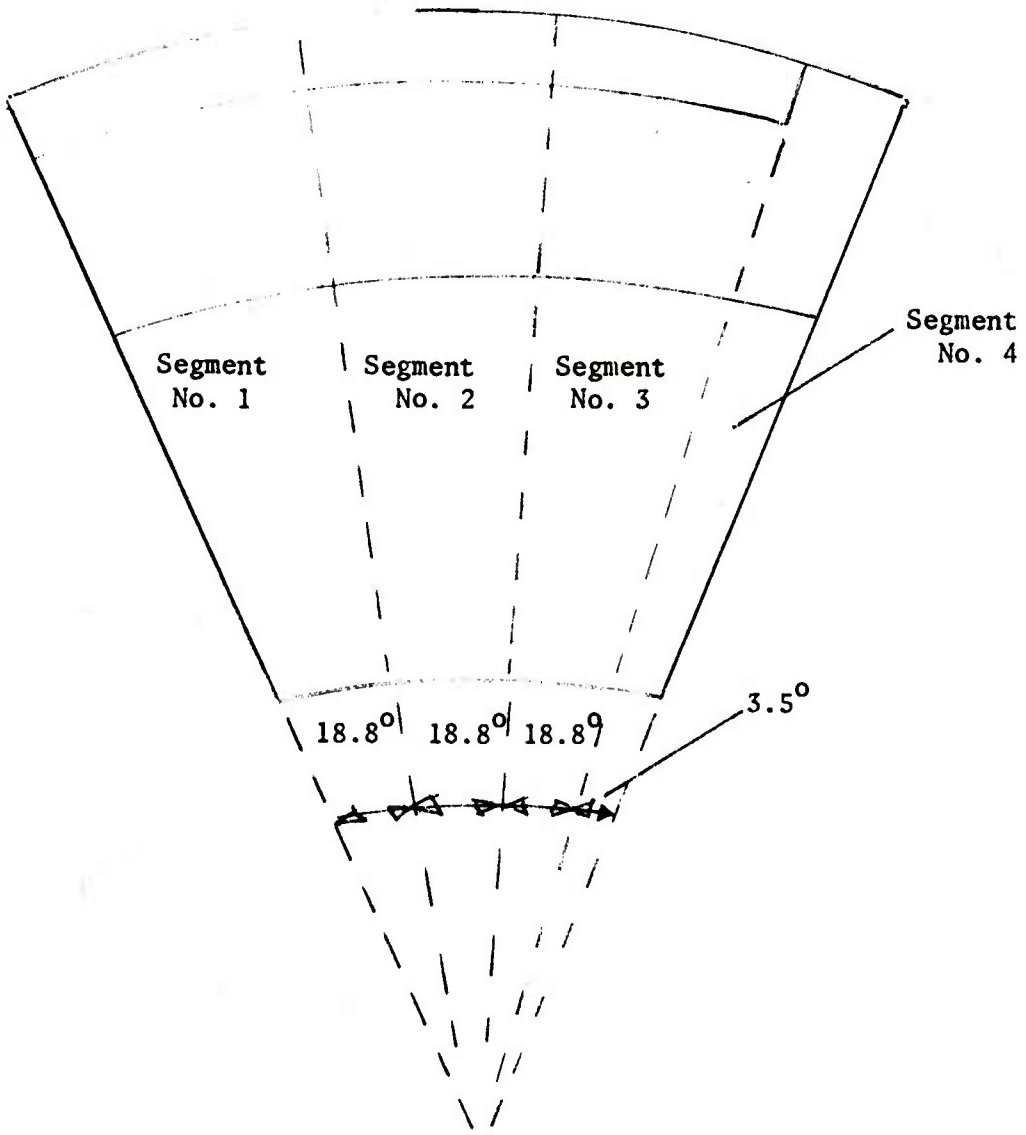

Figure 4. Division of Cylindrical Configuration into Four Segments

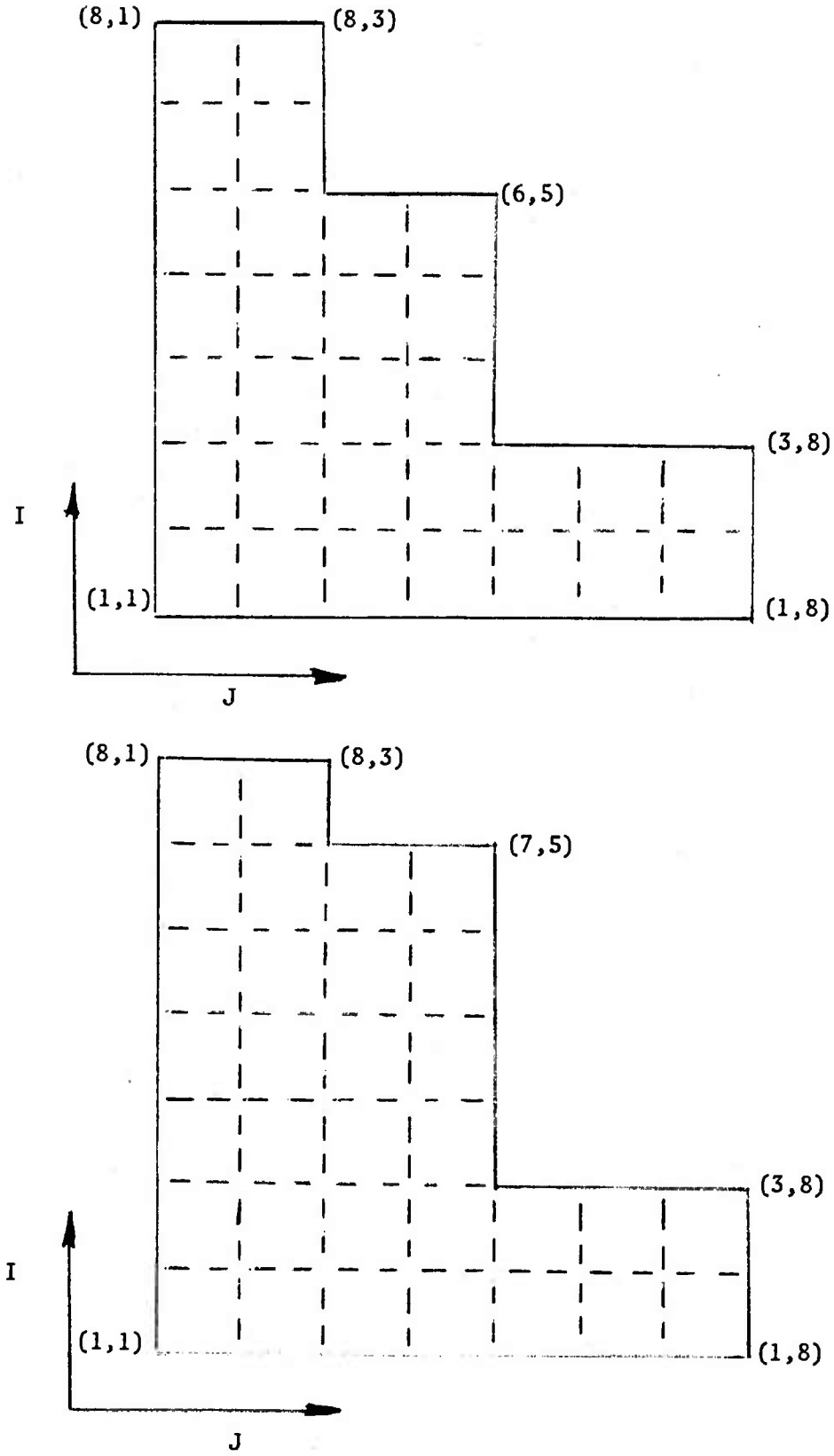

Figure 5. The I-J Grid Used in the Finite Element Model for Segment Types No. 1 and No. 2

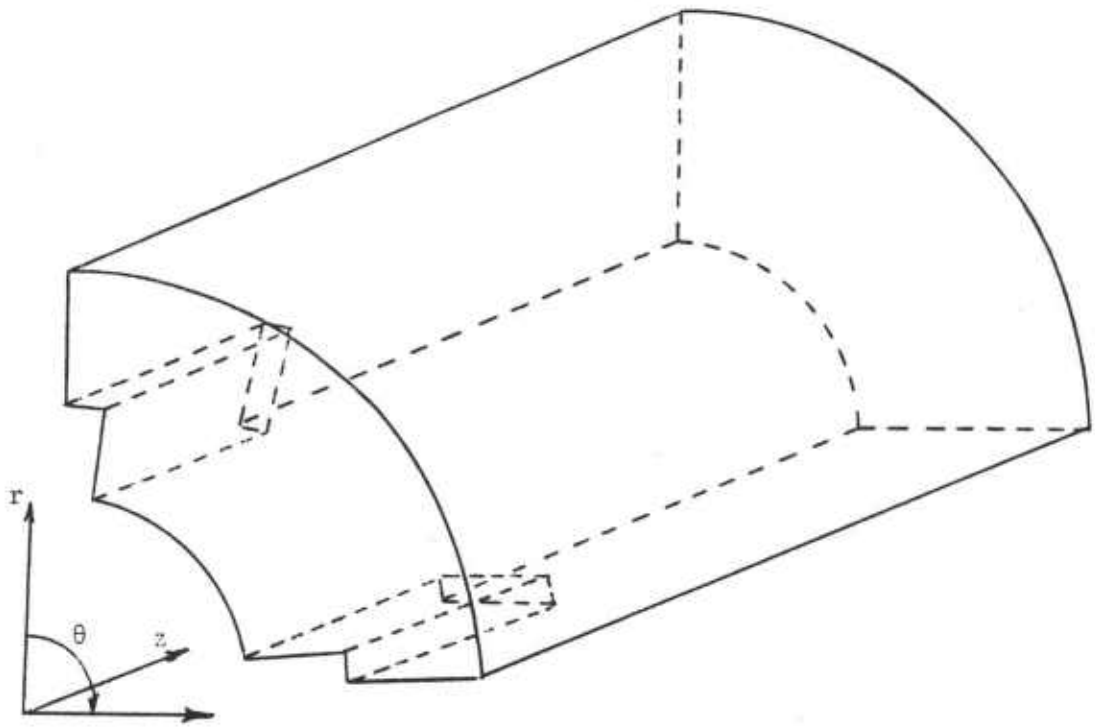

Figure 6. Nonaxisymmetric Cylindrical Configuration Used in the Numerical Example

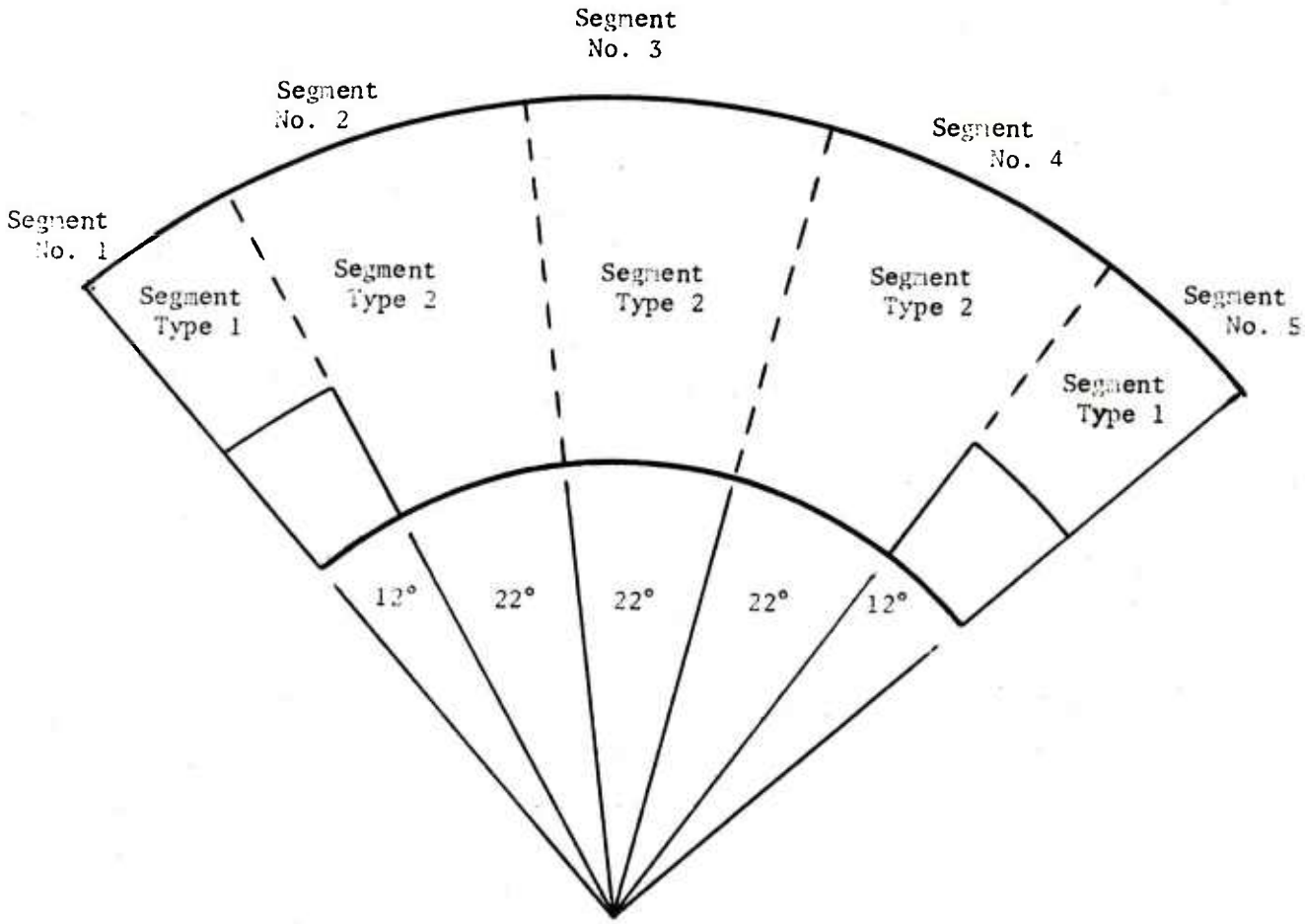

Figure 7. Division of Cylindrical Configuration Into Five Segments

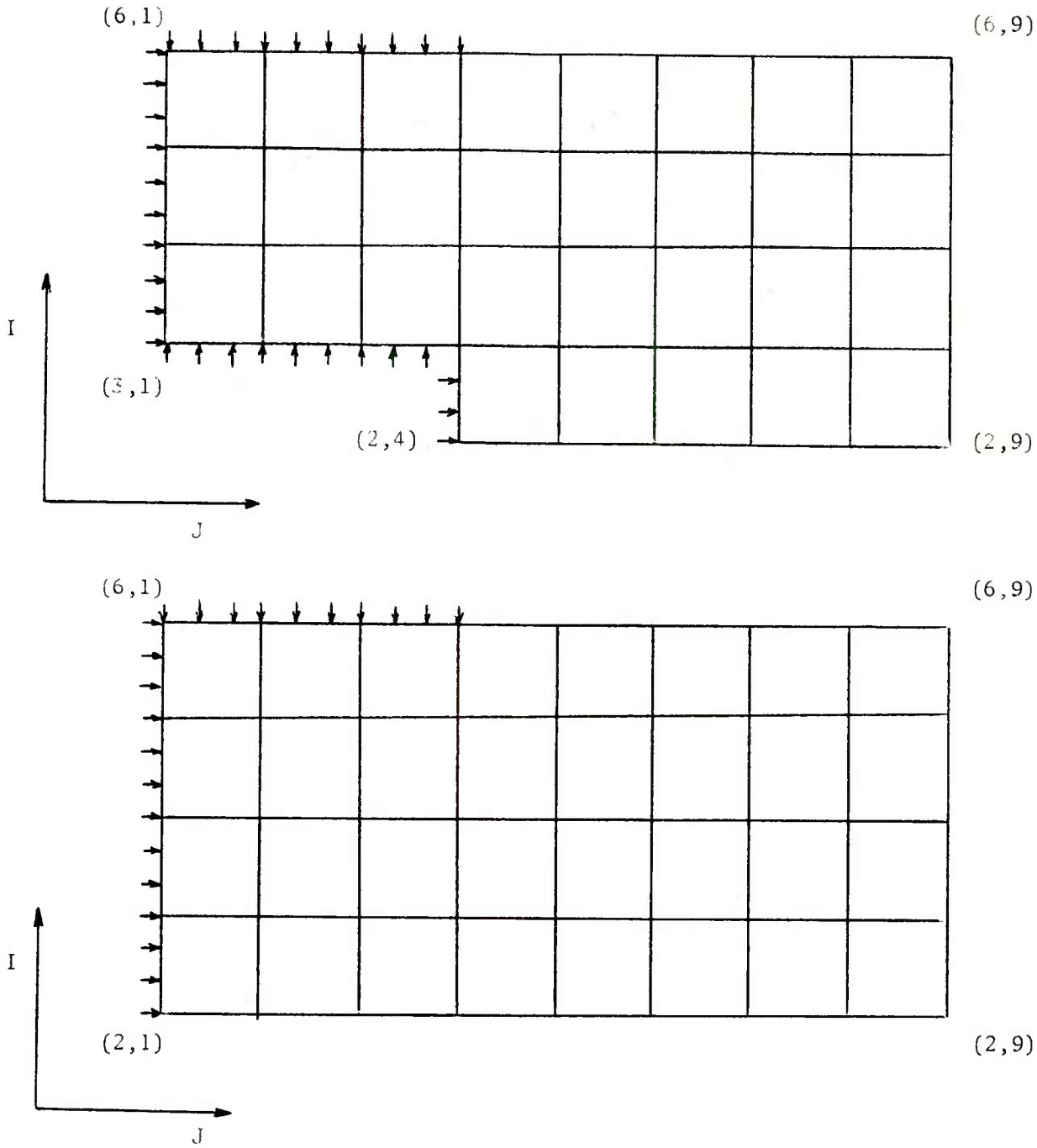

Figure 8. The I-J Grid for the Segment Types 1 and 2 and the External Pressure Load

APPENDIX A

Listing of the ADIGEN Program

```
      PROGRAM ADIGEN(INPUT,OUTPUT,ADNPD,ADNPX,ADECC,ADMPC,ADMAX,ADELD
     1,ADPRL,TAPE5=INPUT,TAPE6=OUTPUT,TAPE2=ADNPD,TAPE7=ADNPX,TAPE9=
     2 ADECC,TAPE8=ADMPC,TAPE4=ADMAX,TAPE3=ADELD,TAPE1=ADPRL)
      INTEGER CODE
      COMMON/NPDATA/R(1000),CODE(1000),XR(1000),Z(1000),XZ(1000),
     1NPNUM(25,80),T(1000),XT(1000)
      COMMON/ARG/RRR(5),ZZZ(5),RR(4),ZZ(4),S(15,15),P(15),TT(6),
     1H(6,15),CRZ(6,6),XI(10),ANGLE(4),SIG(18),EPS(18),N
      COMMON/ELDATA/BETA(1000),EPR(1000),PR(200),SH(200),IX(1000,5),
     1IP(200),JP(200),IS(200),JS(200),ALPHA(1000),IT(200),JT(200),
     2ST(200),INP(1000),JNP(1000)
      COMMON/BASIC/ACELZ,ANGVEL,ANGACC,TREF,VOL,NUMNP,NUMEL,NUMPC,NUMSC,
     1NUMST
      COMMON/NXMESH/THETAN(4),NST(4),NUMS(4,5),NPC(8,8)
      COMMON/ANS1/NUMELS(4),NUMNPS(4)
      COMMON/NXDATA/NTP,NTYPS,NTS,NTOTS
      COMMON/NONAXI/S1(30,30),P1(30),THETA,BS1(6,30)
      COMMON/SOLVE/X(4428),Y(4428),TEM(4428),NUMTC,MBAND
      COMMON/TD/IMIN(100),IMAX(100),JMIN(25),JMAX(25),MAXI,MAXJ,NMTL,NBC
      COMMON/CONVRG/IDONE
      COMMON/PLANE/NPP
      COMMON/RESULT/BS(6,15),D(6,6),C(6,6),AR,BB(6,9),CNS(6,6)
      COMMON/MATP/RO(6),E(12,16,6),EE(16),AOFTS(6)
      COMMON/TOTAL/NENL
      DIMENSION TITLE(20),FTHETA(8)
C*************************************************
C
C     WRITE ADINA DATA BLOCK HEADINGS
C
C*************************************************
      WRITE(1,5001)
      WRITE(2,5002)
      WRITE(3,5003)
      WRITE(4,5004)
      WRITE(7,5007)
      WRITE(8,5008)
      WRITE(9,5009)
 5001 FORMAT("******  3/D PRESSURE LOAD DATA          *****")
 5002 FORMAT("******  NODAL POINT DATA                ******")
 5003 FORMAT("******  3/D ELEMENT DATA                ******")
 5004 FORMAT("******  MATERIAL AXES ORIENTATION       ******")
 5007 FORMAT("******  EXTRA NODES FOR MATERIAL AXES   ******")
 5008 FORMAT("******  MATERIAL PROPERTY CARDS         ******")
 5009 FORMAT("******  ELEMENT COTROL CARD             ******")
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C     READ AND WRITE CONTROL INFORMATION
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
      READ(5,3000) NTYPS,NTOTS
      DO 150 I=1,NTYPS
  150 READ(5,3001) THETAN(I),NST(I)
      DO 151 I=1,NTYPS
      J2 =NST(I)
```

```
  151 READ(5,3000) (NUMS(I,J),J=1,J2)
      DO 152  I= 1,NTOTS
  152 READ(5,3002) (NPC(I,J),J=1,8)
 3000 FORMAT(8I10)
 3001 FORMAT(F10.5,I10)
 3002 FORMAT(8I10)
      NFACE=NTOTS+1
      FTHETA(1)=0.0
      DO 4100 I=1,NTOTS
      DO 4000 J=1,NTYPS
      K2=NST(J)
      DO 4000 K=1,K2
 4000 IF(NUMS(J,K).EQ.I)GO TO 4100
 4100 FTHETA(I+1)=FTHETA(I)+THETAN(J)
      WRITE(6,4200)((FTHETA(I),I),I=1,NTOTS)
 4200 FORMAT("1"," ANGLE IN DEGREES ",5X,"  BEGINNING SEGMENT #"/
     1 (E20.10,I10))
       WRITE(6,3010)
 3010 FORMAT("1","SEGMENT DATA FOR NONAXISYMMETRIC PROBLEM")
       WRITE(6,3011) NTYPS,NTOTS
 3011 FORMAT (" "," NUMBER OF TYPES OF SEGMENTS = ",I5,//,
     1            " NUMBER OF TOTAL SEGMENTS     =",I5)
       DO 153 I=1,NTYPS
      WRITE(6,3012) I,THETAN(I),NST(I)
 3012 FORMAT(" ",///," SEGMENT TYPE = ",I5/,"      THETA = ",F10.5/,
     1  "  NUMBER OF SEGMENTS OF THIS TYPE = ",I5)
       J2 = NST(I)
       WRITE(6,3013)(NUMS(I,J),J=1,J2)
 3013 FORMAT(" "," SEGMENT NUMBERS IN GLOBAL SYSTEM ARE ",5I5)
  153 CONTINUE
       DO 154  I=1,NTOTS
  154  WRITE(6,3014)I,(NPC(I,J),J=1,8)
 3014 FORMAT("  ","CONNECTING NODES FOR SEGMENT",I5," ARE",8I5)
      NENL=0
      DO 950 NTP = 1,NTYPS
       THETA= THETAN(NTP)       /57.295780
   50 READ(5,1000 )TITLE,NNLA,NUMTC,NUMMAT,NUMPC,NUMSC,NUMST,TREF
     1,INERT,NLINC,INCI,INCF,IPLOT
      WRITE(6,2000)TITLE,NNLA,NUMTC,NUMMAT,NUMPC,NUMSC,NUMST,TREF,INERT,
     1NLINC
      NPP=0
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C     GENERATE FINITE ELEMENT MESH
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
  100 CALL MESH
       NUMELS(NTP) = NUMEL
       NUMNPS(NTP) = NUMNP
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C     READ AND WRITE T5MPERATURE DATA
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
  103 IF(NUMTC.EQ.0) GO TO 440
      IF(NUMTC.GT.0) READ(5,1001) (X(I),Y(I),TEM(I),I=1,NUMTC)
```

```
      IF(NUMTC.EQ.-2) CALL TEM2(NUMNP)
      IF(NUMTC.EQ.-2) GO TO 440
      MPRINT=0
      DO 210 I=1,NUMTC
      IF(MPRINT.NE.0) GO TO 200
      WRITE(6,2001)
      MPRINT=59
  200 MPRINT=MPRINT-1
  210 WRITE(6,2002) X(I),Y(I),TEM(I)
      MPRINT=0
      DO 230 N=1,NUMNP
      IF(MPRINT.NE.0) GO TO 220
      WRITE(6,2003)
      MPRINT=59
  220 MPRINT=MPRINT-1
      CALL TEMP(R(N),Z(N),T(N))
  230 WRITE(6,2004) N,R(N),Z(N),T(N)
  440 MPRINT=0
      DO 460 N=1,NUMEL
      IF(MPRINT.NE.0) GO TO 450
      WRITE(6,2008)
      MPRINT=59
  450 MPRINT=MPRINT-1
      II=IX(N,1)
      JJ=IX(N,2)
      KK=IX(N,3)
      LL=IX(N,4)
C
C     TEM IS TEMPORARY STORAGE FOR ELEMENT TEMPERATURES
C
      TEM(N)=(T(II)+T(JJ)+T(KK)+T(LL))/4.00
  460 WRITE(6,2009) N,(IX(N,I),I=1,5),BETA(N),ALPHA(N),TEM(N)
      DO 470 K=1,NUMEL
  470 T(K)=TEM(K)
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C     READ AND WRITE MATERIAL PROPERTIES
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
  500 CONTINUE
      DO 510 M=1,NUMMAT
      READ(5,1004)  MTYPE,(NT,RO(MTYPE),AOFTS(MTYPE))
      WRITE(6,2010) MTYPE,NT,RO(MTYPE)
      READ(5,1005)((E(I,J,MTYPE),J=1,14),I=1,NT)
      IF(AOFTS(MTYPE).NE.1.) WRITE(6,2011)((E(I,J,MTYPE),J=1,13),I=1,NT)
      IF(AOFTS(MTYPE).EQ.1.) WRITE(6,2012)((E(I,J,MTYPE),J=1,13),I=1,NT)
      DO 510 I=NT,12
      DO 510 J=1,16
  510 E(I,J,MTYPE)=E(NT,J,MTYPE)
      DO 900 NL=1,NLINC
      WRITE(6,2030) NL
      ACELZ=0.00
      ANGVEL=0.00
      ANGACC=0.00
```

```
      IF(INERT .EQ. 0) GO TO 511
      IF(NL .NE. 1 .AND. INCI .EQ. 0) GO TO 511
C*************************************************************
C     READ AND WRITE DYNAMIC FORCES
C*************************************************************
      READ(5,1030) ACELZ, ANGVEL, ANGACC
      WRITE(6,2031) ACELZ, ANGVEL, ANGACC
  511 CONTINUE
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C     READ AND WRITE PRESSURE AND SHEAR BOUNDARY CONDITIONS
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
      IF(NL .NE. 1 .AND. INCF .EQ. 0) GO TO 700
  600 IF(NUMPC.EQ.0) GO TO 630
      MPRINT=0
      DO 620 L=1,NUMPC
      IF(MPRINT.NE.0) GO TO 610
      WRITE(6,2013)
      MPRINT=58
  610 MPRINT=MPRINT-1
      READ(5,1006)  IP(L),JP(L),PR(L)
  620 WRITE(6,2014) IP(L),JP(L),PR(L)
  630 IF(NUMSC.EQ.0) GO TO 701
      MPRINT=0
      DO 650 L=1,NUMSC
      IF(MPRINT.NE.0) GO TO 640
      WRITE(6,2015)
      MPRINT=58
  640 MPRINT=MPRINT-1
      READ(5,1006)  IS(L),JS(L),SH(L)
  650 WRITE(6,2014) IS(L),JS(L),SH(L)
  701 IF(NUMST.EQ.0) GO TO 700
      MPRINT=0
      DO 680 L=1,NUMST
      IF(MPRINT.NE.0) GO TO 670
      WRITE(6,2025)
      MPRINT=58
  670 MPRINT=MPRINT-1
      READ(5,1006) IT(L),JT(L),ST(L)
  680 WRITE(6,2014)IT(L),JT(L),ST(L)
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C     DETERMINE BANDWIDTH, INITIALIZE ELASTIC-PLASTIC RATIO,
C     AND CONVERT BETA FROM DEGREES TO RADIANS
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
  700 J=0
      DO 710 N=1,NUMEL
      IX(N,5)=IABS(IX(N,5))
      DO 710 I=1,4
      DO 710 L=1,4
      KK=IABS(IX(N,I)-IX(N,L))
      IF(KK.GE.J) J=KK
  710 CONTINUE
      MBAND=3*J+3
```

```
      IF(NL.GT.1) GO TO 721
      DO 720 N=1,NUMEL
      EPR(N)=1.
      ALPHA(N)=ALPHA(N)/57.295780
  720 BETA(N)=BETA(N)/57.295780
  721 CONTINUE
  900 CONTINUE
      CALL ADINA(NFACE,FTHETA)
  950 CONTINUE
C
C     WRITE ELEMENT CONTROL ON FILE 9
      NZ=0
C
      WRITE(9,4400) NENL,NZ,NZ,NZ,NUMMAT,NZ,NENL,NZ
 4400 FORMAT(3X,"3",3I4,11X,"8",20X,I4,7X,"2",3I4,4X,I4)
C
C     WRITE MATERIAL PROPERTIES ON FILE 8
C
      DO 960 I=1,NUMMAT
      WRITE(8,4500)I,RO(I)
 4500 FORMAT(I5,F10.5)
      WRITE(8,4600) (E(1,J,I),J=2,5),E(1,7,I),E(1,6,I),
     1E(1,8,I),E(1,10,I)
      WRITE(8,4600) E(1,9,I)
  960 CONTINUE
 4600 FORMAT(3F10.0,3F10.3,2F10.0)
  910 CONTINUE
 1000 FORMAT(20A4/6I5,F5.0,5I5)
 1001 FORMAT(3F10.0)
 1004 FORMAT (2I5,2F10.0)
 1005 FORMAT(7F10.0)
 1006 FORMAT (2I5,F10.0)
 1030 FORMAT(3F10.0)
 2000 FORMAT (2H1 ,20A4/
     1 33H0  NUMBER OF APPROXIMATIONS------I4/
     2 33H0  NUMBER OF TEMPERATURE CARDS---I4/
     3 33H0  NUMBER OF MATERIALS----------I4/
     4 33H0  NUMBER OF PRESSURE CARDS------I4/
     5 33H0  NUMBER OF SHEAR CARDS---------I4/
     6 33H0  NUMBER OF TORSION CARDS-------I4/
     7 33H0  REFERENCE TEMPERATURE---------E12.4/
     8 33H0  NUMBER OF INERTIA CARDS-------I4/
     9 33H0  NUMBER OF LOAD INCREMENTS-----I4/)
 2001 FORMAT (1H1,13X,1HR,14X,1HZ,14X,1HT)
 2002 FORMAT (3F15.3)
 2003 FORMAT (35H1   N      R          Z            T)
 2004 FORMAT (I5,2F10.4,F10.0)
 2008 FORMAT (74H1  EL   I   J   K  L    MATERIAL   ANGLE BETA   ANGLE A
     1LPHA    TEMPERATURE)
 2009 FORMAT (I5,4I4,I8,F11.1,2F13.3)
 2010 FORMAT (1H1,"MATERIAL IDENTIFICATION NUMBER =",I2/
     11H ,"NO. OF MATERIAL TEMPERATURE CARDS =",I2/
```

```
     21H ,"MASS DENSITY =",E15.7)
 2011 FORMAT (1H ,"TEMPERATURE =",E15.7/
     11H ,"MODULUS OF ELASTICITY-EN =",E15.7/
     21H ,"MODULUS OF ELASTICITY-ES =",E15.7/
     31H ,"MODULUS OF ELASTICITY-ET =",E15.7/
     41H ,"POISSON RATIO-NUNS =",E15.7/
     51H ,"POISSON RATIO-NUNT =",E15.7/
     61H ,"POISSON RATIO-NUST =",E15.7/
     71H ,"SHEAR MODULUS-GNS =",E15.7/
     81H ,"SHEAR MODULUS-GST =",E15.7/
     91H ,"SHEAR MODULUS-GTN =",E15.7/
     11H ,"COEFFICIENT OF THERMAL EXPANSION-AN =",E15.7/
     21H ,"COEFFICIENT OF THERMAL EXPANSION-AS =",E15.7/
     31H ,"COEFFICIENT OF THERMAL EXPANSION-AT =",E15.7/)
 2012 FORMAT (1H ,"TEMPERATURE =",E15.7/
     11H ,"MODULUS OF ELASTICITY-EN =",E15.7/
     21H ,"MODULUS OF ELASTICITY-ES =",E15.7/
     31H ,"MODULUS OF ELASTICITY-ET =",E15.7/
     41H ,"POISSON RATIO-NUNS =",E15.7/
     51H ,"POISSON RATIO-NUNT =",E15.7/
     61H ,"POISSON RATIO-NUST =",E15.7/
     71H ,"SHEAR MODULUS-GNS =",E15.7/
     81H ,"SHEAR MODULUS-GST =",E15.7/
     91H ,"SHEAR MODULUS-GTN =",E15.7/
     11H ,"FREE THERMAL STRAIN-FN =",E15.7/
     21H ,"FREE THERMAL STRAIN-FS =",E15.7/
     31H ,"FREE THERMAL STRAIN-FT =",E15.7/)
 2013 FORMAT (30H1 PRESSURE BOUNDARY CONDITIONS/20H    I    J  PRESSURE)
 2014 FORMAT (2I5,F10.1)
 2015 FORMAT (27H1 SHEAR BOUNDARY CONDITIONS/17H    I    J  SHEAR)
 2016 FORMAT (26H  THE SYSTEM CONVERGED IN I2,11H ITERATIONS)
 2017 FORMAT (33H  THE SYSTEM DID NOT CONVERGE IN I2,11H ITERATIONS)
 2024 FORMAT (43H0 THE AXISYMMETRIC OPTION NAS BEEN SELECTED)
 2025 FORMAT(30H1  TORSION BOUNDARY CONDITIONS/17H    I    J  SHEAR)
 2030 FORMAT(1H1,"LOAD STEP=",I4)
 2031 FORMAT(1H0 ,"AXIAL ACCELERATION  =",E12.4/
     11H0 ,"ANGULAR VELOCITY    =",E12.4/
     21H0 ,"ANGULAR ACCELERATION=",E12.4)
  920 STOP
      END
      SUBROUTINE ANGLE (R,Z,RC,ZC,ANG)
C     FIND ANGLE OF INCLINATION BETWEEN 0 AND 2*PI
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
      PI=3.1415927
      D1=(Z-ZC)
      D2=(R-RC)
      IF(ABS(R-RC).GT.1.E-8) GO TO 100
      ANG=PI/2.
      IF(D1.GT.1.E-8) RETURN
      ANG=-ANG
      RETURN
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
```

```
C     ALLOW CIRCLE TO CROSS AXIS
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
  100 ANG=ATAN2(D1,D2)
      RETURN
      END
      SUBROUTINE CIRCLE(ANG1,DELPHI,RSTRT,ZSTRT,RC,ZC,I,J)
      INTEGER CODE
      COMMON/TD/IMIN(100),IMAX(100),JMIN(25),JMAX(25),MAXI,MAXJ,NMTL,NBC
      COMMON/NPDATA/R(1000),CODE(1000),XR(1000),Z(1000),XZ(1000),
     1NPNUM(25,80),T(1000),XT(1000)
      DIMENSION AR(25,80),AZ(25,80)
      EQUIVALENCE (R(1),AR),(Z(1),AZ)
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C     FIND INTERSECTION OF LINE AND CIRCLE = NEW R AND Z
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
      ANG1=ANG1+DELPHI
      RR=SQRT((RSTRT-RC)**2+(ZSTRT-ZC)**2)
      AR(I,J)=RC+RR*COS(ANG1)
      AZ(I,J)=ZC+RR*SIN(ANG1)
      RETURN
      END
      SUBROUTINE MESH
      INTEGER CODE
      DIMENSION AR(25,80),AZ(25,80),NCODE(25,80)
      COMMON/TD/IMIN(100),IMAX(100),JMIN(25),JMAX(25),MAXI,MAXJ,NMTL,NBC
      COMMON/NPDATA/R(1000),CODE(1000),XR(1000),Z(1000),XZ(1000),
     1NPNUM(25,80),T(1000),XT(1000)
      COMMON/ELDATA/BETA(1000),EPR(1000),PR(200),SH(200),IX(1000,5),
     1IP(200),JP(200),IS(200),JS(200),ALPHA(1000),IT(200),JT(200),
     2ST(200),INP(1000),JNP(1000)
      EQUIVALENCE (R(1),AR),(Z(1),AZ),(IX(1,1),NCODE)
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C     MESH CONTROL INFORMATION
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
      READ (5,1000) MAXI,MAXJ,NSEG,NBC,NMTL
      WRITE(6,2000) MAXI,MAXJ,NSEG,NBC,NMTL
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C     INITIALIZE
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
      ISEG=-1
      PI=3.1415927
      DO 110 J=1,100
      DO 100 I=1,25
      NCODE(I,J)=0
      AR(I,J)=0.
      AZ(I,J)=0.
      JMAX(I)=0
  100 JMIN(I)=MAXI
      IMIN(J)=MAXJ
  110 IMAX(J)=0
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C     LINE SEGMENT CARDS
```

```
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
  150 ISEG=ISEG+1
  159 IF(ISEG.EQ.NSEG) GO TO 400
      READ(5,1001) I1,J1,R1,Z1,I2,J2,R2,Z2,I3,J3,R3,Z3,IPTION
      WRITE(6,2001)I1,J1,R1,Z1,I2,J2,R2,Z2,I3,J3,R3,Z3,IPTION
      IPTION=IPTION+1
      AR(I1,J1)=R1
      AZ(I1,J1)=Z1
      NCODE(I1,J1)=1
      CALL MNIMX(I1,J1)
      GO TO (150,200,200,300,300,200,200), IPTION
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C     GENERATE STRAIGHT LINES ON BOUNDARY
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
200   DI= ABS(FLOAT(I2-I1))
      DJ= ABS(FLOAT(J2-J1))
      AR(I2,J2)=R2
      AZ(I2,J2)=Z2
      NCODE(I2,J2)=1
      CALL MNIMX(I2,J2)
      ISTRT=I1
      ISTP=I2
      JSTRT=J1
      JSTP=J2
      DIFF=MAX1(DI,DJ)
      ITER=DIFF-1.
      IINC=0
      JINC=0
      IF(I2.NE.I1) IINC=(I2-I1)/IABS(I2-I1)
      IF(J2.NE.J1) JINC=(J2-J1)/IABS(J2-J1)
      KAPPA=1
      IF(I2.NE.I1.AND.J2.NE.J1.AND.IPTION.NE.3) KAPPA=2
      IF(KAPPA.EQ.2) DIFF=2.*DIFF
      RINC=(R2-R1)/DIFF
      ZINC=(Z2-Z1)/DIFF
      WRITE(6,2002) DI,DJ,DIFF,RINC,ZINC,ITER,IINC,JINC,KAPPA
C
C     CHECK FOR INPUT ERROR
C
      IF(KAPPA.NE.2.OR.DI.EQ.DJ) GO TO 210
      WRITE(6,2003)
      GO TO 150
C
C     INTERPOLATE
C
  210 I=I1
      J=J1
      WRITE(6,2004)
      DO 230 M=1,ITER
      IF(ITER.EQ.0.AND.IPTION.EQ.2) GO TO 230
      IF(ITER.EQ.0.AND.IPTION.EQ.6) GO TO 230
      IF(ITER.EQ.0.AND.IPTION.EQ.7) GO TO 230
```

```
      IF(KAPPA.EQ.2) GO TO 220
      IOLD=I
      I=I+IINC
      JOLD=J
      J=J+JINC
      AR(I,J)=AR(IOLD,JOLD)+RINC
      AZ(I,J)=AZ(IOLD,JOLD)+ZINC
      WRITE(6,2005) I,J,AR(I,J),AZ(I,J)
      CALL MNIMX(I,J)
      NCODE(I,J)=1
      GO TO 230
  220 CONTINUE
      IF(I1.GT.I2.AND.IPTION.EQ.7) GO TO 221
      IF(I1.LT.I2.AND.IPTION.EQ.6) GO TO 221
      IOLD=I
      I=I+IINC
      AR(I,J)=AR(IOLD,J)+RINC
      AZ(I,J)=AZ(IOLD,J)+ZINC
      WRITE(6,2005) I,J,AR(I,J),AZ(I,J)
      NCODE(I,J)=1
      CALL MNIMX(I,J)
      JOLD=J
      J=J+JINC
      AR(I,J)=AR(I,JOLD)+RINC
      AZ(I,J)=AZ(I,JOLD)+ZINC
      NCODE(I,J)=1
      WRITE(6,2005) I,J,AR(I,J),AZ(I,J)
      CALL MNIMX(I,J)
      GO TO 230
  221 JOLD=J
      J=J+JINC
      AR(I,J)=AR(I,JOLD)+RINC
      AZ(I,J)=AZ(I,JOLD)+ZINC
      NCODE(I,J)=1
      WRITE(6,2005) I,J,AR(I,J),AZ(I,J)
      CALL MNIMX(I,J)
      IOLD=I
      I=I+IINC
      AR(I,J)=AR(IOLD,J)+RINC
      AZ(I,J)=AZ(IOLD,J)+ZINC
      NCODE(I,J)=1
      WRITE(6,2005) I,J,AR(I,J),AZ(I,J)
      CALL MNIMX(I,J)
  230 CONTINUE
      IF(KAPPA.EQ.1) GO TO 150
      IF(I1.GT.I2.AND.IPTION.EQ.7) GO TO 231
      IF(I1.LT.I2.AND.IPTION.EQ.6) GO TO 231
      IOLD=I
      I=I+IINC
      AR(I,J)=AR(IOLD,J)+RINC
      AZ(I,J)=AZ(IOLD,J)+ZINC
      GO TO 232
```

```
  231 CONTINUE
      JOLD=J
      J=J+JINC
      AR(I,J)=AR(I,JOLD)+RINC
      AZ(I,J)=AZ(I,JOLD)+ZINC
  232 CONTINUE
      NCODE(I,J)=1
      WRITE(6,2005) I,J,AR(I,J),AZ(I,J)
      CALL MNIMX(I,J)
      GO TO 150
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C     GENERATE CIRCULAR ARCS ON BOUNDARY
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
  300 AR(I2,J2)=R2
      AZ(I2,J2)=Z2
      NCODE(I2,J2) = 1
      CALL MNIMX(I2,J2)
      IF(IPTION.EQ.5) GO TO 320
C
C     FIND CENTER OF CIRCLE
C
      AR(I3,J3)=R3
      AZ(I3,J3)=Z3
      NCODE(I3,J3)=1
      CALL MNIMX(I3,J3)
      SLAC=(Z2-Z1)/(R2-R1)
      SLBF=-1./SLAC
      SLCE=(Z3-Z2)/(R3-R2)
      SLDF=-1./SLCE
C
C     CHECK FOR INPUT ERROR
C
      IF(ABS(SLAC-SLCE).GT..001) GO TO 310
      WRITE(6,2006) R1,Z1,R2,Z2,R3,Z3,SLAC,SLCE
      GO TO 150
  310 R4=R1+(R2-R1)/2.
      Z4=Z1+(Z2-Z1)/2.
      R5=R2+(R3-R2)/2.
      Z5=Z2+(Z3-Z2)/2.
      BBF=Z4-SLBF*R4
      BDF=Z5-SLDF*R5
      RC=(BBF-BDF)/(SLDF-SLBF)
      ZC=SLBF*RC+BBF
      WRITE(6,2007) RC,ZC
      KAPPA=1
      GO TO 330
  320 KAPPA=2
      RC=R3
      ZC=Z3
  330 ISTRT=I1
      ISTP=I2
      JSTRT=J1
```

```
      JSTP=J2
      RSTRT=R1
      RSTP=R2
      ZSTRT=Z1
      ZSTP=Z2
  340 CALL ANGLE(RSTRT,ZSTRT,RC,ZC,ANG1)
      CALL ANGLE(RSTP,ZSTP,RC,ZC,ANG2)
      IF(ANG2.LE.ANG1)  ANG2=2.0*PI+ANG2
C
C     FIND ANGULAR INCREMENT
C
      DI= ABS(FLOAT(ISTP-ISTRT))
      DJ= ABS(FLOAT(JSTP-JSTRT))
      IINC=0
      JINC=0
      IF(ISTRT.NE.ISTP) IINC=(ISTP-ISTRT)/IABS(ISTP-ISTRT)
      IF(JSTRT.NE.JSTP) JINC=(JSTP-JSTRT)/IABS(JSTP-JSTRT)
      LAMDA=1
      IF(IINC.NE.0.AND.JINC.NE.0) LAMDA=2
      DIFF=MAX1(DI,DJ)
      ITER=DIFF-1.
      IF(LAMDA.EQ.2) DIFF=2.*DIFF
      DELPHI=(ANG2-ANG1)/DIFF
      WRITE(6,2008) ANG1,ANG2,DIFF,DELPHI
C
C     CHECK FOR INPUT ERROR
C
      IF(LAMDA.NE.2.OR.DI.EQ.DJ) GO TO 350
      WRITE(6,2003)
      GO TO 150
  350 IO=ISTRT
      JO=JSTRT
      WRITE(6,2004)
C
C     INTERPOLATE
C
      NPT=IABS(I2-I1)+IABS(J2-J1)-1
      DO 380 M=1,ITER
  359 IF(LAMDA.EQ.2) GO TO 360
      I=IO+IINC
      J=JO+JINC
      CALL MNIMX(I,J)
      NCODE(I,J)=1
      CALL CIRCLE(ANG1,DELPHI,RSTRT,ZSTRT,RC,ZC,I,J)
      WRITE(6,2005) I,J,AR(I,J),AZ(I,J)
      GO TO 370
  360 I=IO+IINC
      J=JO
      NCODE(I,J)=1
      CALL MNIMX(I,J)
      CALL CIRCLE(ANG1,DELPHI,RSTRT,ZSTRT,RC,ZC,I,J)
      WRITE(6,2005) I,J,AR(I,J),AZ(I,J)
```

```
      J=JO+JINC
      NCODE(I,J)=1
      CALL MNIMX(I,J)
      CALL CIRCLE(ANG1,DELPHI,RSTRT,ZSTRT,RC,ZC,I,J)
      WRITE(6,2005) I,J,AR(I,J),AZ(I,J)
  370 IO=I
  380 JO=J
      IF(LAMDA.NE.2) GO TO 390
      I=IO+IINC
      NCODE(I,J)=1
      CALL MNIMX(I,J)
      CALL CIRCLE(ANG1,DELPHI,RSTRT,ZSTRT,RC,ZC,I,J)
      WRITE(6,2005) I,J,AR(I,J),AZ(I,J)
  390 IF(KAPPA.EQ.2) GO TO 150
      ISTRT=I2
      ISTP=I3
      JSTRT=J2
      JSTP=J3
      RSTRT=R2
      RSTP=R3
      ZSTRT=Z2
      ZSTP=Z3
      KAPPA=2
  399 GO TO 340
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C     CALCULATE COORDINATES OF INTERIOR POINTS
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
  400 IF(MAXJ.LE.2) GO TO 430
      J2=MAXJ-1
      DO 420 N=1,500
      RESID=0.
      DO 410 J=2,J2
      I1=IMIN(J)+1
      I2=IMAX(J)-1
      DO 410 I=I1,I2
      IF(NCODE(I,J).EQ.1) GO TO 410
      DR=(AR(I+1,J)+AR(I-1,J)+AR(I,J+1)+AR(I,J-1))/4.-AR(I,J)
      DZ=(AZ(I+1,J)+AZ(I-1,J)+AZ(I,J+1)+AZ(I,J-1))/4.-AZ(I,J)
      RESID=RESID+ABS(DR)+ABS(DZ)
      AR(I,J)=AR(I,J)+1.8*DR
      AZ(I,J)=AZ(I,J)+1.8*DZ
  410 CONTINUE
      IF(N.EQ.1) RES1=RESID
      IF(N.EQ.1.AND.RESID.EQ.0.)GO TO 430
      IF(RESID/RES1.LT.1.E-5) GO TO 430
  420 CONTINUE
  430 WRITE(6,2009) N
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
      CALL POINTS
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
 1000 FORMAT (5I5)
 1001 FORMAT (3(2I3,2F8.3),I5)
```

```
2000 FORMAT (30H1  MESH GENERATION INFORMATION//
    1 41H0  MAXIMUM VALUE OF I IN THE MESH--------I3/
    2 41H0  MAXIMUM VALUE OF J IN THE MESH--------I3/
    3 41H0  NUMBER OF LINE SEGMENT CARDS----------I3/
    4 41H0  NUMBER OF BOUNDARY CONDITION CARDS----I3/
    5 41H0  NUMBER OF MATERIAL BLOCK CARDS--------I3///)
2001 FORMAT (//88H  INPUT   I1  J1    R1      Z1    I2  J2    R2      Z
    12    I3  J3    R3      Z3    IPTION/8X,3(2I4,2F8.4),I6)
2002 FORMAT (5H  DI=F4.0,5H  DJ=F4.0,7H  DIFF=F4.0,7H  RINC=F8.3,7H  ZI
    1NC=F8.3,7H  ITER=I3,7H  IINC=I3,7H  JINC=I3,8H  KAPPA=I1)
2003 FORMAT(1X,38H**BAD INPUT--THIS LINE IS NOT DIAGONAL)
2004 FORMAT (30H    I    J        AR        AZ)
2005 FORMAT (2I5,2F11.6)
2006 FORMAT (51H ** BAD INPUT - THESE POINTS DO NOT DEFINE A CIRCLE,/,
    13X,6F12.4,10X,2E20.8)
2007 FORMAT(19H  CENTER COORDINATE,(F11.6,1X,F11.6,1X))
2008 FORMAT (7H  ANG1=F9.6,7H  ANG2=F9.6,7H  DIFF=F3.0,9H  DELPHI=F9.6)
2009 FORMAT (//30H  COORDINATES CALCULATED AFTER I3,11H ITERATIONS)
     RETURN
     END
     SUBROUTINE MNIMX(I,J)
     COMMON/TD/IMIN(100),IMAX(100),JMIN(25),JMAX(25),MAXI,MAXJ,NMTL,NBC
     IF(J.LT.JMIN(I)) JMIN(I)=J
     IF(J.GT.JMAX(I)) JMAX(I)=J
     IF(I.LT.IMIN(J)) IMIN(J)=I
     IF(I.GT.IMAX(J)) IMAX(J)=I
     RETURN
     END
     SUBROUTINE ADINA(NFACE,FTHETA)
     INTEGER CODE
     COMMON/NPDATA/R(1000),CODE(1000),XR(1000),Z(1000),XZ(1000),
    1NPNUM(25,80),T(1000),XT(1000)
     COMMON/ARG/RRR(5),ZZZ(5),RR(4),ZZ(4),S(15,15),P(15),TT(6),
    1H(6,15),CRZ(6,6),XI(10),ANGLE(4),SIG(18),EPS(18),N
     COMMON/ELDATA/BETA(1000),EPR(1000),PR(200),SH(200),IX(1000,5),
    1IP(200),JP(200),IS(200),JS(200),ALPHA(1000),IT(200),JT(200),
    2ST(200),INP(1000),JNP(1000)
     COMMON/BASIC/ACELZ,ANGVEL,ANGACC,TREF,VOL,NUMNP,NUMEL,NUMPC,NUMSC,
    1NUMST
     COMMON/NXMESH/THETAN(4),NST(4),NUMS(4,5),NPC(8,8)
     DIMENSION NPG(25,80,9),FTHETA(9),NCOD(8),NEL(8),ID(6)
     COMMON/ANS1/NUMELS(4),NUMNPS(4)
     COMMON/NXDATA/NTP,NTYPS,NTS,NTOTS
     COMMON/NONAXI/S1(30,30),P1(30),THETA,BS1(6,30)
     COMMON/SOLVE/X(4428),Y(4428),TEM(4428),NUMTC,MBAND
     COMMON/TD/IMIN(100),IMAX(100),JMIN(25),JMAX(25),MAXI,MAXJ,NMTL,NBC
     COMMON/CONVRG/IDONE
     COMMON/PLANE/NPP
     COMMON/RESULT/BS(6,15),D(6,6),C(6,6),AR,BB(6,9),CNS(6,6)
     COMMON/MATP/RO(6),E(12,16,6),EE(16),AOFTS(6)
     COMMON/TOTAL/NELN
     DATA NPG/18000*0/
```

```
      NI=NST(NTP)
C LOOP ON # OF SEGMENTS OF TYPE NTP
      DO 500 I=1,NI
      DO 200 J=1,NUMNP
      NFL=NUMS(NTP,I)
      NFR=NFL+1
      IP1=INP(J)
      JP1=JNP(J)
      NF=NFL
      DO 200 K=1,2
      NC=CODE(J)+1
      IF(NPG(IP1,JP1,NF).NE.0)GO TO 200
      NPG(IP1,JP1,NF)=IP1+(JP1-1)*25+NF*200
      DO 10 II=1,8
 10   ID(II)=0
      IF(NC.EQ.3.OR.NC.EQ.4.OR.NC.EQ.7.OR.NC.EQ.8) ID(1)=1
      IF(NF.EQ.1) ID(3)=1
      IF(NF.EQ.NTOTS+1) ID(2)=1
      IF(R(J).EQ.0.0) ID(2)=1
      IF(R(J).EQ.0.0) ID(3)=1
C
C     WRITE NODAL DATA ON FILE 2
C
      WRITE(2,100)NPG(IP1,JP1,NF),(ID(KK),KK=1,6),Z(J),R(J),FTHETA(NF)
  100 FORMAT("X",I4,1X,I4,5I5,3F10.4,4X,"0")
  200 NF=NFR
      DO 350 J=1,NUMEL
      NF=NUMS(NTP,I)
      I1=IX(J,1)
      I2=IX(J,2)
      I3=IX(J,3)
      I4=IX(J,4)
      IP1=INP(I1)
      JP1=JNP(I1)
      NEL(1)=NPG(IP1,JP1,NFL)
      IF(IP1.EQ.1)NEL(1)=NPG(IP1,JP1,1)
      NEL(4)=NPG(IP1,JP1,NFR)
      IP1=INP(I2)
      JP1=JNP(I2)
      NEL(2)=NPG(IP1,JP1,NFL)
      NEL(3)=NPG(IP1,JP1,NFR)
      IP1=INP(I4)
      JP1=JNP(I4)
      NEL(5)=NPG(IP1,JP1,NFL)
      IF(IP1.EQ.1)NEL(5)=NPG(IP1,JP1,1)
      NEL(8)=NPG(IP1,JP1,NFR)
      IP1=INP(I3)
      JP1=JNP(I3)
      NEL(6)=NPG(IP1,JP1,NFL)
      NEL(7)=NPG(IP1,JP1,NFR)
      NELN=NELN+1
      NZ=0
```

```
      N8=8
      WRITE(3,300)NELN,N8,N8,NZ,IX(J,5),NELN,NZ,NZ
  300 FORMAT(8I5,F10.0)
      WRITE(3,301) (NEL(K),K=1,8)
  301 FORMAT(8I5)
C
C     WRITE MATERIAL AXES DATA ON FILE 4
C
      NPI=NEL(1)+7000
      NPJ=NEL(1)+8000
      NPK=NEL(1)+9000
      WRITE(4,302) NELN,NPI,NPJ,NPK
  302 FORMAT(4I5)
C
C     WRITE NODAL POINTS FOR MATERIAL AXES
C
      FTRAD=FTHETA(NF)/57.29578
      CA=COS(ALPHA(J))
      CB=COS(BETA(J))
      SA=SIN(ALPHA(J))
      SB=SIN(BETA(J))
      CF=COS(FTRAD)
      SF=SIN(FTRAD)
      XPI=0.0
      YPI=0.0
      ZPI=0.0
      XPJ=CA*CB
      YPJ=CA*SB*CF+SA*SF
      ZPJ=CA*SB*SF-SA*CF
      XPK=SA*CB
      YPK=SA*SB*CF-CA*SF
      ZPK=SA*SB*SF+CA*CF
      WRITE(7,303)NPI,NZ,NZ,NZ,NZ,NZ,NZ,XPI,YPI,ZPI
      WRITE(7,303)NPJ,NZ,NZ,NZ,NZ,NZ,NZ,XPJ,YPJ,ZPJ
      WRITE(7,303)NPK,NZ,NZ,NZ,NZ,NZ,NZ,XPK,YPK,ZPK
  303 FORMAT(1X,I4,1X,I4,5I5,3F10.4,4X,"0")
  350 CONTINUE
      DO 500 L=1,NUMPC
      IP1=INP(IP(L))
      JP1=JNP(IP(L))
      IP2=INP(JP(L))
      JP2=JNP(JP(L))
      NEL(1)=NPG(IP1,JP1,NFL)
      NEL(2)=NPG(IP2,JP2,NFL)
      NEL(3)=NPG(IP2,JP2,NFR)
      NEL(4)=NPG(IP1,JP1,NFR)
      NCUR=1
      WRITE(1,400)NCUR,(NEL(K),K=1,4)
      WRITE(1,401) PR(L),PR(L),PR(L),PR(L)
  400 FORMAT(9I5)
  401 FORMAT(5F10.0,I5)
  500 CONTINUE
```

```
      END
      FUNCTION NODE(I,J)
      COMMON/TD/IMIN(100),IMAX(100),JMIN(25),JMAX(25),MAXI,MAXJ,NMTL,NBC
      NODE=0
      DO 100 JJ=1,J
      NSTART=IMIN(JJ)
      NSTOP=IMAX(JJ)
      DO 100 II=NSTART,NSTOP
      NODE=NODE+1
      IF(JJ.EQ.J.AND.II.EQ.I) RETURN
  100 CONTINUE
      RETURN
      END
      SUBROUTINE POINTS
      INTEGER CODE
      COMMON/BASIC/ACELZ,ANGVEL,ANGACC,TREF,VOL,NUMNP,NUMEL,NUMPC,NUMSC,
     1NUMST
      COMMON/MATP/RO(6),E(12,16,6),EE(16),AOFTS(6)
      COMMON/NPDATA/R(1000),CODE(1000),XR(1000),Z(1000),XZ(1000),
     1NPNUM(25,80),T(1000),XT(1000)
      COMMON/ELDATA/BETA(1000),EPR(1000),PR(200),SH(200),IX(1000,5),
     1IP(200),JP(200),IS(200),JS(200),ALPHA(1000),IT(200),JT(200),
     2ST(200),INP(1000),JNP(1000)
      COMMON/SOLVE/X(4428),Y(4428),TEM(4428),NUMTC,MBAND
      COMMON/TD/IMIN(100),IMAX(100),JMIN(25),JMAX(25),MAXI,MAXJ,NMTL,NBC
      COMMON/PLANE/NPP
      DIMENSION AR(25,80),AZ(25,80),MATRIL(100,5),BLKANG(100),BLKALF(1
     100)
      DIMENSION IBNG(100),NBNG(100)
      EQUIVALENCE (R(1),AR),(Z(1),AZ)
C     ESTABLISH NODAL POINT INFORMATION
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
      NEL=0
      NODSUM=0
      DO 100 J=1,MAXJ
      NSTART=IMIN(J)
      NSTOP=IMAX(J)
      DO 100 I=NSTART,NSTOP
  100 NODSUM=NODSUM+1
      NELSUM=0
      JJMAX=MAXJ-1
      DO 110 JJ=1,JJMAX
      NSTOP=MIN0(IMAX(JJ),IMAX(JJ+1))-1
      NSTART=MAX0(IMIN(JJ),IMIN(JJ+1))
      DO 110 II=NSTART,NSTOP
  110 NELSUM=NELSUM+1
      NUMNP=NODSUM
      NUMEL=NELSUM
      DO 120 J=1,MAXJ
      NSTART=IMIN(J)
      NSTOP=IMAX(J)
      DO 120 I=NSTART,NSTOP
```

```
      NPNUM(I,J)=NODE(I,J)
      NP=NPNUM(I,J)
      INP(NP)=I
      JNP(NP)=J
      R(NP)=AR(I,J)
  120 Z(NP)=AZ(I,J)
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C     READ AND ASSIGN BOUNDARY CONDITIONS
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C     INITIALIZE
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
      DO 130 I=1,NUMNP
      CODE(I)=0
      IF(R(I).EQ.0..AND.NPP.EQ.0) CODE(I)=1.
      XR(I)=0.
      XZ(I)=0.
      XT(I)=0.0
  130 T(I)=0.
      IF(NBC.EQ.0) GO TO 210
      DO 200 IBCON=1,NBC
      READ(5,1002) I1,I2,J1,J2,ICN,RCON,ZCON,TCON
      DO 200 I=I1,I2
      DO 200 J=J1,J2
      NP=NPNUM(I,J)
      CODE(NP)=ICN
      XR(NP)=RCON
      XT(NP)=TCON
  200 XZ(NP)=ZCON
  210 MPRINT=0
      DO 230 J=1,MAXJ
      NSTART=IMIN(J)
      NSTOP=IMAX(J)
      DO 230 I=NSTART,NSTOP
      NP=NPNUM(I,J)
      IF(MPRINT.NE.0) GO TO 220
      WRITE(6,2000)
      MPRINT=59
  220 MPRINT=MPRINT-1
  230 WRITE(6,2001) I,J,NP,CODE(NP),R(NP),Z(NP),XR(NP),XZ(NP),XT(NP)
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C     ASSIGN MATERIALS IN BLOCKS
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
      DO 300 M1=1,NUMEL
  300 IX(M1,5)=0
      DO 310 IMTL=1,NMTL
      READ (5,1000) MTL,(MATRIL(IMTL,IM),IM=2,5),BLKANG(IMTL),BLKALF(IMT
     1L),IBNG(IMTL),NBNG(IMTL)
  310 MATRIL(IMTL,1)=MTL
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C     ESTABLISH ELEMENT INFORMATION
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
      JJMAX=MAXJ-1
```

```
      N=0
      MTL=1
      KTL=1
      DO 440 JJ=1,JJMAX
      NSTOP=MIN0(IMAX(JJ),IMAX(JJ+1))-1
      NSTART=MAX0(IMIN(JJ),IMIN(JJ+1))
      DO 440 II=NSTART,NSTOP
      NEL=NEL+1
      DO 400 IMTL=1,NMTL
      IF(II.LT.MATRIL(IMTL,2)) GO TO 400
      IF(II.GE.MATRIL(IMTL,3)) GO TO 400
      IF(JJ.LT.MATRIL(IMTL,4)) GO TO 400
      IF(JJ.GE.MATRIL(IMTL,5)) GO TO 400
      KAT=IMTL
      MAT=MATRIL(IMTL,1)
  400 CONTINUE
      IF(KAT.EQ.KTL) GO TO 410
      KTL=KAT
      MTL=MAT
      GO TO 420
  410 IF(II.EQ.NSTART) GO TO 420
      IF(JJ.NE.JJMAX.OR.II.NE.NSTOP) GO TO 440
      M=NEL+1
      IANG=ICNG
      NANG=NCNG
      GO TO 421
  420 I=NPNUM(II,JJ)
      J=I+1
      K=NPNUM(II+1,JJ+1)
      L=K-1
      M=NEL
      IX(M,1)=I
      IX(M,2)=J
      IX(M,3)=K
      IX(M,4)=L
      IX(M,5)=MTL
      BETA(M)=BLKANG(KTL)
      ALPHA(M)=BLKALF(KTL)
      IANG=ICNG
      NANG=NCNG
      ICNG=IBNG(KTL)
      NCNG=NBNG(KTL)
  421 NC=2
  430 N=N+1
      IF(M.LE.N) GO TO 440
      IX(N,1)=IX(N-1,1)+1
      IX(N,2)=IX(N-1,2)+1
      IX(N,3)=IX(N-1,3)+1
      IX(N,4)=IX(N-1,4)+1
      IX(N,5)=IX(N-1,5)
      BETA(N)=BETA(N-1)
      IF(IANG.EQ.1) GO TO 442
```

```
      ALPHA(N)=ALPHA(N-1)
      GO TO 443
  442 CONTINUE
      IF(NC.GT.NANG) GO TO 444
      ALPHA(N)=ALPHA(N-1)
      GO TO 443
  444 NC=1
      ALPHA(N)=-ALPHA(N-1)
  443 CONTINUE
      NC=NC+1
      IF(M.GT.N) GO TO 430
  440 CONTINUE
      IF(NUMNP.GT.2000) WRITE(6,2002)
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C     SET NODAL POINT TEMPERATURE TO REFERENCE TEMPERATURE
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
      IF(NUMTC.NE.0) RETURN
      DO 500 N=1,NUMNP
  500 T(N)=TREF
 1000 FORMAT (5I5,2F10.0,2I5)
 1002 FORMAT(4I5,I10,3F10.0)
 2000 FORMAT (128H1   I    J    NP         TYPE    R-ORDINATE    Z-ORDINA
     1TE  R LOAD OR DISPLACEMENT  Z LOAD OR DISPLACEMENT  T LOAD OR DISP
     2LACEMENT)
 2001 FORMAT (2I5,I6,I12,F13.6,F14.6,E26.7,E24.7,E24.7)
 2002 FORMAT (35H  BAD INPUT - TOO MANY NODAL POINTS)
      RETURN
      END
      SUBROUTINE TEMP(R,Z,T)
      COMMON/SOLVE/X(4428),Y(4428),TEM(4428),NUMTC,MBAND
      DIMENSION SMALL(20),ISM(20)
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C     INITIALIZE
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
      J=1
      JMAX=16
      IF(NUMTC.LT.JMAX) JMAX=NUMTC
      DO 10 I=1,JMAX
      SMALL(I)=0.
   10 ISM(I)=0
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C     FIND THE JMAX CLOSEST POINTS
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
      DO 50 I=1,NUMTC
      DSQ=(X(I)-R)**2+(Y(I)-Z)**2
      IF(DSQ.GT..1E-4) GO TO 20
      T=TEM(I)
      RETURN
   20 IF(I.EQ.1) SMALL(1)=DSQ
      IF(I.EQ.1) ISM(1)=1
      IF(I.EQ.1) GO TO 50
      IF(SMALL(J).LE.DSQ.AND.J.LT.JMAX) SMALL(J+1)=DSQ
```

```
      IF(SMALL(J).LE.DSQ.AND.J.LT.JMAX) ISM(J+1)=I
      IF(SMALL(J).LE.DSQ) GO TO 40
      DO 30 K=1,J
      JB=J-K +1
      IF(JB.EQ.0) GO TO 40
      SMALL(JB+1)=SMALL(JB)
      ISM(JB+1)=ISM(JB)
      SMALL(JB)=DSQ
      ISM(JB)=I
      IF(JB.EQ.1) GO TO 40
      IF(SMALL(JB-1).LE.DSQ) GO TO 40
   30 CONTINUE
   40 IF(J.LT.JMAX) J=J+1
   50 CONTINUE
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C     FIND THE THIRD TEMPERATURE POINT BY AREA TEST
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
      JCHK=JMAX-2
      J=0
      I1=ISM(1)
      I2=ISM(2)
   60 I3=ISM(J+3)
      AREA=.50*(Y(I1)*X(I3)-Y(I3)*X(I1)+Y(I3)*X(I2)-Y(I2)*X(I3)+
     1       Y(I2)*X(I1)-Y(I1)*X(I2))
      D1=(X(I2)-X(I1))**2+(Y(I2)-Y(I1))**2
C     IF D1 IS APPROXIMATELY 0. IT IS ASSUMED THAT THERE EXISTS A
C     DUPLICATION OF INPUT
      IF(D1.GT..1E-3) GO TO 70
      I2=I3
      J=J+1
      GO TO 60
   70 IF(AREA**2.GT..1*D1*SMALL(1)) GO TO 80
      J=J+1
      IF(J.LT.JCHK) GO TO 60
      WRITE(6,2000) I1,I2,I3,J
      T=TEM(I1)
      RETURN
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C     FIND TEMPERATURE INTERCEPT
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
   80 DETA=Y(I1)*(TEM(I3)-TEM(I2))+Y(I2)*(TEM(I1)-TEM(I3))
     1      +Y(I3)*(TEM(I2)-TEM(I1))
      DETB=X(I1)*(TEM(I2)-TEM(I3))+X(I2)*(TEM(I3)-TEM(I1))
     1      +X(I3)*(TEM(I1)-TEM(I2))
      DETC=TEM(I1)*(X(I2)*Y(I3)-X(I3)*Y(I2))+TEM(I2)*(X(I3)*Y(I1)-X(I1)*
     1Y(I3))+TEM(I3)*(X(II)*Y(I2)-X(I2)*Y(II))
      T=(DETA*R+DETB*Z+DETC)/(2.*AREA)
 2000 FORMAT (28H  ERROR IN TEMPERATURE INPUT,5H  I1=I4,5H  I2=I4,
     15H  I3=I4,4H  J=I4)
      RETURN
      END
      SUBROUTINE TEM2(NUMNP)
```

```
      INTEGER CODE
      COMMON/NPDATA/R(1000),CODE(1000),XR(1000),Z(1000),XZ(1000),
     1NPNUM(25,80),T(1000),XT(1000)
      READ(5,1000) TCONST
      DO 100 N=1,NUMNP
  100 T(N)=TCONST
 1000 FORMAT(F10.0)
      RETURN
      END
      END
N
```

APPENDIX B

Listing of the NASGEN Program

```
      PROGRAM NASGEN(INPUT,OUTPUT,NAS,TAPE5=INPUT,TAPE6=OUTPUT,TAPE1=
     1 NAS)
      INTEGER CODE
      COMMON/NPDATA/R(1000),CODE(1000),XR(1000),Z(1000),XZ(1000),
     1NPNUM(25,80),T(1000),XT(1000)
      COMMON/ARG/RRR(5),ZZZ(5),RR(4),ZZ(4),S(15,15),P(15),TT(6),
     1H(6,15),CRZ(6,6),XI(10),ANGLE(4),SIG(18),EPS(18),N
      COMMON/ELDATA/BETA(1000),EPR(1000),PR(200),SH(200),IX(1000,5),
     1IP(200),JP(200),IS(200),JS(200),ALPHA(1000),IT(200),JT(200),
     2ST(200),INP(1000),JNP(1000)
      COMMON/BASIC/ACELZ,ANGVEL,ANGACC,TREF,VOL,NUMNP,NUMEL,NUMPC,NUMSC,
     1NUMST
      COMMON/NXMESH/THETAN(4),NST(4),NUMS(4,5),NPC(8,8)
      COMMON/ANS1/NUMELS(4),NUMNPS(4)
      COMMON/NXDATA/NTP,NTYPS,NTS,NTOTS
      COMMON/NONAXI/S1(30,30),P1(30),THETA,BS1(6,30)
      COMMON/SOLVE/X(4428),Y(4428),TEM(4428),NUMTC,MBAND
      COMMON/TD/IMIN(100),IMAX(100),JMIN(25),JMAX(25),MAXI,MAXJ,NMTL,NBC
      COMMON/CONVRG/IDONE
      COMMON/PLANE/NPP
      COMMON/RESULT/BS(6,15),D(6,6),C(6,6),AR,BB(6,9),CNS(6,6)
      COMMON/MATP/RO(6),E(12,16,6),EE(16),AOFTS(6)
      DIMENSION TITLE(20),FTHETA(8)
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C     READ AND WRITE CONTROL INFORMATION
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
      READ(5,3000) NTYPS,NTOTS
      DO 150 I=1,NTYPS
  150 READ(5,3001) THETAN(I),NST(I)
       DO 151 I=1,NTYPS
      J2 =NST(I)
  151 READ(5,3000) (NUMS(I,J),J=1,J2)
      DO 152  I= 1,NTOTS
  152 READ(5,3002) (NPC(I,J),J=1,8)
 3000 FORMAT(8I10)
 3001 FORMAT(F10.5,I10)
 3002 FORMAT(8I10)
      NFACE=NTOTS+1
      FTHETA(1)=0.0
      DO 4100 I=1,NTOTS
      DO 4000 J=1,NTYPS
      K2=NST(J)
      DO 4000 K=1,K2
 4000 IF(NUMS(J,K).EQ.I)GO TO 4100
 4100 FTHETA(I+1)=FTHETA(I)+THETAN(J)
      WRITE(6,4200)((FTHETA(I),I),I=1,NTOTS)
 4200 FORMAT(" ANGLE IN DEGREES ",5X,"  BEGINNING SEGMENT #"/
     1 (E20.10,I10))
       WRITE(6,3010)
 3010 FORMAT("1","SEGMENT DATA FOR NONAXISYMMETRIC PROBLEM")
       WRITE(6,3011) NTYPS,NTOTS
 3011 FORMAT (" "," NUMBER OF TYPES OF SEGMENTS = ",I5,//,
```

```
     1            " NUMBER OF TOTAL SEGMENTS     =",I5)
       DO 153 I=1,NTYPS
      WRITE(6,3012) I,THETAN(I),NST(I)
 3012 FORMAT(" ",///," SEGMENT TYPE = ",I5/,"      THETA = ",F10.5/,
     1  "  NUMBER OF SEGMENTS OF THIS TYPE = ",I5)
       J2 = NST(I)
       WRITE(6,3013)(NUMS(I,J),J=1,J2)
 3013 FORMAT(" "," SEGMENT NUMBERS IN GLOBAL SYSTEM ARE ",5I5)
  153 CONTINUE
       DO 154  I=1,NTOTS
  154  WRITE(6,3014)I,(NPC(I,J),J=1,8)
 3014 FORMAT("  ","CONNECTING NODES FOR SEGMENT",I5," ARE",8I5)
      DO 950 NTP = 1,NTYPS
       THETA= THETAN(NTP)       /57.295780
   50 READ(5,1000 )TITLE,NNLA,NUMTC,NUMMAT,NUMPC,NUMSC,NUMST,TREF
     1,INERT,NLINC,INCI,INCF,IPLOT
      WRITE(6,2000)TITLE,NNLA,NUMTC,NUMMAT,NUMPC,NUMSC,NUMST,TREF,INERT,
     1NLINC
      NPP=0
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C     GENERATE FINITE ELEMENT MESH
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
  100 CALL MESH
       NUMELS(NTP) = NUMEL
       NUMNPS(NTP) = NUMNP
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C     READ AND WRITE T5MPERATURE DATA
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
  103 IF(NUMTC.EQ.0) GO TO 440
      IF(NUMTC.GT.0) READ(5,1001) (X(I),Y(I),TEM(I),I=1,NUMTC)
      IF(NUMTC.EQ.-2) CALL TEM2(NUMNP)
      IF(NUMTC.EQ.-2) GO TO 440
      MPRINT=0
      DO 210 I=1,NUMTC
      IF(MPRINT.NE.0) GO TO 200
      WRITE(6,2001)
      MPRINT=59
  200 MPRINT=MPRINT-1
  210 WRITE(6,2002) X(I),Y(I),TEM(I)
      MPRINT=0
      DO 230 N=1,NUMNP
      IF(MPRINT.NE.0) GO TO 220
      WRITE(6,2003)
      MPRINT=59
  220 MPRINT=MPRINT-1
      CALL TEMP(R(N),Z(N),T(N))
  230 WRITE(6,2004) N,R(N),Z(N),T(N)
  440 MPRINT=0
      DO 460 N=1,NUMEL
      IF(MPRINT.NE.0) GO TO 450
      WRITE(6,2008)
      MPRINT=59
```

```
  450 MPRINT=MPRINT-1
      II=IX(N,1)
      JJ=IX(N,2)
      KK=IX(N,3)
      LL=IX(N,4)
C
C     TEM IS TEMPORARY STORAGE FOR ELEMENT TEMPERATURES
C
      TEM(N)=(T(II)+T(JJ)+T(KK)+T(LL))/4.00
  460 WRITE(6,2009) N,(IX(N,I),I=1,5),BETA(N),ALPHA(N),TEM(N)
      DO 470 K=1,NUMEL
  470 T(K)=TEM(K)
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C     READ AND WRITE MATERIAL PROPERTIES
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
  500 CONTINUE
      DO 510 M=1,NUMMAT
      READ(5,1004)  MTYPE,(NT,RO(MTYPE),AOFTS(MTYPE))
      WRITE(6,2010) MTYPE,NT,RO(MTYPE)
      READ(5,1005)((E(I,J,MTYPE),J=1,14),I=1,NT)
      IF(AOFTS(MTYPE).NE.1.) WRITE(6,2011)((E(I,J,MTYPE),J=1,13),I=1,NT)
      IF(AOFTS(MTYPE).EQ.1.) WRITE(6,2012)((E(I,J,MTYPE),J=1,13),I=1,NT)
      DO 510 I=NT,12
      DO 510 J=1,16
  510 E(I,J,MTYPE)=E(NT,J,MTYPE)
      DO 900 NL=1,NLINC
      WRITE(6,2030) NL
      ACELZ=0.00
      ANGVEL=0.00
      ANGACC=0.00
      IF(INERT .EQ. 0) GO TO 511
      IF(NL .NE. 1 .AND. INCI .EQ. 0) GO TO 511
C*********************************************************************
C     READ AND WRITE DYNAMIC FORCES
C*********************************************************************
      READ(5,1030) ACELZ, ANGVEL, ANGACC
      WRITE(6,2031) ACELZ, ANGVEL, ANGACC
  511 CONTINUE
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C     READ AND WRITE PRESSURE AND SHEAR BOUNDARY CONDITIONS
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
      IF(NL .NE. 1 .AND. INCF .EQ. 0) GO TO 700
  600 IF(NUMPC.EQ.0) GO TO 630
      MPRINT=0
      DO 620 L=1,NUMPC
      IF(MPRINT.NE.0) GO TO 610
      WRITE(6,2013)
      MPRINT=58
  610 MPRINT=MPRINT-1
      READ(5,1006)  IP(L),JP(L),PR(L)
  620 WRITE(6,2014) IP(L),JP(L),PR(L)
  630 IF(NUMSC.EQ.0) GO TO 701
```

```
      MPRINT=0
      DO 650 L=1,NUMSC
      IF(MPRINT.NE.0) GO TO 640
      WRITE(6,2015)
      MPRINT=58
  640 MPRINT=MPRINT-1
      READ(5,1006)  IS(L),JS(L),SH(L)
  650 WRITE(6,2014) IS(L),JS(L),SH(L)
  701 IF(NUMST.EQ.0) GO TO 700
      MPRINT=0
      DO 680 L=1,NUMST
      IF(MPRINT.NE.0) GO TO 670
      WRITE(6,2025)
      MPRINT=58
  670 MPRINT=MPRINT-1
      READ(5,1006) IT(L),JT(L),ST(L)
  680 WRITE(6,2014)IT(L),JT(L),ST(L)
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C     DETERMINE BANDWIDTH, INITIALIZE ELASTIC-PLASTIC RATIO,
C     AND CONVERT BETA FROM DEGREES TO RADIANS
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
  700 J=0
      DO 710 N=1,NUMEL
      IX(N,5)=IABS(IX(N,5))
      DO 710 I=1,4
      DO 710 L=1,4
      KK=IABS(IX(N,I)-IX(N,L))
      IF(KK.GE.J) J=KK
  710 CONTINUE
      MBAND=3*J+3
      IF(NL.GT.1) GO TO 721
      DO 720 N=1,NUMEL
      EPR(N)=1.
      ALPHA(N)=ALPHA(N)/57.295780
  720 BETA(N)=BETA(N)/57.295780
  721 CONTINUE
  900 CONTINUE
      CALL NASTRN(NFACE,FTHETA)
  950 CONTINUE
      WRITE(1,4500)((MTYPE,E(1,4,MTYPE),E(1,10,MTYPE),E(1,7,MTYPE)
     1 ,RO(MTYPE)),MTYPE=1,NUMMAT)
 4500 FORMAT("MAT1",4X,I8,4E8.2)
      WRITE(1,4600)ACELZ
 4600 FORMAT("CORD1C",9X,"1",5X,"101",5X,"102",5X,"103"/"GRID",9X,
     1 "101",13X,"0.0",5X,"0.0",5X,"0.0",10X,"123456"/"GRID",9X,
     2 "102",13X,"0.0",5X,"0.0",5X,"1.0",10X,"123456"/"GRID",9X,
     3 "103",13X,"1.0",5X,"0.0",5X,"1.0",10X,"123456"/"GRAV",11X
     4 ,"2",7X,"1",E8.2,5X,"0.0",5X,"0.0",4X,"-1.0"/"PSOLID",
     5 9X,"1",7X,"1",7X,"1"/"ENDDATA")
  910 CONTINUE
 1000 FORMAT(20A4/6I5,F5.0,5I5)
 1001 FORMAT(3F10.0)
```

```
1004 FORMAT (2I5,2F10.0)
1005 FORMAT(7F10.0)
1006 FORMAT (2I5,F10.0)
1030 FORMAT(3F10.0)
2000 FORMAT (2H1 ,20A4/
    1 33H0  NUMBER OF APPROXIMATIONS------I4/
    2 33H0  NUMBER OF TEMPERATURE CARDS---I4/
    3 33H0  NUMBER OF MATERIALS----------I4/
    4 33H0  NUMBER OF PRESSURE CARDS------I4/
    5 33H0  NUMBER OF SHEAR CARDS---------I4/
    6 33H0  NUMBER OF TORSION CARDS-------I4/
    7 33H0  REFERENCE TEMPERATURE---------E12.4/
    8 33H0  NUMBER OF INERTIA CARDS-------I4/
    9 33H0  NUMBER OF LOAD INCREMENTS-----I4/)
2001 FORMAT (1H1,13X,1HR,14X,1HZ,14X,1HT)
2002 FORMAT (3F15.3)
2003 FORMAT (35H1   N      R          Z            T)
2004 FORMAT (I5,2F10.4,F10.0)
2008 FORMAT (74H1  EL   I   J   K  L    MATERIAL   ANGLE BETA   ANGLE A
    1LPHA    TEMPERATURE)
2009 FORMAT (I5,4I4,I8,F11.1,2F13.3)
2010 FORMAT (1H1,"MATERIAL IDENTIFICATION NUMBER =",I2/
    11H ,"NO. OF MATERIAL TEMPERATURE CARDS =",I2/
    21H ,"MASS DENSITY =",E15.7)
2011 FORMAT (1H ,"TEMPERATURE =",E15.7/
    11H ,"MODULUS OF ELASTICITY-EN =",E15.7/
    21H ,"MODULUS OF ELASTICITY-ES =",E15.7/
    31H ,"MODULUS OF ELASTICITY-ET =",E15.7/
    41H ,"POISSON RATIO-NUNS =",E15.7/
    51H ,"POISSON RATIO-NUNT =",E15.7/
    61H ,"POISSON RATIO-NUST =",E15.7/
    71H ,"SHEAR MODULUS-GNS =",E15.7/
    81H ,"SHEAR MODULUS-GST =",E15.7/
    91H ,"SHEAR MODULUS-GTN =",E15.7/
    11H ,"COEFFICIENT OF THERMAL EXPANSION-AN =",E15.7/
    21H ,"COEFFICIENT OF THERMAL EXPANSION-AS =",E15.7/
    31H ,"COEFFICIENT OF THERMAL EXPANSION-AT =",E15.7/)
2012 FORMAT (1H ,"TEMPERATURE =",E15.7/
    11H ,"MODULUS OF ELASTICITY-EN =",E15.7/
    21H ,"MODULUS OF ELASTICITY-ES =",E15.7/
    31H ,"MODULUS OF ELASTICITY-ET =",E15.7/
    41H ,"POISSON RATIO-NUNS =",E15.7/
    51H ,"POISSON RATIO-NUNT =",E15.7/
    61H ,"POISSON RATIO-NUST =",E15.7/
    71H ,"SHEAR MODULUS-GNS =",E15.7/
    81H ,"SHEAR MODULUS-GST =",E15.7/
    91H ,"SHEAR MODULUS-GTN =",E15.7/
    11H ,"FREE THERMAL STRAIN-FN =",E15.7/
    21H ,"FREE THERMAL STRAIN-FS =",E15.7/
    31H ,"FREE THERMAL STRAIN-FT =",E15.7/)
2013 FORMAT (30H1 PRESSURE BOUNDARY CONDITIONS/20H    I    J  PRESSURE)
2014 FORMAT (2I5,F10.1)
```

```
 2015 FORMAT (27H1 SHEAR BOUNDARY CONDITIONS/17H    I    J  SHEAR)
 2016 FORMAT (26H  THE SYSTEM CONVERGED IN I2,11H ITERATIONS)
 2017 FORMAT (33H  THE SYSTEM DID NOT CONVERGE IN I2,11H ITERATIONS)
 2024 FORMAT (43H0 THE AXISYMMETRIC OPTION NAS BEEN SELECTED)
 2025 FORMAT(30H1  TORSION BOUNDARY CONDITIONS/17H    I    J  SHEAR)
 2030 FORMAT(1H1,"LOAD STEP=",I4)
 2031 FORMAT(1H0 ,"AXIAL ACCELERATION  =",E12.4/
     11H0 ,"ANGULAR VELOCITY    =",E12.4/
     21H0 ,"ANGULAR ACCELERATION=",E12.4)
  920 STOP
      END
      SUBROUTINE ANGLE (R,Z,RC,ZC,ANG)
C     FIND ANGLE OF INCLINATION BETWEEN 0 AND 2*PI
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
      PI=3.1415927
      D1=(Z-ZC)
      D2=(R-RC)
      IF(ABS(R-RC).GT.1.E-8) GO TO 100
      ANG=PI/2.
      IF(D1.GT.1.E-8) RETURN
      ANG=-ANG
      RETURN
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C     ALLOW CIRCLE TO CROSS AXIS
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
  100 ANG=ATAN2(D1,D2)
      RETURN
      END
      SUBROUTINE CIRCLE(ANG1,DELPHI,RSTRT,ZSTRT,RC,ZC,I,J)
      INTEGER CODE
      COMMON/TD/IMIN(100),IMAX(100),JMIN(25),JMAX(25),MAXI,MAXJ,NMTL,NBC
      COMMON/NPDATA/R(1000),CODE(1000),XR(1000),Z(1000),XZ(1000),
     1NPNUM(25,80),T(1000),XT(1000)
      DIMENSION AR(25,80),AZ(25,80)
      EQUIVALENCE (R(1),AR),(Z(1),AZ)
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C     FIND INTERSECTION OF LINE AND CIRCLE = NEW R AND Z
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
      ANG1=ANG1+DELPHI
      RR=SQRT((RSTRT-RC)**2+(ZSTRT-ZC)**2)
      AR(I,J)=RC+RR*COS(ANG1)
      AZ(I,J)=ZC+RR*SIN(ANG1)
      RETURN
      END
      SUBROUTINE MESH
      INTEGER CODE
      DIMENSION AR(25,80),AZ(25,80),NCODE(25,80)
      COMMON/TD/IMIN(100),IMAX(100),JMIN(25),JMAX(25),MAXI,MAXJ,NMTL,NBC
      COMMON/NPDATA/R(1000),CODE(1000),XR(1000),Z(1000),XZ(1000),
     1NPNUM(25,80),T(1000),XT(1000)
      COMMON/ELDATA/BETA(1000),EPR(1000),PR(200),SH(200),IX(1000,5),
     1IP(200),JP(200),IS(200),JS(200),ALPHA(1000),IT(200),JT(200),
```

```
     2ST(200),INP(1000),JNP(1000)
      EQUIVALENCE (R(1),AR),(Z(1),AZ),(IX(1,1),NCODE)
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C     MESH CONTROL INFORMATION
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
      READ (5,1000) MAXI,MAXJ,NSEG,NBC,NMTL
      WRITE(6,2000) MAXI,MAXJ,NSEG,NBC,NMTL
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C     INITIALIZE
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
      ISEG=-1
      PI=3.1415927
      DO 110 J=1,80
      DO 100 I=1,25
      NCODE(I,J)=0
      AR(I,J)=0.
      AZ(I,J)=0.
      JMAX(I)=0
  100 JMIN(I)=MAXI
      IMIN(J)=MAXJ
  110 IMAX(J)=0
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C     LINE SEGMENT CARDS
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
  150 ISEG=ISEG+1
  159 IF(ISEG.EQ.NSEG) GO TO 400
      READ(5,1001) I1,J1,R1,Z1,I2,J2,R2,Z2,I3,J3,R3,Z3,IPTION
      WRITE(6,2001)I1,J1,R1,Z1,I2,J2,R2,Z2,I3,J3,R3,Z3,IPTION
      IPTION=IPTION+1
      AR(I1,J1)=R1
      AZ(I1,J1)=Z1
      NCODE(I1,J1)=1
      CALL MNIMX(I1,J1)
      GO TO (150,200,200,300,300,200,200), IPTION
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C     GENERATE STRAIGHT LINES ON BOUNDARY
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
200   DI= ABS(FLOAT(I2-I1))
      DJ= ABS(FLOAT(J2-J1))
      AR(I2,J2)=R2
      AZ(I2,J2)=Z2
      NCODE(I2,J2)=1
      CALL MNIMX(I2,J2)
      ISTRT=I1
      ISTP=I2
      JSTRT=J1
      JSTP=J2
      DIFF=MAX1(DI,DJ)
      ITER=DIFF-1.
      IINC=0
      JINC=0
      IF(I2.NE.I1) IINC=(I2-I1)/IABS(I2-I1)
```

```
      IF(J2.NE.J1) JINC=(J2-J1)/IABS(J2-J1)
      KAPPA=1
      IF(I2.NE.I1.AND.J2.NE.J1.AND.IPTION.NE.3) KAPPA=2
      IF(KAPPA.EQ.2) DIFF=2.*DIFF
      RINC=(R2-R1)/DIFF
      ZINC=(Z2-Z1)/DIFF
      WRITE(6,2002) DI,DJ,DIFF,RINC,ZINC,ITER,IINC,JINC,KAPPA
C
C     CHECK FOR INPUT ERROR
C
      IF(KAPPA.NE.2.OR.DI.EQ.DJ) GO TO 210
      WRITE(6,2003)
      GO TO 150
C
C     INTERPOLATE
C
  210 I=I1
      J=J1
      WRITE(6,2004)
      DO 230 M=1,ITER
      IF(ITER.EQ.0.AND.IPTION.EQ.2) GO TO 230
      IF(ITER.EQ.0.AND.IPTION.EQ.6) GO TO 230
      IF(ITER.EQ.0.AND.IPTION.EQ.7) GO TO 230
      IF(KAPPA.EQ.2) GO TO 220
      IOLD=I
      I=I+IINC
      JOLD=J
      J=J+JINC
      AR(I,J)=AR(IOLD,JOLD)+RINC
      AZ(I,J)=AZ(IOLD,JOLD)+ZINC
      WRITE(6,2005) I,J,AR(I,J),AZ(I,J)
      CALL MNIMX(I,J)
      NCODE(I,J)=1
      GO TO 230
  220 CONTINUE
      IF(I1.GT.I2.AND.IPTION.EQ.7) GO TO 221
      IF(I1.LT.I2.AND.IPTION.EQ.6) GO TO 221
      IOLD=I
      I=I+IINC
      AR(I,J)=AR(IOLD,J)+RINC
      AZ(I,J)=AZ(IOLD,J)+ZINC
      WRITE(6,2005) I,J,AR(I,J),AZ(I,J)
      NCODE(I,J)=1
      CALL MNIMX(I,J)
      JOLD=J
      J=J+JINC
      AR(I,J)=AR(I,JOLD)+RINC
      AZ(I,J)=AZ(I,JOLD)+ZINC
      NCODE(I,J)=1
      WRITE(6,2005) I,J,AR(I,J),AZ(I,J)
      CALL MNIMX(I,J)
      GO TO 230
```

```
  221 JOLD=J
      J=J+JINC
      AR(I,J)=AR(I,JOLD)+RINC
      AZ(I,J)=AZ(I,JOLD)+ZINC
      NCODE(I,J)=1
      WRITE(6,2005) I,J,AR(I,J),AZ(I,J)
      CALL MNIMX(I,J)
      IOLD=I
      I=I+IINC
      AR(I,J)=AR(IOLD,J)+RINC
      AZ(I,J)=AZ(IOLD,J)+ZINC
      NCODE(I,J)=1
      WRITE(6,2005) I,J,AR(I,J),AZ(I,J)
      CALL MNIMX(I,J)
  230 CONTINUE
      IF(KAPPA.EQ.1) GO TO 150
      IF(I1.GT.I2.AND.IPTION.EQ.7) GO TO 231
      IF(I1.LT.I2.AND.IPTION.EQ.6) GO TO 231
      IOLD=I
      I=I+IINC
      AR(I,J)=AR(IOLD,J)+RINC
      AZ(I,J)=AZ(IOLD,J)+ZINC
      GO TO 232
  231 CONTINUE
      JOLD=J
      J=J+JINC
      AR(I,J)=AR(I,JOLD)+RINC
      AZ(I,J)=AZ(I,JOLD)+ZINC
  232 CONTINUE
      NCODE(I,J)=1
      WRITE(6,2005) I,J,AR(I,J),AZ(I,J)
      CALL MNIMX(I,J)
      GO TO 150
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C     GENERATE CIRCULAR ARCS ON BOUNDARY
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
  300 AR(I2,J2)=R2
      AZ(I2,J2)=Z2
      NCODE(I2,J2) = 1
      CALL MNIMX(I2,J2)
      IF(IPTION.EQ.5) GO TO 320
C
C     FIND CENTER OF CIRCLE
C
      AR(I3,J3)=R3
      AZ(I3,J3)=Z3
      NCODE(I3,J3)=1
      CALL MNIMX(I3,J3)
      SLAC=(Z2-Z1)/(R2-R1)
      SLBF=-1./SLAC
      SLCE=(Z3-Z2)/(R3-R2)
      SLD       AR(I,J)=AR(IOLD,J)+RINC
      AZ(I,J)=AZ(IOLD,J)+ZINC
      GO TO 232
```

```
  231 CONTINUE
      JOLD=J
      J=J+JINC
      AR(I,J)=AR(I,JOLD)+RINC
      AZ(I,J)=AZ(I,JOLD)+ZINC
  232 CONTINUE
      NCODE(I,J)=1
      WRITE(6,2005) I,J,AR(I,J),AZ(I,J)
      CALL MNIMX(I,J)
      GO TO 150
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C     GENERATE CIRCULAR ARCS ON BOUNDARY
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
  300 AR(I2,J2)=R2
      AZ(I2,J2)=Z2
      NCODE(I2,J2) = 1
      CALL MNIMX(I2,J2)
      IF(IPTION.EQ.5) GO TO 320
C
C     FIND CENTER OF CIRCLE
C
      AR(I3,J3)=R3
      AZ(I3,J3)=Z3
      NCODE(I3,J3)=1
      CALL MNIMX(I3,J3)
      SLAC=(Z2-Z1)/(R2-R1)
      SLBF=-1./SLAC
      SLCE=(Z3-Z2)/(R3-R2)
      SLDF=-1./SLCE
C
C     CHECK FOR INPUT ERROR
C
      IF(ABS(SLAC-SLCE).GT..001) GO TO 310
      WRITE(6,2006) R1,Z1,R2,Z2,R3,Z3,SLAC,SLCE
      GO TO 150
  310 R4=R1+(R2-R1)/2.
      Z4=Z1+(Z2-Z1)/2.
      R5=R2+(R3-R2)/2.
      Z5=Z2+(Z3-Z2)/2.
      BBF=Z4-SLBF*R4
      BDF=Z5-SLDF*R5
      RC=(BBF-BDF)/(SLDF-SLBF)
      ZC=SLBF*RC+BBF
      WRITE(6,2007) RC,ZC
      KAPPA=1
      GO TO 330
  320 KAPPA=2
      RC=R3
      ZC=Z3
  330 ISTRT=I1
      ISTP=I2
      JSTRT=J1
```

```
      JSTP=J2
      RSTRT=R1
      RSTP=R2
      ZSTRT=Z1
      ZSTP=Z2
  340 CALL ANGLE(RSTRT,ZSTRT,RC,ZC,ANG1)
      CALL ANGLE(RSTP,ZSTP,RC,ZC,ANG2)
      IF(ANG2.LE.ANG1)  ANG2=2.0*PI+ANG2
C
C     FIND ANGULAR INCREMENT
C
      DI= ABS(FLOAT(ISTP-ISTRT))
      DJ= ABS(FLOAT(JSTP-JSTRT))
      IINC=0
      JINC=0
      IF(ISTRT.NE.ISTP) IINC=(ISTP-ISTRT)/IABS(ISTP-ISTRT)
      IF(JSTRT.NE.JSTP) JINC=(JSTP-JSTRT)/IABS(JSTP-JSTRT)
      LAMDA=1
      IF(IINC.NE.0.AND.JINC.NE.0) LAMDA=2
      DIFF=MAX1(DI,DJ)
      ITER=DIFF-1.
      IF(LAMDA.EQ.2) DIFF=2.*DIFF
      DELPHI=(ANG2-ANG1)/DIFF
      WRITE(6,2008) ANG1,ANG2,DIFF,DELPHI
C
C     CHECK FOR INPUT ERROR
C
      IF(LAMDA.NE.2.OR.DI.EQ.DJ) GO TO 350
      WRITE(6,2003)
      GO TO 150
  350 IO=ISTRT
      JO=JSTRT
      WRITE(6,2004)
C
C     INTERPOLATE
C
      NPT=IABS(I2-I1)+IABS(J2-J1)-1
      DO 380 M=1,ITER
  359 IF(LAMDA.EQ.2) GO TO 360
      I=IO+IINC
      J=JO+JINC
      CALL MNIMX(I,J)
      NCODE(I,J)=1
      CALL CIRCLE(ANG1,DELPHI,RSTRT,ZSTRT,RC,ZC,I,J)
      WRITE(6,2005) I,J,AR(I,J),AZ(I,J)
      GO TO 370
  360 I=IO+IINC
      J=JO
      NCODE(I,J)=1
      CALL MNIMX(I,J)
      CALL CIRCLE(ANG1,DELPHI,RSTRT,ZSTRT,RC,ZC,I,J)
      WRITE(6,2005) I,J,AR(I,J),AZ(I,J)
```

```
      J=JO+JINC
      NCODE(I,J)=1
      CALL MNIMX(I,J)
      CALL CIRCLE(ANG1,DELPHI,RSTRT,ZSTRT,RC,ZC,I,J)
      WRITE(6,2005) I,J,AR(I,J),AZ(I,J)
  370 IO=I
  380 JO=J
      IF(LAMDA.NE.2) GO TO 390
      I=IO+IINC
      NCODE(I,J)=1
      CALL MNIMX(I,J)
      CALL CIRCLE(ANG1,DELPHI,RSTRT,ZSTRT,RC,ZC,I,J)
      WRITE(6,2005) I,J,AR(I,J),AZ(I,J)
  390 IF(KAPPA.EQ.2) GO TO 150
      ISTRT=I2
      ISTP=I3
      JSTRT=J2
      JSTP=J3
      RSTRT=R2
      RSTP=R3
      ZSTRT=Z2
      ZSTP=Z3
      KAPPA=2
  399 GO TO 340
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C     CALCULATE COORDINATES OF INTERIOR POINTS
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
  400 IF(MAXJ.LE.2) GO TO 430
      J2=MAXJ-1
      DO 420 N=1,500
      RESID=0.
      DO 410 J=2,J2
      I1=IMIN(J)+1
      I2=IMAX(J)-1
      DO 410 I=I1,I2
      IF(NCODE(I,J).EQ.1) GO TO 410
      DR=(AR(I+1,J)+AR(I-1,J)+AR(I,J+1)+AR(I,J-1))/4.-AR(I,J)
      DZ=(AZ(I+1,J)+AZ(I-1,J)+AZ(I,J+1)+AZ(I,J-1))/4.-AZ(I,J)
      RESID=RESID+ABS(DR)+ABS(DZ)
      AR(I,J)=AR(I,J)+1.8*DR
      AZ(I,J)=AZ(I,J)+1.8*DZ
  410 CONTINUE
      IF(N.EQ.1) RES1=RESID
      IF(N.EQ.1.AND.RESID.EQ.0.)GO TO 430
      IF(RESID/RES1.LT.1.E-5) GO TO 430
  420 CONTINUE
  430 WRITE(6,2009) N
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
      CALL POINTS
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
 1000 FORMAT (5I5)
 1001 FORMAT (3(2I3,2F8.3),I5)
```

```
2000 FORMAT (30H1  MESH GENERATION INFORMATION//
     1 41H0  MAXIMUM VALUE OF I IN THE MESH--------I3/
     2 41H0  MAXIMUM VALUE OF J IN THE MESH--------I3/
     3 41H0  NUMBER OF LINE SEGMENT CARDS----------I3/
     4 41H0  NUMBER OF BOUNDARY CONDITION CARDS----I3/
     5 41H0  NUMBER OF MATERIAL BLOCK CARDS--------I3///)
2001 FORMAT (//88H  INPUT    I1  J1     R1       Z1     I2  J2     R2       Z
     12    I3  J3     R3       Z3     IPTION/8X,3(2I4,2F8.4),I6)
2002 FORMAT (5H  DI=F4.0,5H  DJ=F4.0,7H  DIFF=F4.0,7H  RINC=F8.3,7H  ZI
     1NC=F8.3,7H  ITER=I3,7H  IINC=I3,7H  JINC=I3,8H  KAPPA=I1)
2003 FORMAT(1X,38H**BAD INPUT--THIS LINE IS NOT DIAGONAL)
2004 FORMAT (30H    I    J        AR        AZ)
2005 FORMAT (2I5,2F11.6)
2006 FORMAT (51H ** BAD INPUT - THESE POINTS DO NOT DEFINE A CIRCLE,/,
     13X,6F12.4,10X,2E20.8)
2007 FORMAT(19H  CENTER COORDINATE,(F11.6,1X,F11.6,1X))
2008 FORMAT (7H  ANG1=F9.6,7H  ANG2=F9.6,7H  DIFF=F3.0,9H  DELPHI=F9.6)
2009 FORMAT (//30H  COORDINATES CALCULATED AFTER I3,11H ITERATIONS)
      RETURN
      END
      SUBROUTINE MNIMX(I,J)
      COMMON/TD/IMIN(100),IMAX(100),JMIN(25),JMAX(25),MAXI,MAXJ,NMTL,NBC
      IF(J.LT.JMIN(I)) JMIN(I)=J
      IF(J.GT.JMAX(I)) JMAX(I)=J
      IF(I.LT.IMIN(J)) IMIN(J)=I
      IF(I.GT.IMAX(J)) IMAX(J)=I
      RETURN
      END
      SUBROUTINE ADINA(NFACE,FTHETA)
      INTEGER CODE
      COMMON/NPDATA/R(1000),CODE(1000),XR(1000),Z(1000),XZ(1000),
     1NPNUM(25,80),T(1000),XT(1000)
      COMMON/ARG/RRR(5),ZZZ(5),RR(4),ZZ(4),S(15,15),P(15),TT(6),
     1H(6,15),CRZ(6,6),XI(10),ANGLE(4),SIG(18),EPS(18),N
      COMMON/ELDATA/BETA(1000),EPR(1000),PR(200),SH(200),IX(1000,5),
     1IP(200),JP(200),IS(200),JS(200),ALPHA(1000),IT(200),JT(200),
     2ST(200),INP(1000),JNP(1000)
      COMMON/BASIC/ACELZ,ANGVEL,ANGACC,TREF,VOL,NUMNP,NUMEL,NUMPC,NUMSC,
     1NUMST
      COMMON/NXMESH/THETAN(4),NST(4),NUMS(4,5),NPC(8,8)
      DIMENSION NPG(25,80,9),FTHETA(9),NCOD(8),NEL(8),ID(6)
      COMMON/ANS1/NUMELS(4),NUMNPS(4)
      COMMON/NXDATA/NTP,NTYPS,NTS,NTOTS
      COMMON/NONAXI/S1(30,30),P1(30),THETA,BS1(6,30)
      COMMON/SOLVE/X(4428),Y(4428),TEM(4428),NUMTC,MBAND
      COMMON/TD/IMIN(100),IMAX(100),JMIN(25),JMAX(25),MAXI,MAXJ,NMTL,NBC
      COMMON/CONVRG/IDONE
      COMMON/PLANE/NPP
      COMMON/RESULT/BS(6,15),D(6,6),C(6,6),AR,BB(6,9),CNS(6,6)
      COMMON/MATP/RO(6),E(12,16,6),EE(16),AOFTS(6)
      COMMON/TOTAL/NELN
      DATA NPG/18000*0/
```

```
      NI=NST(NTP)
C LOOP ON # OF SEGMENTS OF TYPE NTP
      DO 500 I=1,NI
      DO 200 J=1,NUMNP
      NFL=NUMS(NTP,I)
      NFR=NFL+1
      IP1=INP(J)
      JP1=JNP(J)
      NF=NFL
      DO 200 K=1,2
      NC=CODE(J)+1
      IF(NPG(IP1,JP1,NF).NE.0)GO TO 200
      NPG(IP1,JP1,NF)=IP1+(JP1-1)*25+NF*200
      DO 10 II=1,8
 10   ID(II)=0
      IF(NC.EQ.3.OR.NC.EQ.4.OR.NC.EQ.7.OR.NC.EQ.8) ID(1)=1
      IF(NF.EQ.1) ID(3)=1
      IF(NF.EQ.NTOTS+1) ID(2)=1
      IF(R(J).EQ.0.0) ID(2)=1
      IF(R(J).EQ.0.0) ID(3)=1
C
C     WRITE NODAL DATA ON FILE 2
C
      WRITE(2,100)NPG(IP1,JP1,NF),(ID(KK),KK=1,6),Z(J),R(J),FTHETA(NF)
  100 FORMAT("X",I4,1X,I4,5I5,3F10.4,4X,"0")
  200 NF=NFR
      DO 350 J=1,NUMEL
      NF=NUMS(NTP,I)
      I1=IX(J,1)
      I2=IX(J,2)
      I3=IX(J,3)
      I4=IX(J,4)
      IP1=INP(I1)
      JP1=JNP(I1)
      NEL(1)=NPG(IP1,JP1,NFL)
      IF(IP1.EQ.1)NEL(1)=NPG(IP1,JP1,1)
      NEL(4)=NPG(IP1,JP1,NFR)
      IP1=INP(I2)
      JP1=JNP(I2)
      NEL(2)=NPG(IP1,JP1,NFL)
      NEL(3)=NPG(IP1,JP1,NFR)
      IP1=INP(I4)
      JP1=JNP(I4)
      NEL(5)=NPG(IP1,JP1,NFL)
      IF(IP1.EQ.1)NEL(5)=NPG(IP1,JP1,1)
      NEL(8)=NPG(IP1,JP1,NFR)
      IP1=INP(I3)
      JP1=JNP(I3)
      NEL(6)=NPG(IP1,JP1,NFL)
      NEL(7)=NPG(IP1,JP1,NFR)
      NELN=NELN+1
      NZ=0
```

```
      N8=8
      WRITE(3,300)NELN,N8,N8,NZ,IX(J,5),NELN,NZ,NZ
  300 FORMAT(8I5,F10.0)
      WRITE(3,301) (NEL(K),K=1,8)
  301 FORMAT(8I5)
C
C     WRITE MATERIAL AXES DATA ON FILE 4
C
      NPI=NEL(1)+7000
      NPJ=NEL(1)+8000
      NPK=NEL(1)+9000
      WRITE(4,302) NELN,NPI,NPJ,NPK
  302 FORMAT(4I5)
C
C     WRITE NODAL POINTS FOR MATERIAL AXES
C
      FTRAD=FTHETA(NF)/57.29578
      CA=COS(ALPHA(J))
      CB=COS(BETA(J))
      SA=SIN(ALPHA(J))
      SB=SIN(BETA(J))
      CF=COS(FTRAD)
      SF=SIN(FTRAD)
      XPI=0.0
      YPI=0.0
      ZPI=0.0
      XPJ=CA*CB
      YPJ=CA*SB*CF+SA*SF
      ZPJ=CA*SB*SF-SA*CF
      XPK=SA*CB
      YPK=SA*SB*CF-CA*SF
      ZPK=SA*SB*SF+CA*CF
      WRITE(7,303)NPI,NZ,NZ,NZ,NZ,NZ,NZ,XPI,YPI,ZPI
      WRITE(7,303)NPJ,NZ,NZ,NZ,NZ,NZ,NZ,XPJ,YPJ,ZPJ
      WRITE(7,303)NPK,NZ,NZ,NZ,NZ,NZ,NZ,XPK,YPK,ZPK
  303 FORMAT(1X,I4,1X,I4,5I5,3F10.4,4X,"0")
  350 CONTINUE
      DO 500 L=1,NUMPC
      IP1=INP(IP(L))
      JP1=JNP(IP(L))
      IP2=INP(JP(L))
      JP2=JNP(JP(L))
      NEL(1)=NPG(IP1,JP1,NFL)
      NEL(2)=NPG(IP2,JP2,NFL)
      NEL(3)=NPG(IP2,JP2,NFR)
      NEL(4)=NPG(IP1,JP1,NFR)
      NCUR=1
      WRITE(1,400)NCUR,(NEL(K),K=1,4)
      WRITE(1,401) PR(L),PR(L),PR(L),PR(L)
  400 FORMAT(9I5)
  401 FORMAT(5F10.0,I5)
  500 CONTINUE
```

```
      END
      FUNCTION NODE(I,J)
      COMMON/TD/IMIN(100),IMAX(100),JMIN(25),JMAX(25),MAXI,MAXJ,NMTL,NBC
      NODE=0
      DO 100 JJ=1,J
      NSTART=IMIN(JJ)
      NSTOP=IMAX(JJ)
      DO 100 II=NSTART,NSTOP
      NODE=NODE+1
      IF(JJ.EQ.J.AND.II.EQ.I) RETURN
  100 CONTINUE
      RETURN
      END
      SUBROUTINE POINTS
      INTEGER CODE
      COMMON/BASIC/ACELZ,ANGVEL,ANGACC,TREF,VOL,NUMNP,NUMEL,NUMPC,NUMSC,
     1NUMST
      COMMON/MATP/RO(6),E(12,16,6),EE(16),AOFTS(6)
      COMMON/NPDATA/R(1000),CODE(1000),XR(1000),Z(1000),XZ(1000),
     1NPNUM(25,80),T(1000),XT(1000)
      COMMON/ELDATA/BETA(1000),EPR(1000),PR(200),SH(200),IX(1000,5),
     1IP(200),JP(200),IS(200),JS(200),ALPHA(1000),IT(200),JT(200),
     2ST(200),INP(1000),JNP(1000)
      COMMON/SOLVE/X(4428),Y(4428),TEM(4428),NUMTC,MBAND
      COMMON/TD/IMIN(100),IMAX(100),JMIN(25),JMAX(25),MAXI,MAXJ,NMTL,NBC
      COMMON/PLANE/NPP
      DIMENSION AR(25,80),AZ(25,80),MATRIL(100,5),BLKANG(100),BLKALF(1
     100)
      DIMENSION IBNG(100),NBNG(100)
      EQUIVALENCE (R(1),AR),(Z(1),AZ)
C     ESTABLISH NODAL POINT INFORMATION
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
      NEL=0
      NODSUM=0
      DO 100 J=1,MAXJ
      NSTART=IMIN(J)
      NSTOP=IMAX(J)
      DO 100 I=NSTART,NSTOP
  100 NODSUM=NODSUM+1
      NELSUM=0
      JJMAX=MAXJ-1
      DO 110 JJ=1,JJMAX
      NSTOP=MIN0(IMAX(JJ),IMAX(JJ+1))-1
      NSTART=MAX0(IMIN(JJ),IMIN(JJ+1))
      DO 110 II=NSTART,NSTOP
  110 NELSUM=NELSUM+1
      NUMNP=NODSUM
      NUMEL=NELSUM
      DO 120 J=1,MAXJ
      NSTART=IMIN(J)
      NSTOP=IMAX(J)
      DO 120 I=NSTART,NSTOP
```

```
      NPNUM(I,J)=NODE(I,J)
      NP=NPNUM(I,J)
      INP(NP)=I
      JNP(NP)=J
      R(NP)=AR(I,J)
  120 Z(NP)=AZ(I,J)
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C     READ AND ASSIGN BOUNDARY CONDITIONS
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C     INITIALIZE
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
      DO 130 I=1,NUMNP
      CODE(I)=0
      IF(R(I).EQ.0..AND.NPP.EQ.0) CODE(I)=1.
      XR(I)=0.
      XZ(I)=0.
      XT(I)=0.0
  130 T(I)=0.
      IF(NBC.EQ.0) GO TO 210
      DO 200 IBCON=1,NBC
      READ(5,1002) I1,I2,J1,J2,ICN,RCON,ZCON,TCON
      DO 200 I=I1,I2
      DO 200 J=J1,J2
      NP=NPNUM(I,J)
      CODE(NP)=ICN
      XR(NP)=RCON
      XT(NP)=TCON
  200 XZ(NP)=ZCON
  210 MPRINT=0
      DO 230 J=1,MAXJ
      NSTART=IMIN(J)
      NSTOP=IMAX(J)
      DO 230 I=NSTART,NSTOP
      NP=NPNUM(I,J)
      IF(MPRINT.NE.0) GO TO 220
      WRITE(6,2000)
      MPRINT=59
  220 MPRINT=MPRINT-1
  230 WRITE(6,2001) I,J,NP,CODE(NP),R(NP),Z(NP),XR(NP),XZ(NP),XT(NP)
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C     ASSIGN MATERIALS IN BLOCKS
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
      DO 300 M1=1,NUMEL
  300 IX(M1,5)=0
      DO 310 IMTL=1,NMTL
      READ (5,1000) MTL,(MATRIL(IMTL,IM),IM=2,5),BLKANG(IMTL),BLKALF(IMT
     1L),IBNG(IMTL),NBNG(IMTL)
  310 MATRIL(IMTL,1)=MTL
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C     ESTABLISH ELEMENT INFORMATION
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
      JJMAX=MAXJ-1
```

```
      N=0
      MTL=1
      KTL=1
      DO 440 JJ=1,JJMAX
      NSTOP=MIN0(IMAX(JJ),IMAX(JJ+1))-1
      NSTART=MAX0(IMIN(JJ),IMIN(JJ+1))
      DO 440 II=NSTART,NSTOP
      NEL=NEL+1
      DO 400 IMTL=1,NMTL
      IF(II.LT.MATRIL(IMTL,2)) GO TO 400
      IF(II.GE.MATRIL(IMTL,3)) GO TO 400
      IF(JJ.LT.MATRIL(IMTL,4)) GO TO 400
      IF(JJ.MR(T)GO0  KT      MATM(L) 4CIE  KQ.KTL) GOO
 G4                                                        K=KT  TA
    0(II.Q.NSTAXI.NE.NSTOP) GO TO 440
      M=NEL+1
      IANG=ICNG
      NANG=NCNG
      GO TO 421
 420  I=NPNUM(II,JJ)
=
=K    K=(,+
      IX(M,)  I2
                 I)=K
    X,L  I)
ANGFIBE(KT
       N     *INTERRUPTED*
get,diab
 *TERMINATED*
/diab,p=yes,s=41
 #PAGES=   53
```

```
      N=0
      MTL=1
      KTL=1
      DO 440 JJ=1,JJMAX
      NSTOP=MIN0(IMAX(JJ),IMAX(JJ+1))-1
      NSTART=MAX0(IMIN(JJ),IMIN(JJ+1))
      DO 440 II=NSTART,NSTOP
      NEL=NEL+1
      DO 400 IMTL=1,NMTL
      IF(II.LT.MATRIL(IMTL,2)) GO TO 400
      IF(II.GE.MATRIL(IMTL,3)) GO TO 400
      IF(JJ.LT.MATRIL(IMTL,4)) GO TO 400
      IF(JJ.GE.MATRIL(IMTL,5)) GO TO 400
      KAT=IMTL
      MAT=MATRIL(IMTL,1)
  400 CONTINUE
      IF(KAT.EQ.KTL) GO TO 410
      KTL=KAT
      MTL=MAT
      GO TO 420
  410 IF(II.EQ.NSTART) GO TO 420
      IF(JJ.NE.JJMAX.OR.II.NE.NSTOP) GO TO 440
      M=NEL+1
      IANG=ICNG
      NANG=NCNG
      GO TO 421
  420 I=NPNUM(II,JJ)
      J=I+1
      K=NPNUM(II+1,JJ+1)
      L=K-1
      M=NEL
      IX(M,1)=I
      IX(M,2)=J
      IX(M,3)=K
      IX(M,4)=L
      IX(M,5)=MTL
      BETA(M)=BLKANG(KTL)
      ALPHA(M)=BLKALF(KTL)
      IANG=ICNG
      NANG=NCNG
      ICNG=IBNG(KTL)
      NCNG=NBNG(KTL)
  421 NC=2
  430 N=N+1
      IF(M.LE.N) GO TO 440
      IX(N,1)=IX(N-1,1)+1
      IX(N,2)=IX(N-1,2)+1
      IX(N,3)=IX(N-1,3)+1
      IX(N,4)=IX(N-1,4)+1
      IX(N,5)=IX(N-1,5)
      BETA(N)=BETA(N-1)
      IF(IANG.EQ.1) GO TO 442
```

```
      ALPHA(N)=ALPHA(N-1)
      GO TO 443
  442 CONTINUE
      IF(NC.GT.NANG) GO TO 444
      ALPHA(N)=ALPHA(N-1)
      GO TO 443
  444 NC=1
      ALPHA(N)=-ALPHA(N-1)
  443 CONTINUE
      NC=NC+1
      IF(M.GT.N) GO TO 430
  440 CONTINUE
      IF(NUMNP.GT.2000) WRITE(6,2002)
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C     SET NODAL POINT TEMPERATURE TO REFERENCE TEMPERATURE
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
      IF(NUMTC.NE.0) RETURN
      DO 500 N=1,NUMNP
  500 T(N)=TREF
 1000 FORMAT (5I5,2F10.0,2I5)
 1002 FORMAT(4I5,I10,3F10.0)
 2000 FORMAT (128H1   I    J    NP        TYPE    R-ORDINATE    Z-ORDINA
     1TE  R LOAD OR DISPLACEMENT  Z LOAD OR DISPLACEMENT  T LOAD OR DISP
     2LACEMENT)
 2001 FORMAT (2I5,I6,I12,F13.6,F14.6,E26.7,E24.7,E24.7)
 2002 FORMAT (35H  BAD INPUT - TOO MANY NODAL POINTS)
      RETURN
      END
      SUBROUTINE TEMP(R,Z,T)
      COMMON/SOLVE/X(4428),Y(4428),TEM(4428),NUMTC,MBAND
      DIMENSION SMALL(20),ISM(20)
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C     INITIALIZE
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
      J=1
      JMAX=16
      IF(NUMTC.LT.JMAX) JMAX=NUMTC
      DO 10 I=1,JMAX
      SMALL(I)=0.
   10 ISM(I)=0
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C     FIND THE JMAX CLOSEST POINTS
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
      DO 50 I=1,NUMTC
      DSQ=(X(I)-R)**2+(Y(I)-Z)**2
      IF(DSQ.GT..1E-4) GO TO 20
      T=TEM(I)
      RETURN
   20 IF(I.EQ.1) SMALL(1)=DSQ
      IF(I.EQ.1) ISM(1)=1
      IF(I.EQ.1) GO TO 50
      IF(SMALL(J).LE.DSQ.AND.J.LT.JMAX) SMALL(J+1)=DSQ
```

```
      IF(SMALL(J).LE.DSQ.AND.J.LT.JMAX) ISM(J+1)=I
      IF(SMALL(J).LE.DSQ) GO TO 40
      DO 30 K=1,J
      JB=J-K +1
      IF(JB.EQ.0) GO TO 40
      SMALL(JB+1)=SMALL(JB)
      ISM(JB+1)=ISM(JB)
      SMALL(JB)=DSQ
      ISM(JB)=I
      IF(JB.EQ.1) GO TO 40
      IF(SMALL(JB-1).LE.DSQ) GO TO 40
   30 CONTINUE
   40 IF(J.LT.JMAX) J=J+1
   50 CONTINUE
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C     FIND THE THIRD TEMPERATURE POINT BY AREA TEST
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
      JCHK=JMAX-2
      J=0
      I1=ISM(1)
      I2=ISM(2)
   60 I3=ISM(J+3)
      AREA=.50*(Y(I1)*X(I3)-Y(I3)*X(I1)+Y(I3)*X(I2)-Y(I2)*X(I3)+
     1        Y(I2)*X(I1)-Y(I1)*X(I2))
      D1=(X(I2)-X(I1))**2+(Y(I2)-Y(I1))**2
C     IF D1 IS APPROXIMATELY 0. IT IS ASSUMED THAT THERE EXISTS A
C     DUPLICATION OF INPUT
      IF(D1.GT..1E-3) GO TO 70
      I2=I3
      J=J+1
      GO TO 60
   70 IF(AREA**2.GT..1*D1*SMALL(1)) GO TO 80
      J=J+1
      IF(J.LT.JCHK) GO TO 60
      WRITE(6,2000) I1,I2,I3,J
      T=TEM(I1)
      RETURN
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C     FIND TEMPERATURE INTERCEPT
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
   80 DETA=Y(I1)*(TEM(I3)-TEM(I2))+Y(I2)*(TEM(I1)-TEM(I3))
     1      +Y(I3)*(TEM(I2)-TEM(I1))
      DETB=X(I1)*(TEM(I2)-TEM(I3))+X(I2)*(TEM(I3)-TEM(I1))
     1      +X(I3)*(TEM(I1)-TEM(I2))
      DETC=TEM(I1)*(X(I2)*Y(I3)-X(I3)*Y(I2))+TEM(I2)*(X(I3)*Y(I1)-X(I1)*
     1Y(I3))+TEM(I3)*(X(II)*Y(I2)-X(I2)*Y(II))
      T=(DETA*R+DETB*Z+DETC)/(2.*AREA)
 2000 FORMAT (28H  ERROR IN TEMPERATURE INPUT,5H  I1=I4,5H  I2=I4,
     15H  I3=I4,4H  J=I4)
      RETURN
      END
      SUBROUTINE TEM2(NUMNP)
```

```
      INTEGER CODE
      COMMON/NPDATA/R(1000),CODE(1000),XR(1000),Z(1000),XZ(1000),
     1NPNUM(25,80),T(1000),XT(1000)
      READ(5,1000) TCONST
      DO 100 N=1,NUMNP
  100 T(N)=TCONST
 1000 FORMAT(F10.0)
      RETURN
      END
      END
N
```

APPENDIX C

Listing of File ADI,

the Input Cards for ADINA Program

X 201	1	1	1	0	0	0	7.9950	0.0000	0.0000	0 ADNPD
X 401	1	1	1	0	0	0	7.9950	0.0000	18.8000	0
X 202	1	1	1	0	0	0	7.9950	.2363	0.0000	0
X 402	1	1	1	0	0	0	7.9950	.2363	18.8000	0
X 203	1	1	1	0	0	0	7.9950	.4725	0.0000	0
X 403	1	1	1	0	0	0	7.9950	.4725	18.8000	0
X 204	1	1	1	0	0	0	7.9950	.5050	0.0000	0
X 404	1	1	1	0	0	0	7.9950	.5050	18.8000	0
X 205	1	1	1	0	0	0	7.9950	.5375	0.0000	0
X 405	1	1	1	0	0	0	7.9950	.5375	18.8000	0
X 206	1	1	1	0	0	0	8.0218	1.0467	0.0000	0
X 406	1	1	1	0	0	0	8.0218	1.0467	18.8000	0
X 207	0	0	1	0	0	0	8.0486	1.5558	0.0000	0
X 407	0	0	1	0	0	0	8.0486	1.5558	18.8000	0
X 208	0	0	1	0	0	0	8.0754	2.0650	0.0000	0
X 408	0	0	1	0	0	0	8.0754	2.0650	18.8000	0
X 226	0	1	1	0	0	0	9.3024	0.0000	0.0000	0
X 426	0	1	1	0	0	0	9.3024	0.0000	18.8000	0
X 227	0	0	1	0	0	0	9.3234	.2362	0.0000	0
X 427	0	0	1	0	0	0	9.3234	.2362	18.8000	0
X 228	0	0	1	0	0	0	9.3024	.4725	0.0000	0
X 428	0	0	1	0	0	0	9.3024	.4725	18.8000	0
X 229	0	0	1	0	0	0	9.3230	.5091	0.0000	0
X 429	0	0	1	0	0	0	9.3230	.5091	18.8000	0
X 230	0	0	1	0	0	0	9.3024	.5375	0.0000	0
X 430	0	0	1	0	0	0	9.3024	.5375	18.8000	0
X 231	0	0	1	0	0	0	9.2469	1.0393	0.0000	0
X 431	0	0	1	0	0	0	9.2469	1.0393	18.8000	0
X 232	0	0	1	0	0	0	9.0533	1.5509	0.0000	0
X 432	0	0	1	0	0	0	9.0533	1.5509	18.8000	0
X 233	0	0	1	0	0	0	8.8254	2.0650	0.0000	0
X 433	0	0	1	0	0	0	8.8254	2.0650	18.8000	0
X 251	0	1	1	0	0	0	10.6098	0.0000	0.0000	0
X 451	0	1	1	0	0	0	10.6098	0.0000	18.8000	0
X 252	0	0	1	0	0	0	10.6944	.2363	0.0000	0
X 452	0	0	1	0	0	0	10.6944	.2363	18.8000	0
X 253	0	0	1	0	0	0	10.6098	.4725	0.0000	0
X 453	0	0	1	0	0	0	10.6098	.4725	18.8000	0
X 254	0	0	1	0	0	0	10.6919	.5214	0.0000	0
X 454	0	0	1	0	0	0	10.6919	.5214	18.8000	0
X 255	0	0	1	0	0	0	10.6098	.5375	0.0000	0
X 455	0	0	1	0	0	0	10.6098	.5375	18.8000	0
X 256	0	0	1	0	0	0	10.6098	1.0222	0.0000	0
X 456	0	0	1	0	0	0	10.6098	1.0222	18.8000	0
X 257	0	0	1	0	0	0	10.0926	1.5436	0.0000	0
X 457	0	0	1	0	0	0	10.0926	1.5436	18.8000	0
X 258	0	0	1	0	0	0	9.5754	2.0650	0.0000	0
X 458	0	0	1	0	0	0	9.5754	2.0650	18.8000	0
X 276	0	1	1	0	0	0	12.2166	0.0000	0.0000	0
X 476	0	1	1	0	0	0	12.2166	0.0000	18.8000	0
X 277	0	0	1	0	0	0	12.2338	.2362	0.0000	0
X 477	0	0	1	0	0	0	12.2338	.2362	18.8000	0
X 278	0	0	1	0	0	0	12.2300	.4725	0.0000	0
X 478	0	0	1	0	0	0	12.2300	.4725	18.8000	0
X 279	0	0	1	0	0	0	12.2255	.5666	0.0000	0
X 479	0	0	1	0	0	0	12.2255	.5666	18.8000	0
X 280	0	0	1	0	0	0	12.2211	.6607	0.0000	0
X 480	0	0	1	0	0	0	12.2211	.6607	18.8000	0
X 281	0	0	1	0	0	0	12.2166	.7548	0.0000	0

X 481	0	0	1	0	0	0	12.2166	.7548	18.8000	0
X 301	0	1	1	0	0	0	13.8234	0.0000	0.0000	0
X 501	0	1	1	0	0	0	13.8234	0.0000	18.8000	0
X 302	0	0	1	0	0	0	13.7950	.2363	0.0000	0
X 502	0	0	1	0	0	0	13.7950	.2363	18.8000	0
X 303	0	0	1	0	0	0	13.8234	.4725	0.0000	0
X 503	0	0	1	0	0	0	13.8234	.4725	18.8000	0
X 304	0	0	1	0	0	0	13.8234	.4775	0.0000	0
X 504	0	0	1	0	0	0	13.8234	.4775	18.8000	0
X 305	0	0	1	0	0	0	13.8234	.4825	0.0000	0
X 505	0	0	1	0	0	0	13.8234	.4825	18.8000	0
X 306	0	0	1	0	0	0	13.8234	.4875	0.0000	0
X 506	0	0	1	0	0	0	13.8234	.4875	18.8000	0
X 326	0	1	1	0	0	0	15.3068	0.0000	0.0000	0
X 526	0	1	1	0	0	0	15.3068	0.0000	18.8000	0
X 327	0	0	1	0	0	0	15.2992	.2362	0.0000	0
X 527	0	0	1	0	0	0	15.2992	.2362	18.8000	0
X 328	0	0	1	0	0	0	15.3068	.4725	0.0000	0
X 528	0	0	1	0	0	0	15.3068	.4725	18.8000	0
X 351	0	1	1	0	0	0	16.7903	0.0000	0.0000	0
X 551	0	1	1	0	0	0	16.7903	0.0000	18.8000	0
X 352	0	0	1	0	0	0	16.7883	.2362	0.0000	0
X 552	0	0	1	0	0	0	16.7883	.2362	18.8000	0
X 353	0	0	1	0	0	0	16.7903	.4725	0.0000	0
X 553	0	0	1	0	0	0	16.7903	.4725	18.8000	0
X 376	0	1	1	0	0	0	18.2737	0.0000	0.0000	0
X 576	0	1	1	0	0	0	18.2737	0.0000	18.8000	0
X 377	0	0	1	0	0	0	18.2737	.2363	0.0000	0
X 577	0	0	1	0	0	0	18.2737	.2363	18.8000	0
X 378	0	0	1	0	0	0	18.2737	.4725	0.0000	0
X 578	0	0	1	0	0	0	18.2737	.4725	18.8000	0
X 601	0	0	1	0	0	0	7.9950	0.0000	37.6000	0
X 602	0	0	1	0	0	0	7.9950	.2363	37.6000	0
X 603	0	0	1	0	0	0	7.9950	.4725	37.6000	0
X 604	0	0	1	0	0	0	7.9950	.5050	37.6000	0
X 605	0	0	1	0	0	0	7.9950	.5375	37.6000	0
X 606	0	0	1	0	0	0	8.0218	1.0467	37.6000	0
X 607	0	0	1	0	0	0	8.0486	1.5558	37.6000	0
X 608	0	0	1	0	0	0	8.0754	2.0650	37.6000	0
X 626	0	0	1	0	0	0	9.3024	0.0000	37.6000	0
X 627	0	0	1	0	0	0	9.3234	.2362	37.6000	0
X 628	0	0	1	0	0	0	9.3024	.4725	37.6000	0
X 629	0	0	1	0	0	0	9.3230	.5091	37.6000	0
X 630	0	0	1	0	0	0	9.3024	.5375	37.6000	0
X 631	0	0	1	0	0	0	9.2469	1.0393	37.6000	0
X 632	0	0	1	0	0	0	9.0533	1.5509	37.6000	0
X 633	0	0	1	0	0	0	8.8254	2.0650	37.6000	0
X 651	0	0	1	0	0	0	10.6098	0.0000	37.6000	0
X 652	0	0	1	0	0	0	10.6944	.2363	37.6000	0
X 653	0	0	1	0	0	0	10.6098	.4725	37.6000	0
X 654	0	0	1	0	0	0	10.6919	.5214	37.6000	0
X 655	0	0	1	0	0	0	10.6098	.5375	37.6000	0
X 656	0	0	1	0	0	0	10.6098	1.0222	37.6000	0
X 657	0	0	1	0	0	0	10.0926	1.5436	37.6000	0
X 658	0	0	1	0	0	0	9.5754	2.0650	37.6000	0
X 676	0	0	1	0	0	0	12.2166	0.0000	37.6000	0
X 677	0	0	1	0	0	0	12.2338	.2362	37.6000	0
X 678	0	0	1	0	0	0	12.2300	.4725	37.6000	0
X 679	0	0	1	0	0	0	12.2255	.5666	37.6000	0
X 680	0	0	1	0	0	0	12.2211	.6607	37.6000	0

X 681	0	0	1	0	0	0	12.2166	.7548	37.6000	0
X 701	0	0	1	0	0	0	13.8234	0.0000	37.6000	0
X 702	0	0	1	0	0	0	13.7950	.2363	37.6000	0
X 703	0	0	1	0	0	0	13.8234	.4725	37.6000	0
X 704	0	0	1	0	0	0	13.8234	.4775	37.6000	0
X 705	0	0	1	0	0	0	13.8234	.4825	37.6000	0
X 706	0	0	1	0	0	0	13.8234	.4875	37.6000	0
X 726	0	0	1	0	0	0	15.3068	0.0000	37.6000	0
X 727	0	0	1	0	0	0	15.2992	.2362	37.6000	0
X 728	0	0	1	0	0	0	15.3068	.4725	37.6000	0
X 751	0	0	1	0	0	0	16.7903	0.0000	37.6000	0
X 752	0	0	1	0	0	0	16.7883	.2362	37.6000	0
X 753	0	0	1	0	0	0	16.7903	.4725	37.6000	0
X 776	0	0	1	0	0	0	18.2737	0.0000	37.6000	0
X 777	0	0	1	0	0	0	18.2737	.2363	37.6000	0
X 778	0	0	1	0	0	0	18.2737	.4725	37.6000	0
X 801	0	0	1	0	0	0	7.9950	0.0000	56.4000	0
X 802	0	0	1	0	0	0	7.9950	.2363	56.4000	0
X 803	0	0	1	0	0	0	7.9950	.4725	56.4000	0
X 804	0	0	1	0	0	0	7.9950	.5050	56.4000	0
X 805	0	0	1	0	0	0	7.9950	.5375	56.4000	0
X 806	0	0	1	0	0	0	8.0218	1.0467	56.4000	0
X 807	0	0	1	0	0	0	8.0486	1.5558	56.4000	0
X 808	0	0	1	0	0	0	8.0754	2.0650	56.4000	0
X 826	0	0	1	0	0	0	9.3024	0.0000	56.4000	0
X 827	0	0	1	0	0	0	9.3234	.2362	56.4000	0
X 828	0	0	1	0	0	0	9.3024	.4725	56.4000	0
X 829	0	0	1	0	0	0	9.3230	.5091	56.4000	0
X 830	0	0	1	0	0	0	9.3024	.5375	56.4000	0
X 831	0	0	1	0	0	0	9.2469	1.0393	56.4000	0
X 832	0	0	1	0	0	0	9.0533	1.5509	56.4000	0
X 833	0	0	1	0	0	0	8.8254	2.0650	56.4000	0
X 851	0	0	1	0	0	0	10.6098	0.0000	56.4000	0
X 852	0	0	1	0	0	0	10.6944	.2363	56.4000	0
X 853	0	0	1	0	0	0	10.6098	.4725	56.4000	0
X 854	0	0	1	0	0	0	10.6919	.5214	56.4000	0
X 855	0	0	1	0	0	0	10.6098	.5375	56.4000	0
X 856	0	0	1	0	0	0	10.6098	1.0222	56.4000	0
X 857	0	0	1	0	0	0	10.0926	1.5436	56.4000	0
X 858	0	0	1	0	0	0	9.5754	2.0650	56.4000	0
X 876	0	0	1	0	0	0	12.2166	0.0000	56.4000	0
X 877	0	0	1	0	0	0	12.2338	.2362	56.4000	0
X 878	0	0	1	0	0	0	12.2300	.4725	56.4000	0
X 879	0	0	1	0	0	0	12.2255	.5666	56.4000	0
X 880	0	0	1	0	0	0	12.2211	.6607	56.4000	0
X 881	0	0	1	0	0	0	12.2166	.7548	56.4000	0
X 901	0	0	1	0	0	0	13.8234	0.0000	56.4000	0
X 902	0	0	1	0	0	0	13.7950	.2363	56.4000	0
X 903	0	0	1	0	0	0	13.8234	.4725	56.4000	0
X 904	0	0	1	0	0	0	13.8234	.4775	56.4000	0
X 905	0	0	1	0	0	0	13.8234	.4825	56.4000	0
X 906	0	0	1	0	0	0	13.8234	.4875	56.4000	0
X 926	0	0	1	0	0	0	15.3068	0.0000	56.4000	0
X 927	0	0	1	0	0	0	15.2992	.2362	56.4000	0
X 928	0	0	1	0	0	0	15.3068	.4725	56.4000	0
X 951	0	0	1	0	0	0	16.7903	0.0000	56.4000	0
X 952	0	0	1	0	0	0	16.7883	.2362	56.4000	0
X 953	0	0	1	0	0	0	16.7903	.4725	56.4000	0
X 976	0	0	1	0	0	0	18.2737	0.0000	56.4000	0
X 977	0	0	1	0	0	0	18.2737	.2363	56.4000	0

X 978	0	0	1	0	0	0	18.2737	.4725	56.4000	0
X1001	1	1	1	0	0	0	7.9950	0.0000	59.9000	0
X1002	1	1	1	0	0	0	7.9950	.2363	59.9000	0
X1003	1	1	1	0	0	0	7.9950	.4725	59.9000	0
X1004	1	1	1	0	0	0	7.9950	.5050	59.9000	0
X1005	1	1	1	0	0	0	7.9950	.5375	59.9000	0
X1006	1	1	1	0	0	0	8.0218	1.0467	59.9000	0
X1007	0	0	1	0	0	0	8.0486	1.5558	59.9000	0
X1008	0	0	1	0	0	0	8.0754	2.0650	59.9000	0
X1026	0	1	1	0	0	0	9.3024	0.0000	59.9000	0
X1027	0	0	1	0	0	0	9.3237	.2363	59.9000	0
X1028	0	0	1	0	0	0	9.3024	.4725	59.9000	0
X1029	0	0	1	0	0	0	9.3229	.5091	59.9000	0
X1030	0	0	1	0	0	0	9.3024	.5375	59.9000	0
X1031	0	0	1	0	0	0	9.2273	1.0241	59.9000	0
X1032	0	0	1	0	0	0	9.0485	1.5471	59.9000	0
X1033	0	0	1	0	0	0	8.8254	2.0650	59.9000	0
X1051	0	1	1	0	0	0	10.6098	0.0000	59.9000	0
X1052	0	0	1	0	0	0	10.6942	.2362	59.9000	0
X1053	0	0	1	0	0	0	10.6098	.4725	59.9000	0
X1054	0	0	1	0	0	0	10.6920	.5214	59.9000	0
X1055	0	0	1	0	0	0	10.6098	.5375	59.9000	0
X1056	1	1	1	0	0	0	10.5366	.9650	59.9000	0
X1057	1	1	1	0	0	0	10.0926	1.5436	59.9000	0
X1058	0	0	1	0	0	0	9.5754	2.0650	59.9000	0
X1076	0	1	1	0	0	0	12.2166	0.0000	59.9000	0
X1077	0	0	1	0	0	0	12.2341	.2363	59.9000	0
X1078	0	0	1	0	0	0	12.2300	.4725	59.9000	0
X1079	0	0	1	0	0	0	12.2255	.5666	59.9000	0
X1080	0	0	1	0	0	0	12.2211	.6607	59.9000	0
X1081	1	1	1	0	0	0	12.2166	.7548	59.9000	0
X 882	1	1	1	0	0	0	11.9580	1.5436	56.4000	0
X1082	0	0	1	0	0	0	11.9580	1.5436	59.9000	0
X1101	0	1	1	0	0	0	13.8234	0.0000	59.9000	0
X1102	0	0	1	0	0	0	13.7950	.2363	59.9000	0
X1103	0	0	1	0	0	0	13.8234	.4725	59.9000	0
X1104	0	0	1	0	0	0	13.8234	.4775	59.9000	0
X1105	0	0	1	0	0	0	13.8234	.4825	59.9000	0
X1106	1	1	1	0	0	0	13.8234	.4875	59.9000	0
X 907	1	1	1	0	0	0	13.8234	1.5436	56.4000	0
X1107	0	0	1	0	0	0	13.8234	1.5436	59.9000	0
X1126	0	1	1	0	0	0	15.3068	0.0000	59.9000	0
X1127	0	0	1	0	0	0	15.2993	.2363	59.9000	0
X1128	0	0	1	0	0	0	15.3068	.4725	59.9000	0
X1151	0	1	1	0	0	0	16.7903	0.0000	59.9000	0
X1152	0	0	1	0	0	0	16.7884	.2363	59.9000	0
X1153	0	0	1	0	0	0	16.7903	.4725	59.9000	0
X1176	0	1	1	0	0	0	18.2737	0.0000	59.9000	0
X1177	0	0	1	0	0	0	18.2737	.2363	59.9000	0
X1178	0	0	1	0	0	0	18.2737	.4725	59.9000	0
7201	0	0	0	0	0	0	0.0000	0.0000	0.0000	0 ADNPX
8201	0	0	0	0	0	0	1.0000	0.0000	0.0000	0
9201	0	0	0	0	0	0	0.0000	0.0000	1.0000	0
7202	0	0	0	0	0	0	0.0000	0.0000	0.0000	0
8202	0	0	0	0	0	0	1.0000	0.0000	0.0000	0
9202	0	0	0	0	0	0	0.0000	0.0000	1.0000	0
7203	0	0	0	0	0	0	0.0000	0.0000	0.0000	0
8203	0	0	0	0	0	0	1.0000	0.0000	0.0000	0
9203	0	0	0	0	0	0	0.0000	0.0000	1.0000	0
7204	0	0	0	0	0	0	0.0000	0.0000	0.0000	0

8204	0	0	0	0	0	0	1.0000	0.0000	0.0000	0
9204	0	0	0	0	0	0	0.0000	0.0000	1.0000	0
7205	0	0	0	0	0	0	0.0000	0.0000	0.0000	0
8205	0	0	0	0	0	0	1.0000	0.0000	0.0000	0
9205	0	0	0	0	0	0	0.0000	0.0000	1.0000	0
7206	0	0	0	0	0	0	0.0000	0.0000	0.0000	0
8206	0	0	0	0	0	0	1.0000	0.0000	0.0000	0
9206	0	0	0	0	0	0	0.0000	0.0000	1.0000	0
7207	0	0	0	0	0	0	0.0000	0.0000	0.0000	0
8207	0	0	0	0	0	0	1.0000	0.0000	0.0000	0
9207	0	0	0	0	0	0	0.0000	0.0000	1.0000	0
7226	0	0	0	0	0	0	0.0000	0.0000	0.0000	0
8226	0	0	0	0	0	0	1.0000	0.0000	0.0000	0
9226	0	0	0	0	0	0	0.0000	0.0000	1.0000	0
7227	0	0	0	0	0	0	0.0000	0.0000	0.0000	0
8227	0	0	0	0	0	0	1.0000	0.0000	0.0000	0
9227	0	0	0	0	0	0	0.0000	0.0000	1.0000	0
7228	0	0	0	0	0	0	0.0000	0.0000	0.0000	0
8228	0	0	0	0	0	0	1.0000	0.0000	0.0000	0
9228	0	0	0	0	0	0	0.0000	0.0000	1.0000	0
7229	0	0	0	0	0	0	0.0000	0.0000	0.0000	0
8229	0	0	0	0	0	0	1.0000	0.0000	0.0000	0
9229	0	0	0	0	0	0	0.0000	0.0000	1.0000	0
7230	0	0	0	0	0	0	0.0000	0.0000	0.0000	0
8230	0	0	0	0	0	0	1.0000	0.0000	0.0000	0
9230	0	0	0	0	0	0	0.0000	0.0000	1.0000	0
7231	0	0	0	0	0	0	0.0000	0.0000	0.0000	0
8231	0	0	0	0	0	0	1.0000	0.0000	0.0000	0
9231	0	0	0	0	0	0	0.0000	0.0000	1.0000	0
7232	0	0	0	0	0	0	0.0000	0.0000	0.0000	0
8232	0	0	0	0	0	0	1.0000	0.0000	0.0000	0
9232	0	0	0	0	0	0	0.0000	0.0000	1.0000	0
7251	0	0	0	0	0	0	0.0000	0.0000	0.0000	0
8251	0	0	0	0	0	0	1.0000	0.0000	0.0000	0
9251	0	0	0	0	0	0	0.0000	0.0000	1.0000	0
7252	0	0	0	0	0	0	0.0000	0.0000	0.0000	0
8252	0	0	0	0	0	0	1.0000	0.0000	0.0000	0
9252	0	0	0	0	0	0	0.0000	0.0000	1.0000	0
7253	0	0	0	0	0	0	0.0000	0.0000	0.0000	0
8253	0	0	0	0	0	0	1.0000	0.0000	0.0000	0
9253	0	0	0	0	0	0	0.0000	0.0000	1.0000	0
7254	0	0	0	0	0	0	0.0000	0.0000	0.0000	0
8254	0	0	0	0	0	0	1.0000	0.0000	0.0000	0
9254	0	0	0	0	0	0	0.0000	0.0000	1.0000	0
7255	0	0	0	0	0	0	0.0000	0.0000	0.0000	0
8255	0	0	0	0	0	0	1.0000	0.0000	0.0000	0
9255	0	0	0	0	0	0	0.0000	0.0000	1.0000	0
7276	0	0	0	0	0	0	0.0000	0.0000	0.0000	0
8276	0	0	0	0	0	0	1.0000	0.0000	0.0000	0
9276	0	0	0	0	0	0	0.0000	0.0000	1.0000	0
7277	0	0	0	0	0	0	0.0000	0.0000	0.0000	0
8277	0	0	0	0	0	0	1.0000	0.0000	0.0000	0
9277	0	0	0	0	0	0	0.0000	0.0000	1.0000	0
7278	0	0	0	0	0	0	0.0000	0.0000	0.0000	0
8278	0	0	0	0	0	0	1.0000	0.0000	0.0000	0
9278	0	0	0	0	0	0	0.0000	0.0000	1.0000	0
7279	0	0	0	0	0	0	0.0000	0.0000	0.0000	0
8279	0	0	0	0	0	0	1.0000	0.0000	0.0000	0
9279	0	0	0	0	0	0	0.0000	0.0000	1.0000	0
7280	0	0	0	0	0	0	0.0000	0.0000	0.0000	0

8280	0	0	0	0	0	0	1.0000	0.0000	0.0000	0
9280	0	0	0	0	0	0	0.0000	0.0000	1.0000	0
7301	0	0	0	0	0	0	0.0000	0.0000	0.0000	0
8301	0	0	0	0	0	0	1.0000	0.0000	0.0000	0
9301	0	0	0	0	0	0	0.0000	0.0000	1.0000	0
7302	0	0	0	0	0	0	0.0000	0.0000	0.0000	0
8302	0	0	0	0	0	0	1.0000	0.0000	0.0000	0
9302	0	0	0	0	0	0	0.0000	0.0000	1.0000	0
7326	0	0	0	0	0	0	0.0000	0.0000	0.0000	0
8326	0	0	0	0	0	0	1.0000	0.0000	0.0000	0
9326	0	0	0	0	0	0	0.0000	0.0000	1.0000	0
7327	0	0	0	0	0	0	0.0000	0.0000	0.0000	0
8327	0	0	0	0	0	0	1.0000	0.0000	0.0000	0
9327	0	0	0	0	0	0	0.0000	0.0000	1.0000	0
7351	0	0	0	0	0	0	0.0000	0.0000	0.0000	0
8351	0	0	0	0	0	0	1.0000	0.0000	0.0000	0
9351	0	0	0	0	0	0	0.0000	0.0000	1.0000	0
7352	0	0	0	0	0	0	0.0000	0.0000	0.0000	0
8352	0	0	0	0	0	0	1.0000	0.0000	0.0000	0
9352	0	0	0	0	0	0	0.0000	0.0000	1.0000	0
7201	0	0	0	0	0	0	0.0000	0.0000	0.0000	0
8201	0	0	0	0	0	0	1.0000	0.0000	0.0000	0
9201	0	0	0	0	0	0	0.0000	-.3223	.9466	0
7402	0	0	0	0	0	0	0.0000	0.0000	0.0000	0
8402	0	0	0	0	0	0	1.0000	0.0000	0.0000	0
9402	0	0	0	0	0	0	0.0000	-.3223	.9466	0
7403	0	0	0	0	0	0	0.0000	0.0000	0.0000	0
8403	0	0	0	0	0	0	1.0000	0.0000	0.0000	0
9403	0	0	0	0	0	0	0.0000	-.3223	.9466	0
7404	0	0	0	0	0	0	0.0000	0.0000	0.0000	0
8404	0	0	0	0	0	0	1.0000	0.0000	0.0000	0
9404	0	0	0	0	0	0	0.0000	-.3223	.9466	0
7405	0	0	0	0	0	0	0.0000	0.0000	0.0000	0
8405	0	0	0	0	0	0	1.0000	0.0000	0.0000	0
9405	0	0	0	0	0	0	0.0000	-.3223	.9466	0
7406	0	0	0	0	0	0	0.0000	0.0000	0.0000	0
8406	0	0	0	0	0	0	1.0000	0.0000	0.0000	0
9406	0	0	0	0	0	0	0.0000	-.3223	.9466	0
7407	0	0	0	0	0	0	0.0000	0.0000	0.0000	0
8407	0	0	0	0	0	0	1.0000	0.0000	0.0000	0
9407	0	0	0	0	0	0	0.0000	-.3223	.9466	0
7226	0	0	0	0	0	0	0.0000	0.0000	0.0000	0
8226	0	0	0	0	0	0	1.0000	0.0000	0.0000	0
9226	0	0	0	0	0	0	0.0000	-.3223	.9466	0
7427	0	0	0	0	0	0	0.0000	0.0000	0.0000	0
8427	0	0	0	0	0	0	1.0000	0.0000	0.0000	0
9427	0	0	0	0	0	0	0.0000	-.3223	.9466	0
7428	0	0	0	0	0	0	0.0000	0.0000	0.0000	0
8428	0	0	0	0	0	0	1.0000	0.0000	0.0000	0
9428	0	0	0	0	0	0	0.0000	-.3223	.9466	0
7429	0	0	0	0	0	0	0.0000	0.0000	0.0000	0
8429	0	0	0	0	0	0	1.0000	0.0000	0.0000	0
9429	0	0	0	0	0	0	0.0000	-.3223	.9466	0
7430	0	0	0	0	0	0	0.0000	0.0000	0.0000	0
8430	0	0	0	0	0	0	1.0000	0.0000	0.0000	0
9430	0	0	0	0	0	0	0.0000	-.3223	.9466	0
7431	0	0	0	0	0	0	0.0000	0.0000	0.0000	0
8431	0	0	0	0	0	0	1.0000	0.0000	0.0000	0
9431	0	0	0	0	0	0	0.0000	-.3223	.9466	0
7432	0	0	0	0	0	0	0.0000	0.0000	0.0000	0

8432	0	0	0	0	0	0	1.0000	0.0000	0.0000	0
9432	0	0	0	0	0	0	0.0000	-.3223	.9466	0
7251	0	0	0	0	0	0	0.0000	0.0000	0.0000	0
8251	0	0	0	0	0	0	1.0000	0.0000	0.0000	0
9251	0	0	0	0	0	0	0.0000	-.3223	.9466	0
7452	0	0	0	0	0	0	0.0000	0.0000	0.0000	0
8452	0	0	0	0	0	0	1.0000	0.0000	0.0000	0
9452	0	0	0	0	0	0	0.0000	-.3223	.9466	0
7453	0	0	0	0	0	0	0.0000	0.0000	0.0000	0
8453	0	0	0	0	0	0	1.0000	0.0000	0.0000	0
9453	0	0	0	0	0	0	0.0000	-.3223	.9466	0
7454	0	0	0	0	0	0	0.0000	0.0000	0.0000	0
8454	0	0	0	0	0	0	1.0000	0.0000	0.0000	0
9454	0	0	0	0	0	0	0.0000	-.3223	.9466	0
7455	0	0	0	0	0	0	0.0000	0.0000	0.0000	0
8455	0	0	0	0	0	0	1.0000	0.0000	0.0000	0
9455	0	0	0	0	0	0	0.0000	-.3223	.9466	0
7276	0	0	0	0	0	0	0.0000	0.0000	0.0000	0
8276	0	0	0	0	0	0	1.0000	0.0000	0.0000	0
9276	0	0	0	0	0	0	0.0000	-.3223	.9466	0
7477	0	0	0	0	0	0	0.0000	0.0000	0.0000	0
8477	0	0	0	0	0	0	1.0000	0.0000	0.0000	0
9477	0	0	0	0	0	0	0.0000	-.3223	.9466	0
7478	0	0	0	0	0	0	0.0000	0.0000	0.0000	0
8478	0	0	0	0	0	0	1.0000	0.0000	0.0000	0
9478	0	0	0	0	0	0	0.0000	-.3223	.9466	0
7479	0	0	0	0	0	0	0.0000	0.0000	0.0000	0
8479	0	0	0	0	0	0	1.0000	0.0000	0.0000	0
9479	0	0	0	0	0	0	0.0000	-.3223	.9466	0
7480	0	0	0	0	0	0	0.0000	0.0000	0.0000	0
8480	0	0	0	0	0	0	1.0000	0.0000	0.0000	0
9480	0	0	0	0	0	0	0.0000	-.3223	.9466	0
7301	0	0	0	0	0	0	0.0000	0.0000	0.0000	0
8301	0	0	0	0	0	0	1.0000	0.0000	0.0000	0
9301	0	0	0	0	0	0	0.0000	-.3223	.9466	0
7502	0	0	0	0	0	0	0.0000	0.0000	0.0000	0
8502	0	0	0	0	0	0	1.0000	0.0000	0.0000	0
9502	0	0	0	0	0	0	0.0000	-.3223	.9466	0
7326	0	0	0	0	0	0	0.0000	0.0000	0.0000	0
8326	0	0	0	0	0	0	1.0000	0.0000	0.0000	0
9326	0	0	0	0	0	0	0.0000	-.3223	.9466	0
7527	0	0	0	0	0	0	0.0000	0.0000	0.0000	0
8527	0	0	0	0	0	0	1.0000	0.0000	0.0000	0
9527	0	0	0	0	0	0	0.0000	-.3223	.9466	0
7351	0	0	0	0	0	0	0.0000	0.0000	0.0000	0
8351	0	0	0	0	0	0	1.0000	0.0000	0.0000	0
9351	0	0	0	0	0	0	0.0000	-.3223	.9466	0
7552	0	0	0	0	0	0	0.0000	0.0000	0.0000	0
8552	0	0	0	0	0	0	1.0000	0.0000	0.0000	0
9552	0	0	0	0	0	0	0.0000	-.3223	.9466	0
7201	0	0	0	0	0	0	0.0000	0.0000	0.0000	0
8201	0	0	0	0	0	0	1.0000	0.0000	0.0000	0
9201	0	0	0	0	0	0	0.0000	-.6101	.7923	0
7602	0	0	0	0	0	0	0.0000	0.0000	0.0000	0
8602	0	0	0	0	0	0	1.0000	0.0000	0.0000	0
9602	0	0	0	0	0	0	0.0000	-.6101	.7923	0
7603	0	0	0	0	0	0	0.0000	0.0000	0.0000	0
8603	0	0	0	0	0	0	1.0000	0.0000	0.0000	0
9603	0	0	0	0	0	0	0.0000	-.6101	.7923	0
7604	0	0	0	0	0	0	0.0000	0.0000	0.0000	0

8604	0	0	0	0	0	0	1.0000	0.0000	0.0000	0
9604	0	0	0	0	0	0	0.0000	-.6101	.7923	0
7605	0	0	0	0	0	0	0.0000	0.0000	0.0000	0
8605	0	0	0	0	0	0	1.0000	0.0000	0.0000	0
9605	0	0	0	0	0	0	0.0000	-.6101	.7923	0
7606	0	0	0	0	0	0	0.0000	0.0000	0.0000	0
8606	0	0	0	0	0	0	1.0000	0.0000	0.0000	0
9606	0	0	0	0	0	0	0.0000	-.6101	.7923	0
7607	0	0	0	0	0	0	0.0000	0.0000	0.0000	0
8607	0	0	0	0	0	0	1.0000	0.0000	0.0000	0
9607	0	0	0	0	0	0	0.0000	-.6101	.7923	0
7226	0	0	0	0	0	0	0.0000	0.0000	0.0000	0
8226	0	0	0	0	0	0	1.0000	0.0000	0.0000	0
9226	0	0	0	0	0	0	0.0000	-.6101	.7923	0
7627	0	0	0	0	0	0	0.0000	0.0000	0.0000	0
8627	0	0	0	0	0	0	1.0000	0.0000	0.0000	0
9627	0	0	0	0	0	0	0.0000	-.6101	.7923	0
7628	0	0	0	0	0	0	0.0000	0.0000	0.0000	0
8628	0	0	0	0	0	0	1.0000	0.0000	0.0000	0
9628	0	0	0	0	0	0	0.0000	-.6101	.7923	0
7629	0	0	0	0	0	0	0.0000	0.0000	0.0000	0
8629	0	0	0	0	0	0	1.0000	0.0000	0.0000	0
9629	0	0	0	0	0	0	0.0000	-.6101	.7923	0
7630	0	0	0	0	0	0	0.0000	0.0000	0.0000	0
8630	0	0	0	0	0	0	1.0000	0.0000	0.0000	0
9630	0	0	0	0	0	0	0.0000	-.6101	.7923	0
7631	0	0	0	0	0	0	0.0000	0.0000	0.0000	0
8631	0	0	0	0	0	0	1.0000	0.0000	0.0000	0
9631	0	0	0	0	0	0	0.0000	-.6101	.7923	0
7632	0	0	0	0	0	0	0.0000	0.0000	0.0000	0
8632	0	0	0	0	0	0	1.0000	0.0000	0.0000	0
9632	0	0	0	0	0	0	0.0000	-.6101	.7923	0
7251	0	0	0	0	0	0	0.0000	0.0000	0.0000	0
8251	0	0	0	0	0	0	1.0000	0.0000	0.0000	0
9251	0	0	0	0	0	0	0.0000	-.6101	.7923	0
7652	0	0	0	0	0	0	0.0000	0.0000	0.0000	0
8652	0	0	0	0	0	0	1.0000	0.0000	0.0000	0
9652	0	0	0	0	0	0	0.0000	-.6101	.7923	0
7653	0	0	0	0	0	0	0.0000	0.0000	0.0000	0
8653	0	0	0	0	0	0	1.0000	0.0000	0.0000	0
9653	0	0	0	0	0	0	0.0000	-.6101	.7923	0
7654	0	0	0	0	0	0	0.0000	0.0000	0.0000	0
8654	0	0	0	0	0	0	1.0000	0.0000	0.0000	0
9654	0	0	0	0	0	0	0.0000	-.6101	.7923	0
7655	0	0	0	0	0	0	0.0000	0.0000	0.0000	0
8655	0	0	0	0	0	0	1.0000	0.0000	0.0000	0
9655	0	0	0	0	0	0	0.0000	-.6101	.7923	0
7276	0	0	0	0	0	0	0.0000	0.0000	0.0000	0
8276	0	0	0	0	0	0	1.0000	0.0000	0.0000	0
9276	0	0	0	0	0	0	0.0000	-.6101	.7923	0
7677	0	0	0	0	0	0	0.0000	0.0000	0.0000	0
8677	0	0	0	0	0	0	1.0000	0.0000	0.0000	0
9677	0	0	0	0	0	0	0.0000	-.6101	.7923	0
7678	0	0	0	0	0	0	0.0000	0.0000	0.0000	0
8678	0	0	0	0	0	0	1.0000	0.0000	0.0000	0
9678	0	0	0	0	0	0	0.0000	-.6101	.7923	0
7679	0	0	0	0	0	0	0.0000	0.0000	0.0000	0
8679	0	0	0	0	0	0	1.0000	0.0000	0.0000	0
9679	0	0	0	0	0	0	0.0000	-.6101	.7923	0
7680	0	0	0	0	0	0	0.0000	0.0000	0.0000	0

8680	0	0	0	0	0	0	1.0000	0.0000	0.0000	0
9680	0	0	0	0	0	0	0.0000	-.6101	.7923	0
7301	0	0	0	0	0	0	0.0000	0.0000	0.0000	0
8301	0	0	0	0	0	0	1.0000	0.0000	0.0000	0
9301	0	0	0	0	0	0	0.0000	-.6101	.7923	0
7702	0	0	0	0	0	0	0.0000	0.0000	0.0000	0
8702	0	0	0	0	0	0	1.0000	0.0000	0.0000	0
9702	0	0	0	0	0	0	0.0000	-.6101	.7923	0
7326	0	0	0	0	0	0	0.0000	0.0000	0.0000	0
8326	0	0	0	0	0	0	1.0000	0.0000	0.0000	0
9326	0	0	0	0	0	0	0.0000	-.6101	.7923	0
7727	0	0	0	0	0	0	0.0000	0.0000	0.0000	0
8727	0	0	0	0	0	0	1.0000	0.0000	0.0000	0
9727	0	0	0	0	0	0	0.0000	-.6101	.7923	0
7351	0	0	0	0	0	0	0.0000	0.0000	0.0000	0
8351	0	0	0	0	0	0	1.0000	0.0000	0.0000	0
9351	0	0	0	0	0	0	0.0000	-.6101	.7923	0
7752	0	0	0	0	0	0	0.0000	0.0000	0.0000	0
8752	0	0	0	0	0	0	1.0000	0.0000	0.0000	0
9752	0	0	0	0	0	0	0.0000	-.6101	.7923	0
7201	0	0	0	0	0	0	0.0000	0.0000	0.0000	0
8201	0	0	0	0	0	0	1.0000	0.0000	0.0000	0
9201	0	0	0	0	0	0	0.0000	-.8329	.5534	0
7802	0	0	0	0	0	0	0.0000	0.0000	0.0000	0
8802	0	0	0	0	0	0	1.0000	0.0000	0.0000	0
9802	0	0	0	0	0	0	0.0000	-.8329	.5534	0
7803	0	0	0	0	0	0	0.0000	0.0000	0.0000	0
8803	0	0	0	0	0	0	1.0000	0.0000	0.0000	0
9803	0	0	0	0	0	0	0.0000	-.8329	.5534	0
7804	0	0	0	0	0	0	0.0000	0.0000	0.0000	0
8804	0	0	0	0	0	0	1.0000	0.0000	0.0000	0
9804	0	0	0	0	0	0	0.0000	-.8329	.5534	0
7805	0	0	0	0	0	0	0.0000	0.0000	0.0000	0
8805	0	0	0	0	0	0	1.0000	0.0000	0.0000	0
9805	0	0	0	0	0	0	0.0000	-.8329	.5534	0
7806	0	0	0	0	0	0	0.0000	0.0000	0.0000	0
8806	0	0	0	0	0	0	1.0000	0.0000	0.0000	0
9806	0	0	0	0	0	0	0.0000	-.8329	.5534	0
7807	0	0	0	0	0	0	0.0000	0.0000	0.0000	0
8807	0	0	0	0	0	0	1.0000	0.0000	0.0000	0
9807	0	0	0	0	0	0	0.0000	-.8329	.5534	0
7226	0	0	0	0	0	0	0.0000	0.0000	0.0000	0
8226	0	0	0	0	0	0	1.0000	0.0000	0.0000	0
9226	0	0	0	0	0	0	0.0000	-.8329	.5534	0
7827	0	0	0	0	0	0	0.0000	0.0000	0.0000	0
8827	0	0	0	0	0	0	1.0000	0.0000	0.0000	0
9827	0	0	0	0	0	0	0.0000	-.8329	.5534	0
7828	0	0	0	0	0	0	0.0000	0.0000	0.0000	0
8828	0	0	0	0	0	0	1.0000	0.0000	0.0000	0
9828	0	0	0	0	0	0	0.0000	-.8329	.5534	0
7829	0	0	0	0	0	0	0.0000	0.0000	0.0000	0
8829	0	0	0	0	0	0	1.0000	0.0000	0.0000	0
9829	0	0	0	0	0	0	0.0000	-.8329	.5534	0
7830	0	0	0	0	0	0	0.0000	0.0000	0.0000	0
8830	0	0	0	0	0	0	1.0000	0.0000	0.0000	0
9830	0	0	0	0	0	0	0.0000	-.8329	.5534	0
7831	0	0	0	0	0	0	0.0000	0.0000	0.0000	0
8831	0	0	0	0	0	0	1.0000	0.0000	0.0000	0
9831	0	0	0	0	0	0	0.0000	-.8329	.5534	0
7832	0	0	0	0	0	0	0.0000	0.0000	0.0000	0

```
8832    0    0    0    0    0    0    1.0000    0.0000    0.0000    0
9832    0    0    0    0    0    0    0.0000    -.8329     .5534    0
7251    0    0    0    0    0    0    0.0000    0.0000    0.0000    0
8251    0    0    0    0    0    0    1.0000    0.0000    0.0000    0
9251    0    0    0    0    0    0    0.0000    -.8329     .5534    0
7852    0    0    0    0    0    0    0.0000    0.0000    0.0000    0
8852    0    0    0    0    0    0    1.0000    0.0000    0.0000    0
9852    0    0    0    0    0    0    0.0000    -.8329     .5534    0
7853    0    0    0    0    0    0    0.0000    0.0000    0.0000    0
8853    0    0    0    0    0    0    1.0000    0.0000    0.0000    0
9853    0    0    0    0    0    0    0.0000    -.8329     .5534    0
7854    0    0    0    0    0    0    0.0000    0.0000    0.0000    0
8854    0    0    0    0    0    0    1.0000    0.0000    0.0000    0
9854    0    0    0    0    0    0    0.0000    -.8329     .5534    0
7855    0    0    0    0    0    0    0.0000    0.0000    0.0000    0
8855    0    0    0    0    0    0    1.0000    0.0000    0.0000    0
9855    0    0    0    0    0    0    0.0000    -.8329     .5534    0
7856    0    0    0    0    0    0    0.0000    0.0000    0.0000    0
8856    0    0    0    0    0    0    1.0000    0.0000    0.0000    0
9856    0    0    0    0    0    0    0.0000    -.8329     .5534    0
7276    0    0    0    0    0    0    0.0000    0.0000    0.0000    0
8276    0    0    0    0    0    0    1.0000    0.0000    0.0000    0
9276    0    0    0    0    0    0    0.0000    -.8329     .5534    0
7877    0    0    0    0    0    0    0.0000    0.0000    0.0000    0
8877    0    0    0    0    0    0    1.0000    0.0000    0.0000    0
9877    0    0    0    0    0    0    0.0000    -.8329     .5534    0
7878    0    0    0    0    0    0    0.0000    0.0000    0.0000    0
8878    0    0    0    0    0    0    1.0000    0.0000    0.0000    0
9878    0    0    0    0    0    0    0.0000    -.8329     .5534    0
7879    0    0    0    0    0    0    0.0000    0.0000    0.0000    0
8879    0    0    0    0    0    0    1.0000    0.0000    0.0000    0
9879    0    0    0    0    0    0    0.0000    -.8329     .5534    0
7880    0    0    0    0    0    0    0.0000    0.0000    0.0000    0
8880    0    0    0    0    0    0    1.0000    0.0000    0.0000    0
9880    0    0    0    0    0    0    0.0000    -.8329     .5534    0
7881    0    0    0    0    0    0    0.0000    0.0000    0.0000    0
8881    0    0    0    0    0    0    1.0000    0.0000    0.0000    0
9881    0    0    0    0    0    0    0.0000    -.8329     .5534    0
7301    0    0    0    0    0    0    0.0000    0.0000    0.0000    0
8301    0    0    0    0    0    0    1.0000    0.0000    0.0000    0
9301    0    0    0    0    0    0    0.0000    -.8329     .5534    0
7902    0    0    0    0    0    0    0.0000    0.0000    0.0000    0
8902    0    0    0    0    0    0    1.0000    0.0000    0.0000    0
9902    0    0    0    0    0    0    0.0000    -.8329     .5534    0
7326    0    0    0    0    0    0    0.0000    0.0000    0.0000    0
8326    0    0    0    0    0    0    1.0000    0.0000    0.0000    0
9326    0    0    0    0    0    0    0.0000    -.8329     .5534    0
7927    0    0    0    0    0    0    0.0000    0.0000    0.0000    0
8927    0    0    0    0    0    0    1.0000    0.0000    0.0000    0
9927    0    0    0    0    0    0    0.0000    -.8329     .5534    0
7351    0    0    0    0    0    0    0.0000    0.0000    0.0000    0
8351    0    0    0    0    0    0    1.0000    0.0000    0.0000    0
9351    0    0    0    0    0    0    0.0000    -.8329     .5534    0
7952    0    0    0    0    0    0    0.0000    0.0000    0.0000    0
8952    0    0    0    0    0    0    1.0000    0.0000    0.0000    0
9952    0    0    0    0    0    0    0.0000    -.8329     .5534    0
  3 122    0    0          8                        0      2   3    0 122 ADECC
0
    1    .10000
10000000. 10000000. 10000000.      .320      .320      .320 3790000.  ADMPC
                                                                       379
```

```
0000.
       3790000.
         2    .67000
      20000000. 20000000. 20000000.       .290       .290       .290  7750000.  775
0000.
       7750000.
         3    .10000
      10000000. 10000000. 10000000.       .320       .320       .320  3790000.  379
0000.
       3790000.
         1 7201 8201 9201                                                        ADMAX
         2 7202 8202 9202
         3 7203 8203 9203
         4 7204 8204 9204
         5 7205 8205 9205
         6 7206 8206 9206
         7 7207 8207 9207
         8 7226 8226 9226
         9 7227 8227 9227
        10 7228 8228 9228
        11 7229 8229 9229
        12 7230 8230 9230
        13 7231 8231 9231
        14 7232 8232 9232
        15 7251 8251 9251
        16 7252 8252 9252
        17 7253 8253 9253
        18 7254 8254 9254
        19 7255 8255 9255
        20 7276 8276 9276
        21 7277 8277 9277
        22 7278 8278 9278
        23 7279 8279 9279
        24 7280 8280 9280
        25 7301 8301 9301
        26 7302 8302 9302
        27 7326 8326 9326
        28 7327 8327 9327
        29 7351 8351 9351
        30 7352 8352 9352
        31 7201 8201 9201
        32 7402 8402 9402
        33 7403 8403 9403
        34 7404 8404 9404
        35 7405 8405 9405
        36 7406 8406 9406
        37 7407 8407 9407
        38 7226 8226 9226
        39 7427 8427 9427
        40 7428 8428 9428
        41 7429 8429 9429
        42 7430 8430 9430
        43 7431 8431 9431
        44 7432 8432 9432
        45 7251 8251 9251
        46 7452 8452 9452
        47 7453 8453 9453
        48 7454 8454 9454
        49 7455 8455 9455
        50 7276 8276 9276
```

51	7477	8477	9477
52	7478	8478	9478
53	7479	8479	9479
54	7480	8480	9480
55	7301	8301	9301
56	7502	8502	9502
57	7326	8326	9326
58	7527	8527	9527
59	7351	8351	9351
60	7552	8552	9552
61	7201	8201	9201
62	7602	8602	9602
63	7603	8603	9603
64	7604	8604	9604
65	7605	8605	9605
66	7606	8606	9606
67	7607	8607	9607
68	7226	8226	9226
69	7627	8627	9627
70	7628	8628	9628
71	7629	8629	9629
72	7630	8630	9630
73	7631	8631	9631
74	7632	8632	9632
75	7251	8251	9251
76	7652	8652	9652
77	7653	8653	9653
78	7654	8654	9654
79	7655	8655	9655
80	7276	8276	9276
81	7677	8677	9677
82	7678	8678	9678
83	7679	8679	9679
84	7680	8680	9680
85	7301	8301	9301
86	7702	8702	9702
87	7326	8326	9326
88	7727	8727	9727
89	7351	8351	9351
90	7752	8752	9752
91	7201	8201	9201
92	7802	8802	9802
93	7803	8803	9803
94	7804	8804	9804
95	7805	8805	9805
96	7806	8806	9806
97	7807	8807	9807
98	7226	8226	9226
99	7827	8827	9827
100	7828	8828	9828
101	7829	8829	9829
102	7830	8830	9830
103	7831	8831	9831
104	7832	8832	9832
105	7251	8251	9251
106	7852	8852	9852
107	7853	8853	9853
108	7854	8854	9854
109	7855	8855	9855
110	7856	8856	9856

```
111 7276 8276 9276
112 7877 8877 9877
113 7878 8878 9878
114 7879 8879 9879
115 7880 8880 9880
116 7881 8881 9881
117 7301 8301 9301
118 7902 8902 9902
119 7326 8326 9326
120 7927 8927 9927
121 7351 8351 9351
122 7952 8952 9952
  1    8    8    0    2    1    0    0
201  202  402  401  226  227  427  426
  2    8    8    0    2    2    0    0
202  203  403  402  227  228  428  427
  3    8    8    0    1    3    0    0
203  204  404  403  228  229  429  428
  4    8    8    0    1    4    0    0
204  205  405  404  229  230  430  429
  5    8    8    0    1    5    0    0
205  206  406  405  230  231  431  430
  6    8    8    0    1    6    0    0
206  207  407  406  231  232  432  431
  7    8    8    0    1    7    0    0
207  208  408  407  232  233  433  432
  8    8    8    0    2    8    0    0
226  227  427  426  251  252  452  451
  9    8    8    0    2    9    0    0
227  228  428  427  252  253  453  452
 10    8    8    0    1   10    0    0
228  229  429  428  253  254  454  453
 11    8    8    0    1   11    0    0
229  230  430  429  254  255  455  454
 12    8    8    0    1   12    0    0
230  231  431  430  255  256  456  455
 13    8    8    0    1   13    0    0
231  232  432  431  256  257  457  456
 14    8    8    0    1   14    0    0
232  233  433  432  257  258  458  457
 15    8    8    0    2   15    0    0
251  252  452  451  276  277  477  476
 16    8    8    0    2   16    0    0
252  253  453  452  277  278  478  477
 17    8    8    0    1   17    0    0
253  254  454  453  278  279  479  478
 18    8    8    0    1   18    0    0
254  255  455  454  279  280  480  479
 19    8    8    0    1   19    0    0
255  256  456  455  280  281  481  480
 20    8    8    0    2   20    0    0
276  277  477  476  301  302  502  501
 21    8    8    0    2   21    0    0
277  278  478  477  302  303  503  502
 22    8    8    0    1   22    0    0
278  279  479  478  303  304  504  503
 23    8    8    0    1   23    0    0
279  280  480  479  304  305  505  504
 24    8    8    0    1   24    0    0
280  281  481  480  305  306  506  505
```

ADELD

25	8	8	0	2	25	0	0
301	302	502	501	326	327	527	526
26	8	8	0	2	26	0	0
302	303	503	502	327	328	528	527
27	8	8	0	2	27	0	0
326	327	527	526	351	352	552	551
28	8	8	0	2	28	0	0
327	328	528	527	352	353	553	552
29	8	8	0	1	29	0	0
351	352	552	551	376	377	577	576
30	8	8	0	1	30	0	0
352	353	553	552	377	378	578	577
31	8	8	0	2	31	0	0
201	402	602	601	226	427	627	626
32	8	8	0	2	32	0	0
402	403	603	602	427	428	628	627
33	8	8	0	1	33	0	0
403	404	604	603	428	429	629	628
34	8	8	0	1	34	0	0
404	405	605	604	429	430	630	629
35	8	8	0	1	35	0	0
405	406	606	605	430	431	631	630
36	8	8	0	1	36	0	0
406	407	607	606	431	432	632	631
37	8	8	0	1	37	0	0
407	408	608	607	432	433	633	632
38	8	8	0	2	38	0	0
226	427	627	626	251	452	652	651
39	8	8	0	2	39	0	0
427	428	628	627	452	453	653	652
40	8	8	0	1	40	0	0
428	429	629	628	453	454	654	653
41	8	8	0	1	41	0	0
429	430	630	629	454	455	655	654
42	8	8	0	1	42	0	0
430	431	631	630	455	456	656	655
43	8	8	0	1	43	0	0
431	432	632	631	456	457	657	656
44	8	8	0	1	44	0	0
432	433	633	632	457	458	658	657
45	8	8	0	2	45	0	0
251	452	652	651	276	477	677	676
46	8	8	0	2	46	0	0
452	453	653	652	477	478	678	677
47	8	8	0	1	47	0	0
453	454	654	653	478	479	679	678
48	8	8	0	1	48	0	0
454	455	655	654	479	480	680	679
49	8	8	0	1	49	0	0
455	456	656	655	480	481	681	680
50	8	8	0	2	50	0	0
276	477	677	676	301	502	702	701
51	8	8	0	2	51	0	0
477	478	678	677	502	503	703	702
52	8	8	0	1	52	0	0
478	479	679	678	503	504	704	703
53	8	8	0	1	53	0	0
479	480	680	679	504	505	705	704
54	8	8	0	1	54	0	0
480	481	681	680	505	506	706	705

55	8	8	0	2	55	0	0
301	502	702	701	326	527	727	726
56	8	8	0	2	56	0	0
502	503	703	702	527	528	728	727
57	8	8	0	2	57	0	0
326	527	727	726	351	552	752	751
58	8	8	0	2	58	0	0
527	528	728	727	552	553	753	752
59	8	8	0	1	59	0	0
351	552	752	751	376	577	777	776
60	8	8	0	1	60	0	0
552	553	753	752	577	578	778	777
61	8	8	0	2	61	0	0
201	602	802	801	226	627	827	826
62	8	8	0	2	62	0	0
602	603	803	802	627	628	828	827
63	8	8	0	1	63	0	0
603	604	804	803	628	629	829	828
64	8	8	0	1	64	0	0
604	605	805	804	629	630	830	829
65	8	8	0	1	65	0	0
605	606	806	805	630	631	831	830
66	8	8	0	1	66	0	0
606	607	807	806	631	632	832	831
67	8	8	0	1	67	0	0
607	608	808	807	632	633	833	832
68	8	8	0	2	68	0	0
226	627	827	826	251	652	852	851
69	8	8	0	2	69	0	0
627	628	828	827	652	653	853	852
70	8	8	0	1	70	0	0
628	629	829	828	653	654	854	853
71	8	8	0	1	71	0	0
629	630	830	829	654	655	855	854
72	8	8	0	1	72	0	0
630	631	831	830	655	656	856	855
73	8	8	0	1	73	0	0
631	632	832	831	656	657	857	856
74	8	8	0	1	74	0	0
632	633	833	832	657	658	858	857
75	8	8	0	2	75	0	0
251	652	852	851	276	677	877	876
76	8	8	0	2	76	0	0
652	653	853	852	677	678	878	877
77	8	8	0	1	77	0	0
653	654	854	853	678	679	879	878
78	8	8	0	1	78	0	0
654	655	855	854	679	680	880	879
79	8	8	0	1	79	0	0
655	656	856	855	680	681	881	880
80	8	8	0	2	80	0	0
276	677	877	876	301	702	902	901
81	8	8	0	2	81	0	0
677	678	878	877	702	703	903	902
82	8	8	0	1	82	0	0
678	679	879	878	703	704	904	903
83	8	8	0	1	83	0	0
679	680	880	879	704	705	905	904
84	8	8	0	1	84	0	0
680	681	881	880	705	706	906	905

85	8	8	0	2	85	0	0
301	702	902	901	326	727	927	926
86	8	8	0	2	86	0	0
702	703	903	902	727	728	928	927
87	8	8	0	2	87	0	0
326	727	927	926	351	752	952	951
88	8	8	0	2	88	0	0
727	728	928	927	752	753	953	952
89	8	8	0	1	89	0	0
351	752	952	951	376	777	977	976
90	8	8	0	1	90	0	0
752	753	953	952	777	778	978	977
91	8	8	0	2	91	0	0
201	802	1002	1001	226	827	1027	1026
92	8	8	0	2	92	0	0
802	803	1003	1002	827	828	1028	1027
93	8	8	0	1	93	0	0
803	804	1004	1003	828	829	1029	1028
94	8	8	0	1	94	0	0
804	805	1005	1004	829	830	1030	1029
95	8	8	0	1	95	0	0
805	806	1006	1005	830	831	1031	1030
96	8	8	0	1	96	0	0
806	807	1007	1006	831	832	1032	1031
97	8	8	0	1	97	0	0
807	808	1008	1007	832	833	1033	1032
98	8	8	0	2	98	0	0
226	827	1027	1026	251	852	1052	1051
99	8	8	0	2	99	0	0
827	828	1028	1027	852	853	1053	1052
100	8	8	0	1	100	0	0
828	829	1029	1028	853	854	1054	1053
101	8	8	0	1	101	0	0
829	830	1030	1029	854	855	1055	1054
102	8	8	0	1	102	0	0
830	831	1031	1030	855	856	1056	1055
103	8	8	0	1	103	0	0
831	832	1032	1031	856	857	1057	1056
104	8	8	0	1	104	0	0
832	833	1033	1032	857	858	1058	1057
105	8	8	0	2	105	0	0
251	852	1052	1051	276	877	1077	1076
106	8	8	0	2	106	0	0
852	853	1053	1052	877	878	1078	1077
107	8	8	0	1	107	0	0
853	854	1054	1053	878	879	1079	1078
108	8	8	0	1	108	0	0
854	855	1055	1054	879	880	1080	1079
109	8	8	0	1	109	0	0
855	856	1056	1055	880	881	1081	1080
110	8	8	0	3	110	0	0
856	857	1057	1056	881	882	1082	1081
111	8	8	0	2	111	0	0
276	877	1077	1076	301	902	1102	1101
112	8	8	0	2	112	0	0
877	878	1078	1077	902	903	1103	1102
113	8	8	0	1	113	0	0
878	879	1079	1078	903	904	1104	1103
114	8	8	0	1	114	0	0
879	880	1080	1079	904	905	1105	1104

```
115    8    8    0    1  115    0    0
880  881 1081 1080  905  906 1106 1105
116    8    8    0    3  116    0    0
881  882 1082 1081  906  907 1107 1106
117    8    8    0    2  117    0    0
301  902 1102 1101  326  927 1127 1126
118    8    8    0    2  118    0    0
902  903 1103 1102  927  928 1128 1127
119    8    8    0    2  119    0    0
326  927 1127 1126  351  952 1152 1151
120    8    8    0    2  120    0    0
927  928 1128 1127  952  953 1153 1152
121    8    8    0    1  121    0    0
351  952 1152 1151  376  977 1177 1176
122    8    8    0    1  122    0    0
952  953 1153 1152  977  978 1178 1177
```

APPENDIX D

Listing of File NAS,

the Input Cards for NASTRAN Program

```
GRID        2003        1  1.2500   0.0000   0.0000        1     2456
GRID        4003        1  1.2500  12.0000   0.0000        1      456
GRID        2004        1  1.5000   0.0000   0.0000        1     2456
GRID        4004        1  1.5000  12.0000   0.0000        1      456
GRID        2005        1  1.7500   0.0000   0.0000        1     2456
GRID        4005        1  1.7500  12.0000   0.0000        1      456
GRID        2006        1  2.0000   0.0000   0.0000        1     2456
GRID        4006        1  2.0000  12.0000   0.0000        1      456
GRID        2028        1  1.2500   0.0000    .5000        1     2456
GRID        4028        1  1.2500  12.0000    .5000        1      456
GRID        2029        1  1.5000   0.0000    .5000        1     2456
GRID        4029        1  1.5000  12.0000    .5000        1      456
GRID        2030        1  1.7500   0.0000    .5000        1     2456
GRID        4030        1  1.7500  12.0000    .5000        1      456
GRID        2031        1  2.0000   0.0000    .5000        1     2456
GRID        4031        1  2.0000  12.0000    .5000        1      456
GRID        2053        1  1.2500   0.0000   1.0000        1     2456
GRID        4053        1  1.2500  12.0000   1.0000        1      456
GRID        2054        1  1.5000   0.0000   1.0000        1     2456
GRID        4054        1  1.5000  12.0000   1.0000        1      456
GRID        2055        1  1.7500   0.0000   1.0000        1     2456
GRID        4055        1  1.7500  12.0000   1.0000        1      456
GRID        2056        1  2.0000   0.0000   1.0000        1     2456
GRID        4056        1  2.0000  12.0000   1.0000        1      456
GRID        2077        1  1.0000   0.0000   1.5000        1     2456
GRID        4077        1  1.0000  12.0000   1.5000        1      456
GRID        2078        1  1.2500   0.0000   1.5000        1     2456
GRID        4078        1  1.2500  12.0000   1.5000        1      456
GRID        2079        1  1.5000   0.0000   1.5000        1     2456
GRID        4079        1  1.5000  12.0000   1.5000        1      456
GRID        2080        1  1.7500   0.0000   1.5000        1     2456
GRID        4080        1  1.7500  12.0000   1.5000        1      456
GRID        2081        1  2.0000   0.0000   1.5000        1     2456
GRID        4081        1  2.0000  12.0000   1.5000        1      456
GRID        2102        1  1.0000   0.0000   2.0000        1     2456
GRID        4102        1  1.0000  12.0000   2.0000        1      456
GRID        2103        1  1.2500   0.0000   2.0000        1     2456
GRID        4103        1  1.2500  12.0000   2.0000        1      456
GRID        2104        1  1.5000   0.0000   2.0000        1     2456
GRID        4104        1  1.5000  12.0000   2.0000        1      456
GRID        2105        1  1.7500   0.0000   2.0000        1     2456
GRID        4105        1  1.7500  12.0000   2.0000        1      456
GRID        2106        1  2.0000   0.0000   2.0000        1     2456
GRID        4106        1  2.0000  12.0000   2.0000        1      456
GRID        2127        1  1.0000   0.0000   2.5000        1     2456
GRID        4127        1  1.0000  12.0000   2.5000        1      456
GRID        2128        1  1.2500   0.0000   2.5000        1     2456
GRID        4128        1  1.2500  12.0000   2.5000        1      456
GRID        2129        1  1.5000   0.0000   2.5000        1     2456
GRID        4129        1  1.5000  12.0000   2.5000        1      456
GRID        2130        1  1.7500   0.0000   2.5000        1     2456
GRID        4130        1  1.7500  12.0000   2.5000        1      456
GRID        2131        1  2.0000   0.0000   2.5000        1     2456
GRID        4131        1  2.0000  12.0000   2.5000        1      456
GRID        2152        1  1.0000   0.0000   3.0000        1     2456
GRID        4152        1  1.0000  12.0000   3.0000        1      456
GRID        2153        1  1.2500   0.0000   3.0000        1     2456
GRID        4153        1  1.2500  12.0000   3.0000        1      456
GRID        2154        1  1.5000   0.0000   3.0000        1     2456
```

```
GRID           4154          1  1.5000 12.0000  3.0000        1      456
GRID           2155          1  1.7500  0.0000  3.0000        1     2456
GRID           4155          1  1.7500 12.0000  3.0000        1      456
GRID           2156          1  2.0000  0.0000  3.0000        1     2456
GRID           4156          1  2.0000 12.0000  3.0000        1      456
GRID           2177          1  1.0000  0.0000  3.5000        1     2456
GRID           4177          1  1.0000 12.0000  3.5000        1      456
GRID           2178          1  1.2500  0.0000  3.5000        1     2456
GRID           4178          1  1.2500 12.0000  3.5000        1      456
GRID           2179          1  1.5000  0.0000  3.5000        1     2456
GRID           4179          1  1.5000 12.0000  3.5000        1      456
GRID           2180          1  1.7500  0.0000  3.5000        1     2456
GRID           4180          1  1.7500 12.0000  3.5000        1      456
GRID           2181          1  2.0000  0.0000  3.5000        1     2456
GRID           4181          1  2.0000 12.0000  3.5000        1      456
GRID           2202          1  1.0000  0.0000  4.0000        1    23456
GRID           4202          1  1.0000 12.0000  4.0000        1     3456
GRID           2203          1  1.2500  0.0000  4.0000        1     2456
GRID           4203          1  1.2500 12.0000  4.0000        1      456
GRID           2204          1  1.5000  0.0000  4.0000        1     2456
GRID           4204          1  1.5000 12.0000  4.0000        1      456
GRID           2205          1  1.7500  0.0000  4.0000        1     2456
GRID           4205          1  1.7500 12.0000  4.0000        1      456
GRID           2206          1  2.0000  0.0000  4.0000        1     2456
GRID           4206          1  2.0000 12.0000  4.0000        1      456
CHEXA          2001          1     2003     2004     4004     4003     2028     2029CBC
2001
+BC 2001       4029       4028
CHEXA          2002          1     2004     2005     4005     4004     2029     2030CBC
2002
+BC 2002       4030       4029
CHEXA          2003          1     2005     2006     4006     4005     2030     2031CBC
2003
+BC 2003       4031       4030
CHEXA          2004          1     2028     2029     4029     4028     2053     2054CBC
2004
+BC 2004       4054       4053
CHEXA          2005          1     2029     2030     4030     4029     2054     2055CBC
2005
+BC 2005       4055       4054
CHEXA          2006          1     2030     2031     4031     4030     2055     2056CBC
2006
+BC 2006       4056       4055
CHEXA          2007          1     2053     2054     4054     4053     2078     2079CBC
2007
+BC 2007       4079       4078
CHEXA          2008          1     2054     2055     4055     4054     2079     2080CBC
2008
+BC 2008       4080       4079
CHEXA          2009          1     2055     2056     4056     4055     2080     2081CBC
2009
+BC 2009       4081       4080
CHEXA          2010          1     2077     2078     4078     4077     2102     2103CBC
2010
+BC 2010       4103       4102
CHEXA          2011          1     2078     2079     4079     4078     2103     2104CBC
2011
+BC 2011       4104       4103
CHEXA          2012          1     2079     2080     4080     4079     2104     2105CBC
2012
```

```
    +BC 2012    4105    4104
    CHEXA       2013       1    2080    2081    4081    4080    2105    2106CBC
2013
    +BC 2013    4106    4105
    CHEXA       2014       1    2102    2103    4103    4102    2127    2128CBC
2014
    +BC 2014    4128    4127
    CHEXA       2015       1    2103    2104    4104    4103    2128    2129CBC
2015
    +BC 2015    4129    4128
    CHEXA       2016       1    2104    2105    4105    4104    2129    2130CBC
2016
    +BC 2016    4130    4129
    CHEXA       2017       1    2105    2106    4106    4105    2130    2131CBC
2017
    +BC 2017    4131    4130
    CHEXA       2018       1    2127    2128    4128    4127    2152    2153CBC
2018
    +BC 2018    4153    4152
    CHEXA       2019       1    2128    2129    4129    4128    2153    2154CBC
2019
    +BC 2019    4154    4153
    CHEXA       2020       1    2129    2130    4130    4129    2154    2155CBC
2020
    +BC 2020    4155    4154
    CHEXA       2021       1    2130    2131    4131    4130    2155    2156CBC
2021
    +BC 2021    4156    4155
    CHEXA       2022       1    2152    2153    4153    4152    2177    2178CBC
2022
    +BC 2022    4178    4177
    CHEXA       2023       1    2153    2154    4154    4153    2178    2179CBC
2023
    +BC 2023    4179    4178
    CHEXA       2024       1    2154    2155    4155    4154    2179    2180CBC
2024
    +BC 2024    4180    4179
    CHEXA       2025       1    2155    2156    4156    4155    2180    2181CBC
2025
    +BC 2025    4181    4180
    CHEXA       2026       1    2177    2178    4178    4177    2202    2203CBC
2026
    +BC 2026    4203    4202
    CHEXA       2027       1    2178    2179    4179    4178    2203    2204CBC
2027
    +BC 2027    4204    4203
    CHEXA       2028       1    2179    2180    4180    4179    2204    2205CBC
2028
    +BC 2028    4205    4204
    CHEXA       2029       1    2180    2181    4181    4180    2205    2206CBC
2029
    +BC 2029    4206    4205
    PLOAD       130000.00    2077    2078    4078    4077
    PLOAD       130000.00    2078    2053    4053    4078
    PLOAD       130000.00    2053    2028    4028    4053
    PLOAD       130000.00    2028    2003    4003    4028
    PLOAD       130000.00    2003    2004    4004    4003
    PLOAD       130000.00    2004    2005    4005    4004
    PLOAD       130000.00    2005    2006    4006    4005
    PLOAD       130000.00    2006    2031    4031    4006
```

```
PLOAD          130000.00     2031     2056     4056     4031
PLOAD          130000.00     2056     2081     4081     4056
GRID    10003          1  1.2500 78.0000   0.0000        1      456
GRID    12003          1  1.2500 90.0000   0.0000        1     2456
GRID    10004          1  1.5000 78.0000   0.0000        1      456
GRID    12004          1  1.5000 90.0000   0.0000        1     2456
GRID    10005          1  1.7500 78.0000   0.0000        1      456
GRID    12005          1  1.7500 90.0000   0.0000        1     2456
GRID    10006          1  2.0000 78.0000   0.0000        1      456
GRID    12006          1  2.0000 90.0000   0.0000        1     2456
GRID    10028          1  1.2500 78.0000    .5000        1      456
GRID    12028          1  1.2500 90.0000    .5000        1     2456
GRID    10029          1  1.5000 78.0000    .5000        1      456
GRID    12029          1  1.5000 90.0000    .5000        1     2456
GRID    10030          1  1.7500 78.0000    .5000        1      456
GRID    12030          1  1.7500 90.0000    .5000        1     2456
GRID    10031          1  2.0000 78.0000    .5000        1      456
GRID    12031          1  2.0000 90.0000    .5000        1     2456
GRID    10053          1  1.2500 78.0000   1.0000        1      456
GRID    12053          1  1.2500 90.0000   1.0000        1     2456
GRID    10054          1  1.5000 78.0000   1.0000        1      456
GRID    12054          1  1.5000 90.0000   1.0000        1     2456
GRID    10055          1  1.7500 78.0000   1.0000        1      456
GRID    12055          1  1.7500 90.0000   1.0000        1     2456
GRID    10056          1  2.0000 78.0000   1.0000        1      456
GRID    12056          1  2.0000 90.0000   1.0000        1     2456
GRID    10077          1  1.0000 78.0000   1.5000        1      456
GRID    12077          1  1.0000 90.0000   1.5000        1     2456
GRID    10078          1  1.2500 78.0000   1.5000        1      456
GRID    12078          1  1.2500 90.0000   1.5000        1     2456
GRID    10079          1  1.5000 78.0000   1.5000        1      456
GRID    12079          1  1.5000 90.0000   1.5000        1     2456
GRID    10080          1  1.7500 78.0000   1.5000        1      456
GRID    12080          1  1.7500 90.0000   1.5000        1     2456
GRID    10081          1  2.0000 78.0000   1.5000        1      456
GRID    12081          1  2.0000 90.0000   1.5000        1     2456
GRID    10102          1  1.0000 78.0000   2.0000        1      456
GRID    12102          1  1.0000 90.0000   2.0000        1     2456
GRID    10103          1  1.2500 78.0000   2.0000        1      456
GRID    12103          1  1.2500 90.0000   2.0000        1     2456
GRID    10104          1  1.5000 78.0000   2.0000        1      456
GRID    12104          1  1.5000 90.0000   2.0000        1     2456
GRID    10105          1  1.7500 78.0000   2.0000        1      456
GRID    12105          1  1.7500 90.0000   2.0000        1     2456
GRID    10106          1  2.0000 78.0000   2.0000        1      456
GRID    12106          1  2.0000 90.0000   2.0000        1     2456
GRID    10127          1  1.0000 78.0000   2.5000        1      456
GRID    12127          1  1.0000 90.0000   2.5000        1     2456
GRID    10128          1  1.2500 78.0000   2.5000        1      456
GRID    12128          1  1.2500 90.0000   2.5000        1     2456
GRID    10129          1  1.5000 78.0000   2.5000        1      456
GRID    12129          1  1.5000 90.0000   2.5000        1     2456
GRID    10130          1  1.7500 78.0000   2.5000        1      456
GRID    12130          1  1.7500 90.0000   2.5000        1     2456
GRID    10131          1  2.0000 78.0000   2.5000        1      456
GRID    12131          1  2.0000 90.0000   2.5000        1     2456
GRID    10152          1  1.0000 78.0000   3.0000        1      456
GRID    12152          1  1.0000 90.0000   3.0000        1     2456
GRID    10153          1  1.2500 78.0000   3.0000        1      456
GRID    12153          1  1.2500 90.0000   3.0000        1     2456
```

```
      GRID        10154         1  1.5000 78.0000  3.0000         1      456
      GRID        12154         1  1.5000 90.0000  3.0000         1     2456
      GRID        10155         1  1.7500 78.0000  3.0000         1      456
      GRID        12155         1  1.7500 90.0000  3.0000         1     2456
      GRID        10156         1  2.0000 78.0000  3.0000         1      456
      GRID        12156         1  2.0000 90.0000  3.0000         1     2456
      GRID        10177         1  1.0000 78.0000  3.5000         1      456
      GRID        12177         1  1.0000 90.0000  3.5000         1     2456
      GRID        10178         1  1.2500 78.0000  3.5000         1      456
      GRID        12178         1  1.2500 90.0000  3.5000         1     2456
      GRID        10179         1  1.5000 78.0000  3.5000         1      456
      GRID        12179         1  1.5000 90.0000  3.5000         1     2456
      GRID        10180         1  1.7500 78.0000  3.5000         1      456
      GRID        12180         1  1.7500 90.0000  3.5000         1     2456
      GRID        10181         1  2.0000 78.0000  3.5000         1      456
      GRID        12181         1  2.0000 90.0000  3.5000         1     2456
      GRID        10202         1  1.0000 78.0000  4.0000         1     3456
      GRID        12202         1  1.0000 90.0000  4.0000         1    23456
      GRID        10203         1  1.2500 78.0000  4.0000         1      456
      GRID        12203         1  1.2500 90.0000  4.0000         1     2456
      GRID        10204         1  1.5000 78.0000  4.0000         1      456
      GRID        12204         1  1.5000 90.0000  4.0000         1     2456
      GRID        10205         1  1.7500 78.0000  4.0000         1      456
      GRID        12205         1  1.7500 90.0000  4.0000         1     2456
      GRID        10206         1  2.0000 78.0000  4.0000         1      456
      GRID        12206         1  2.0000 90.0000  4.0000         1     2456
      CHEXA       10001         1    10003   10004   12004   12003   10028   10029CBC
10001
      +BC10001    12029     12028
      CHEXA       10002         1    10004   10005   12005   12004   10029   10030CBC
10002
      +BC10002    12030     12029
      CHEXA       10003         1    10005   10006   12006   12005   10030   10031CBC
10003
      +BC10003    12031     12030
      CHEXA       10004         1    10028   10029   12029   12028   10053   10054CBC
10004
      +BC10004    12054     12053
      CHEXA       10005         1    10029   10030   12030   12029   10054   10055CBC
10005
      +BC10005    12055     12054
      CHEXA       10006         1    10030   10031   12031   12030   10055   10056CBC
10006
      +BC10006    12056     12055
      CHEXA       10007         1    10053   10054   12054   12053   10078   10079CBC
10007
      +BC10007    12079     12078
      CHEXA       10008         1    10054   10055   12055   12054   10079   10080CBC
10008
      +BC10008    12080     12079
      CHEXA       10009         1    10055   10056   12056   12055   10080   10081CBC
10009
      +BC10009    12081     12080
      CHEXA       10010         1    10077   10078   12078   12077   10102   10103CBC
10010
      +BC10010    12103     12102
      CHEXA       10011         1    10078   10079   12079   12078   10103   10104CBC
10011
      +BC10011    12104     12103
      CHEXA       10012         1    10079   10080   12080   12079   10104   10105CBC
```

```
10012
     +BC10012  12105   12104
     CHEXA     10013       1   10080   10081   12081   12080   10105   10106CBC
10013
     +BC10013  12106   12105
     CHEXA     10014       1   10102   10103   12103   12102   10127   10128CBC
10014
     +BC10014  12128   12127
     CHEXA     10015       1   10103   10104   12104   12103   10128   10129CBC
10015
     +BC10015  12129   12128
     CHEXA     10016       1   10104   10105   12105   12104   10129   10130CBC
10016
     +BC10016  12130   12129
     CHEXA     10017       1   10105   10106   12106   12105   10130   10131CBC
10017
     +BC10017  12131   12130
     CHEXA     10018       1   10127   10128   12128   12127   10152   10153CBC
10018
     +BC10018  12153   12152
     CHEXA     10019       1   10128   10129   12129   12128   10153   10154CBC
10019
     +BC10019  12154   12153
     CHEXA     10020       1   10129   10130   12130   12129   10154   10155CBC
10020
     +BC10020  12155   12154
     CHEXA     10021       1   10130   10131   12131   12130   10155   10156CBC
10021
     +BC10021  12156   12155
     CHEXA     10022       1   10152   10153   12153   12152   10177   10178CBC
10022
     +BC10022  12178   12177
     CHEXA     10023       1   10153   10154   12154   12153   10178   10179CBC
10023
     +BC10023  12179   12178
     CHEXA     10024       1   10154   10155   12155   12154   10179   10180CBC
10024
     +BC10024  12180   12179
     CHEXA     10025       1   10155   10156   12156   12155   10180   10181CBC
10025
     +BC10025  12181   12180
     CHEXA     10026       1   10177   10178   12178   12177   10202   10203CBC
10026
     +BC10026  12203   12202
     CHEXA     10027       1   10178   10179   12179   12178   10203   10204CBC
10027
     +BC10027  12204   12203
     CHEXA     10028       1   10179   10180   12180   12179   10204   10205CBC
10028
     +BC10028  12205   12204
     CHEXA     10029       1   10180   10181   12181   12180   10205   10206CBC
10029
     +BC10029  12206   12205
     PLOAD     130000.00     10077   10078   12078   12077
     PLOAD     130000.00     10078   10053   12053   12078
     PLOAD     130000.00     10053   10028   12028   12053
     PLOAD     130000.00     10028   10003   12003   12028
     PLOAD     130000.00     10003   10004   12004   12003
     PLOAD     130000.00     10004   10005   12005   12004
     PLOAD     130000.00     10005   10006   12006   12005
```

```
PLOAD          130000.00    10006    10031    12031    12006
PLOAD          130000.00    10031    10056    12056    12031
PLOAD          130000.00    10056    10081    12081    12056
GRID     4002         1  1.0000 12.0000  0.0000        1     456
GRID     6002         1  1.0000 34.0000  0.0000        1     456
GRID     6003         1  1.2500 34.0000  0.0000        1     456
GRID     6004         1  1.5000 34.0000  0.0000        1     456
GRID     6005         1  1.7500 34.0000  0.0000        1     456
GRID     6006         1  2.0000 34.0000  0.0000        1     456
GRID     4027         1  1.0000 12.0000   .5000        1     456
GRID     6027         1  1.0000 34.0000   .5000        1     456
GRID     6028         1  1.2500 34.0000   .5000        1     456
GRID     6029         1  1.5000 34.0000   .5000        1     456
GRID     6030         1  1.7500 34.0000   .5000        1     456
GRID     6031         1  2.0000 34.0000   .5000        1     456
GRID     4052         1  1.0000 12.0000  1.0000        1     456
GRID     6052         1  1.0000 34.0000  1.0000        1     456
GRID     6053         1  1.2500 34.0000  1.0000        1     456
GRID     6054         1  1.5000 34.0000  1.0000        1     456
GRID     6055         1  1.7500 34.0000  1.0000        1     456
GRID     6056         1  2.0000 34.0000  1.0000        1     456
GRID     6077         1  1.0000 34.0000  1.5000        1     456
GRID     6078         1  1.2500 34.0000  1.5000        1     456
GRID     6079         1  1.5000 34.0000  1.5000        1     456
GRID     6080         1  1.7500 34.0000  1.5000        1     456
GRID     6081         1  2.0000 34.0000  1.5000        1     456
GRID     6102         1  1.0000 34.0000  2.0000        1     456
GRID     6103         1  1.2500 34.0000  2.0000        1     456
GRID     6104         1  1.5000 34.0000  2.0000        1     456
GRID     6105         1  1.7500 34.0000  2.0000        1     456
GRID     6106         1  2.0000 34.0000  2.0000        1     456
GRID     6127         1  1.0000 34.0000  2.5000        1     456
GRID     6128         1  1.2500 34.0000  2.5000        1     456
GRID     6129         1  1.5000 34.0000  2.5000        1     456
GRID     6130         1  1.7500 34.0000  2.5000        1     456
GRID     6131         1  2.0000 34.0000  2.5000        1     456
GRID     6152         1  1.0000 34.0000  3.0000        1     456
GRID     6153         1  1.2500 34.0000  3.0000        1     456
GRID     6154         1  1.5000 34.0000  3.0000        1     456
GRID     6155         1  1.7500 34.0000  3.0000        1     456
GRID     6156         1  2.0000 34.0000  3.0000        1     456
GRID     6177         1  1.0000 34.0000  3.5000        1     456
GRID     6178         1  1.2500 34.0000  3.5000        1     456
GRID     6179         1  1.5000 34.0000  3.5000        1     456
GRID     6180         1  1.7500 34.0000  3.5000        1     456
GRID     6181         1  2.0000 34.0000  3.5000        1     456
GRID     6202         1  1.0000 34.0000  4.0000        1    3456
GRID     6203         1  1.2500 34.0000  4.0000        1     456
GRID     6204         1  1.5000 34.0000  4.0000        1     456
GRID     6205         1  1.7500 34.0000  4.0000        1     456
GRID     6206         1  2.0000 34.0000  4.0000        1     456
CHEXA    4001         1    4002    4003    6003    6002    4027    4028CBC
4001
+BC 4001 6028    6027
CHEXA    4002         1    4003    4004    6004    6003    4028    4029CBC
4002
+BC 4002 6029    6028
CHEXA    4003         1    4004    4005    6005    6004    4029    4030CBC
4003
+BC 4003 6030    6029
```

```
    CHEXA     4004      1    4005    4006    6006    6005    4030    4031CBC
4004
    +BC 4004  6031    6030
    CHEXA     4005      1    4027    4028    6028    6027    4052    4053CBC
4005
    +BC 4005  6053    6052
    CHEXA     4006      1    4028    4029    6029    6028    4053    4054CBC
4006
    +BC 4006  6054    6053
    CHEXA     4007      1    4029    4030    6030    6029    4054    4055CBC
4007
    +BC 4007  6055    6054
    CHEXA     4008      1    4030    4031    6031    6030    4055    4056CBC
4008
    +BC 4008  6056    6055
    CHEXA     4009      1    4052    4053    6053    6052    4077    4078CBC
4009
    +BC 4009  6078    6077
    CHEXA     4010      1    4053    4054    6054    6053    4078    4079CBC
4010
    +BC 4010  6079    6078
    CHEXA     4011      1    4054    4055    6055    6054    4079    4080CBC
4011
    +BC 4011  6080    6079
    CHEXA     4012      1    4055    4056    6056    6055    4080    4081CBC
4012
    +BC 4012  6081    6080
    CHEXA     4013      1    4077    4078    6078    6077    4102    4103CBC
4013
    +BC 4013  6103    6102
    CHEXA     4014      1    4078    4079    6079    6078    4103    4104CBC
4014
    +BC 4014  6104    6103
    CHEXA     4015      1    4079    4080    6080    6079    4104    4105CBC
4015
    +BC 4015  6105    6104
    CHEXA     4016      1    4080    4081    6081    6080    4105    4106CBC
4016
    +BC 4016  6106    6105
    CHEXA     4017      1    4102    4103    6103    6102    4127    4128CBC
4017
    +BC 4017  6128    6127
    CHEXA     4018      1    4103    4104    6104    6103    4128    4129CBC
4018
    +BC 4018  6129    6128
    CHEXA     4019      1    4104    4105    6105    6104    4129    4130CBC
4019
    +BC 4019  6130    6129
    CHEXA     4020      1    4105    4106    6106    6105    4130    4131CBC
4020
    +BC 4020  6131    6130
    CHEXA     4021      1    4127    4128    6128    6127    4152    4153CBC
4021
    +BC 4021  6153    6152
    CHEXA     4022      1    4128    4129    6129    6128    4153    4154CBC
4022
    +BC 4022  6154    6153
    CHEXA     4023      1    4129    4130    6130    6129    4154    4155CBC
4023
    +BC 4023  6155    6154
```

```
    CHEXA       4024         1     4130     4131     6131     6130     4155     4156CBC
4024
    +BC 4024    6156     6155
    CHEXA       4025         1     4152     4153     6153     6152     4177     4178CBC
4025
    +BC 4025    6178     6177
    CHEXA       4026         1     4153     4154     6154     6153     4178     4179CBC
4026
    +BC 4026    6179     6178
    CHEXA       4027         1     4154     4155     6155     6154     4179     4180CBC
4027
    +BC 4027    6180     6179
    CHEXA       4028         1     4155     4156     6156     6155     4180     4181CBC
4028
    +BC 4028    6181     6180
    CHEXA       4029         1     4177     4178     6178     6177     4202     4203CBC
4029
    +BC 4029    6203     6202
    CHEXA       4030         1     4178     4179     6179     6178     4203     4204CBC
4030
    +BC 4030    6204     6203
    CHEXA       4031         1     4179     4180     6180     6179     4204     4205CBC
4031
    +BC 4031    6205     6204
    CHEXA       4032         1     4180     4181     6181     6180     4205     4206CBC
4032
    +BC 4032    6206     6205
    PLOAD      130000.00          4002     4003     6003     6002
    PLOAD      130000.00          4003     4004     6004     6003
    PLOAD      130000.00          4004     4005     6005     6004
    PLOAD      130000.00          4005     4006     6006     6005
    PLOAD      130000.00          4006     4031     6031     6006
    PLOAD      130000.00          4031     4056     6056     6031
    PLOAD      130000.00          4056     4081     6081     6056
    GRID        8002         1  1.0000  56.0000   0.0000        1      456
    GRID        8003         1  1.2500  56.0000   0.0000        1      456
    GRID        8004         1  1.5000  56.0000   0.0000        1      456
    GRID        8005         1  1.7500  56.0000   0.0000        1      456
    GRID        8006         1  2.0000  56.0000   0.0000        1      456
    GRID        8027         1  1.0000  56.0000    .5000        1      456
    GRID        8028         1  1.2500  56.0000    .5000        1      456
    GRID        8029         1  1.5000  56.0000    .5000        1      456
    GRID        8030         1  1.7500  56.0000    .5000        1      456
    GRID        8031         1  2.0000  56.0000    .5000        1      456
    GRID        8052         1  1.0000  56.0000   1.0000        1      456
    GRID        8053         1  1.2500  56.0000   1.0000        1      456
    GRID        8054         1  1.5000  56.0000   1.0000        1      456
    GRID        8055         1  1.7500  56.0000   1.0000        1      456
    GRID        8056         1  2.0000  56.0000   1.0000        1      456
    GRID        8077         1  1.0000  56.0000   1.5000        1      456
    GRID        8078         1  1.2500  56.0000   1.5000        1      456
    GRID        8079         1  1.5000  56.0000   1.5000        1      456
    GRID        8080         1  1.7500  56.0000   1.5000        1      456
    GRID        8081         1  2.0000  56.0000   1.5000        1      456
    GRID        8102         1  1.0000  56.0000   2.0000        1      456
    GRID        8103         1  1.2500  56.0000   2.0000        1      456
    GRID        8104         1  1.5000  56.0000   2.0000        1      456
    GRID        8105         1  1.7500  56.0000   2.0000        1      456
    GRID        8106         1  2.0000  56.0000   2.0000        1      456
    GRID        8127         1  1.0000  56.0000   2.5000        1      456
```

```
GRID            8128       1  1.2500 56.0000  2.5000       1     456
GRID            8129       1  1.5000 56.0000  2.5000       1     456
GRID            8130       1  1.7500 56.0000  2.5000       1     456
GRID            8131       1  2.0000 56.0000  2.5000       1     456
GRID            8152       1  1.0000 56.0000  3.0000       1     456
GRID            8153       1  1.2500 56.0000  3.0000       1     456
GRID            8154       1  1.5000 56.0000  3.0000       1     456
GRID            8155       1  1.7500 56.0000  3.0000       1     456
GRID            8156       1  2.0000 56.0000  3.0000       1     456
GRID            8177       1  1.0000 56.0000  3.5000       1     456
GRID            8178       1  1.2500 56.0000  3.5000       1     456
GRID            8179       1  1.5000 56.0000  3.5000       1     456
GRID            8180       1  1.7500 56.0000  3.5000       1     456
GRID            8181       1  2.0000 56.0000  3.5000       1     456
GRID            8202       1  1.0000 56.0000  4.0000       1    3456
GRID            8203       1  1.2500 56.0000  4.0000       1     456
GRID            8204       1  1.5000 56.0000  4.0000       1     456
GRID            8205       1  1.7500 56.0000  4.0000       1     456
GRID            8206       1  2.0000 56.0000  4.0000       1     456
CHEXA           6001       1    6002    6003    8003    8002    6027    6028CBC
6001
+BC 6001        8028    8027
CHEXA           6002       1    6003    6004    8004    8003    6028    6029CBC
6002
+BC 6002        8029    8028
CHEXA           6003       1    6004    6005    8005    8004    6029    6030CBC
6003
+BC 6003        8030    8029
CHEXA           6004       1    6005    6006    8006    8005    6030    6031CBC
6004
+BC 6004        8031    8030
CHEXA           6005       1    6027    6028    8028    8027    6052    6053CBC
6005
+BC 6005        8053    8052
CHEXA           6006       1    6028    6029    8029    8028    6053    6054CBC
6006
+BC 6006        8054    8053
CHEXA           6007       1    6029    6030    8030    8029    6054    6055CBC
6007
+BC 6007        8055    8054
CHEXA           6008       1    6030    6031    8031    8030    6055    6056CBC
6008
+BC 6008        8056    8055
CHEXA           6009       1    6052    6053    8053    8052    6077    6078CBC
6009
+BC 6009        8078    8077
CHEXA           6010       1    6053    6054    8054    8053    6078    6079CBC
6010
+BC 6010        8079    8078
CHEXA           6011       1    6054    6055    8055    8054    6079    6080CBC
6011
+BC 6011        8080    8079
CHEXA           6012       1    6055    6056    8056    8055    6080    6081CBC
6012
+BC 6012        8081    8080
CHEXA           6013       1    6077    6078    8078    8077    6102    6103CBC
6013
+BC 6013        8103    8102
CHEXA           6014       1    6078    6079    8079    8078    6103    6104CBC
6014
```

```
    +BC 6014    8104    8103
    CHEXA       6015       1    6079    6080    8080    8079    6104    6105CBC
6015
    +BC 6015    8105    8104
    CHEXA       6016       1    6080    6081    8081    8080    6105    6106CBC
6016
    +BC 6016    8106    8105
    CHEXA       6017       1    6102    6103    8103    8102    6127    6128CBC
6017
    +BC 6017    8128    8127
    CHEXA       6018       1    6103    6104    8104    8103    6128    6129CBC
6018
    +BC 6018    8129    8128
    CHEXA       6019       1    6104    6105    8105    8104    6129    6130CBC
6019
    +BC 6019    8130    8129
    CHEXA       6020       1    6105    6106    8106    8105    6130    6131CBC
6020
    +BC 6020    8131    8130
    CHEXA       6021       1    6127    6128    8128    8127    6152    6153CBC
6021
    +BC 6021    8153    8152
    CHEXA       6022       1    6128    6129    8129    8128    6153    6154CBC
6022
    +BC 6022    8154    8153
    CHEXA       6023       1    6129    6130    8130    8129    6154    6155CBC
6023
    +BC 6023    8155    8154
    CHEXA       6024       1    6130    6131    8131    8130    6155    6156CBC
6024
    +BC 6024    8156    8155
    CHEXA       6025       1    6152    6153    8153    8152    6177    6178CBC
6025
    +BC 6025    8178    8177
    CHEXA       6026       1    6153    6154    8154    8153    6178    6179CBC
6026
    +BC 6026    8179    8178
    CHEXA       6027       1    6154    6155    8155    8154    6179    6180CBC
6027
    +BC 6027    8180    8179
    CHEXA       6028       1    6155    6156    8156    8155    6180    6181CBC
6028
    +BC 6028    8181    8180
    CHEXA       6029       1    6177    6178    8178    8177    6202    6203CBC
6029
    +BC 6029    8203    8202
    CHEXA       6030       1    6178    6179    8179    8178    6203    6204CBC
6030
    +BC 6030    8204    8203
    CHEXA       6031       1    6179    6180    8180    8179    6204    6205CBC
6031
    +BC 6031    8205    8204
    CHEXA       6032       1    6180    6181    8181    8180    6205    6206CBC
6032
    +BC 6032    8206    8205
    PLOAD       130000.00    6002    6003    8003    8002
    PLOAD       130000.00    6003    6004    8004    8003
    PLOAD       130000.00    6004    6005    8005    8004
    PLOAD       130000.00    6005    6006    8006    8005
    PLOAD       130000.00    6006    6031    8031    8006
```

```
PLOAD          130000.00     6031     6056     8056     8031
PLOAD          130000.00     6056     6081     8081     8056
GRID    10002          1   1.0000  78.0000   0.0000        1      456
GRID    10027          1   1.0000  78.0000    .5000        1      456
GRID    10052          1   1.0000  78.0000   1.0000        1      456
CHEXA    8001          1     8002     8003    10003    10002     8027     8028CBC
8001
+BC 8001 10028    10027
CHEXA    8002          1     8003     8004    10004    10003     8028     8029CBC
8002
+BC 8002 10029    10028
CHEXA    8003          1     8004     8005    10005    10004     8029     8030CBC
8003
+BC 8003 10030    10029
CHEXA    8004          1     8005     8006    10006    10005     8030     8031CBC
8004
+BC 8004 10031    10030
CHEXA    8005          1     8027     8028    10028    10027     8052     8053CBC
8005
+BC 8005 10053    10052
CHEXA    8006          1     8028     8029    10029    10028     8053     8054CBC
8006
+BC 8006 10054    10053
CHEXA    8007          1     8029     8030    10030    10029     8054     8055CBC
8007
+BC 8007 10055    10054
CHEXA    8008          1     8030     8031    10031    10030     8055     8056CBC
8008
+BC 8008 10056    10055
CHEXA    8009          1     8052     8053    10053    10052     8077     8078CBC
8009
+BC 8009 10078    10077
CHEXA    8010          1     8053     8054    10054    10053     8078     8079CBC
8010
+BC 8010 10079    10078
CHEXA    8011          1     8054     8055    10055    10054     8079     8080CBC
8011
+BC 8011 10080    10079
CHEXA    8012          1     8055     8056    10056    10055     8080     8081CBC
8012
+BC 8012 10081    10080
CHEXA    8013          1     8077     8078    10078    10077     8102     8103CBC
8013
+BC 8013 10103    10102
CHEXA    8014          1     8078     8079    10079    10078     8103     8104CBC
8014
+BC 8014 10104    10103
CHEXA    8015          1     8079     8080    10080    10079     8104     8105CBC
8015
+BC 8015 10105    10104
CHEXA    8016          1     8080     8081    10081    10080     8105     8106CBC
8016
+BC 8016 10106    10105
CHEXA    8017          1     8102     8103    10103    10102     8127     8128CBC
8017
+BC 8017 10128    10127
CHEXA    8018          1     8103     8104    10104    10103     8128     8129CBC
8018
+BC 8018 10129    10128
CHEXA    8019          1     8104     8105    10105    10104     8129     8130CBC
```

```
8019
     +BC 8019    10130    10129
     CHEXA        8020        1     8105     8106    10106    10105     8130    8131CBC
8020
     +BC 8020    10131    10130
     CHEXA        8021        1     8127     8128    10128    10127     8152    8153CBC
8021
     +BC 8021    10153    10152
     CHEXA        8022        1     8128     8129    10129    10128     8153    8154CBC
8022
     +BC 8022    10154    10153
     CHEXA        8023        1     8129     8130    10130    10129     8154    8155CBC
8023
     +BC 8023    10155    10154
     CHEXA        8024        1     8130     8131    10131    10130     8155    8156CBC
8024
     +BC 8024    10156    10155
     CHEXA        8025        1     8152     8153    10153    10152     8177    8178CBC
8025
     +BC 8025    10178    10177
     CHEXA        8026        1     8153     8154    10154    10153     8178    8179CBC
8026
     +BC 8026    10179    10178
     CHEXA        8027        1     8154     8155    10155    10154     8179    8180CBC
8027
     +BC 8027    10180    10179
     CHEXA        8028        1     8155     8156    10156    10155     8180    8181CBC
8028
     +BC 8028    10181    10180
     CHEXA        8029        1     8177     8178    10178    10177     8202    8203CBC
8029
     +BC 8029    10203    10202
     CHEXA        8030        1     8178     8179    10179    10178     8203    8204CBC
8030
     +BC 8030    10204    10203
     CHEXA        8031        1     8179     8180    10180    10179     8204    8205CBC
8031
     +BC 8031    10205    10204
     CHEXA        8032        1     8180     8181    10181    10180     8205    8206CBC
8032
     +BC 8032    10206    10205
     PLOAD           130000.00     8002     8003    10003    10002
     PLOAD           130000.00     8003     8004    10004    10003
     PLOAD           130000.00     8004     8005    10005    10004
     PLOAD           130000.00     8005     8006    10006    10005
     PLOAD           130000.00     8006     8031    10031    10006
     PLOAD           130000.00     8031     8056    10056    10031
     PLOAD           130000.00     8056     8081    10081    10056
     MAT1                1  .10E+08  .38E+07  .32E+00  .10E+00
     CORD1C              1      101      102      103
     GRID              101               0.0      0.0      0.0              123456
     GRID              102               0.0      0.0      1.0              123456
     GRID              103               1.0      0.0      1.0              123456
     GRAV                2        1  .76E+05      0.0      0.0     -1.0
     PSOLID              1        1        1
     ENDDATA
```

DISTRIBUTION LIST

No. of Copies	Organization
12	Commander Defense Technical Info Center ATTN: DDC-DDA Cameron Station Alexandria, VA 22314
1	Director Defense Advanced Research Projects Agency 1400 Wilson Boulevard Arlington, VA 22209
3	Director Defense Nuclear Agency ATTN: STSP (SPLN) STTI STRA (RARP) Washington, DC 20305
1	HQDA (DAMA-MS) Washington, DC 20310
1	Commander US Army BMD Advanced Technology Center ATTN: BMDATC-M, Mr. P. Boyd P.O. Box 1500 Huntsville, AL 35807
1	Commander US Army Materiel Development and Readiness Command ATTN: DRCDMD-ST 5001 Eisenhower Avenue Alexandria, VA 22333
2	Commander USA ARRADCOM ATTN: DRDAR-SE, COL J.S. Chesbro Dover, NJ 07801
3	Commander USA ARRADCOM ATTN: DRDAR-SCS, Mr. D. Brandt DRDAR-SCS, Mr. S. Jacobson DRDAR-SCF, Mr. K. Pfleger Dover, NJ 07801
2	Commander USA ARRADCOM ATTN: DRDAR-TSS (2 cys) Dover, NJ 07801
3	Commander USA ARRADCOM ATTN: DRDAR-TD DRDAR-TDA DRDAR-TDS Dover, NJ 07801
10	Commander USA ARRADCOM ATTN: DRDAR-LCU, Mr. E. Barrieres DRDAR-LCU, Mr. R. Davitt DRDAR-LCU-M, Mr. D. Robertson DRDAR-LCU-M, Mr. J. Sikra DRDAR-LCU-M, Mr. M. Weinstock DRDAR-LCA, Mr. C. Larson DRDAR-LCA, Mr. B. Knutelski DRDAR-LCR-R, Mr. E.H. Moore III DRDAR-LCA, Dr. H. Fair DRDAR-LCS, Mr. J. Gregorits Dover, NJ 07801
5	Commander USA ARRADCOM ATTN: DRDAR-SCM, Dr. E. Bloore DRDAR-SCM, Mr. J. Mulherin DRDAR-SCS, Mr. B. Brodman DRDAR-SCS, Dr. T. Hung DRDAR-SCA, Mr. W. Gadomski Dover, NJ 07801
1	Commander USA ARRADCOM ATTN: Army Fuze Mgt Project Office DRDAR-FU Dover, NJ 07801
2	Commander USA ARRADCOM ATTN: Development Project Office for Selected Ammunitions DRDAR-DP Dover, NJ 07801

DISTRIBUTION LIST

No. of Copies	Organization
2	Commander USA ARRADCOM ATTN: Product Assurance Directorate DRDAR-QA Dover, NJ 07801
1	Commander US Army Armament Materiel Readiness Command ATTN: DRDAR-LEP-L, Tech Lib Rock Island, IL 61299
2	Director USA ARRADCOM Benet Weapons Laboratory ATTN: DRDAR-LCB-TL, Mr. Rummel Watervliet, NY 12189
1	Commander US Army Aviation Research and Development Command ATTN: DRDAV-E 4300 Goodfellow Blvd. St. Louis, MO 63120
1	Director US Army Air Mobility Research and Development Laboratory Ames Research Center Moffett Field, CA 94035
2	Director US Army Research and Technology Laboratories (AVRADCOM) Fort Eustis, VA 23604
1	Commander US Army Communications Research and Development Command ATTN: DRDCO-PPA-SA Fort Monmouth, NJ 07703
1	Commander US Army Electronics Research and Development Command Technical Support Activity ATTN: DELSD-L Fort Monmouth, NJ 07703
1	Commander US Army Harry Diamond Laboratories ATTN: DELHD-TA-L 2800 Powder Mill Road Adelphi, MD 20783
1	Commander US Army Missile Command ATTN: DRSMI-R Redstone Arsenal, AL 35809
1	Commander US Army Missile Command ATTN: DRSMI-RBL Redstone Arsenal, AL 35809
1	Commander US Army Missile Command ATTN: DRSMI-YDL Redstone Arsenal, AL 35809
2	Commander US Army Mobility Equipment Research & Development Command Fort Belvoir, VA 22060
2	Commander US Army Tank Automotive Research and Development Command ATTN: DRDTA-UL Technical Director Warren, MI 48090
3	Project Manager Cannon Artillery Weapons System ATTN: DRCPM-CAWS Dover, NJ 07801
2	Project Manager Nuclear Munitions ATTN: DRCPM-NUC Dover, NJ 07801
2	Project Manager Tank Main Armament Systems ATTN: DRCPM-TMA Dover, NJ 07801

DISTRIBUTION LIST

No. of Copies	Organization
2	Project Manager Division Air Defense Gun ATTN: DRCPM-ADG Dover, NJ 07801
1	Product Manager for 30mm Ammo. ATTN: DRCPM-AAH-30mm Dover, NJ 07801
2	Product Manager M110E2 Weapon System, DARCOM ATTN: DRCPM-M110E2 Rock Island, IL 61299
2	Commander US Army Materials and Mechanics Research Center ATTN: J. Mescall Tech Library Watertown, MA 02172
3	Commander US Army Research Office P.O. Box 12211 ATTN: Technical Director Engineering Division Metallurgy & Materials Division Research Triangle Park NC 27709
2	Director US Army TRADOC Systems Analysis Activity ATTN: ATAA-SL, Tech Lib White Sands Missile Range NM 88002
1	Commander Naval Air Systems Command ATTN: AIR-604 Washington, DC 20360
2	Commander David W. Taylor Ship Research and Development Center Bethesda, MD 20084
1	Commander Naval Air Development Center Johnsville Warminster, PA 18974
1	Commander Naval Missile Center Point Mugu, CA 95041
2	Commander Naval Surface Weapons Center Silver Spring, MD 20910
2	Commander Naval Surface Weapons Center ATTN: Code TX, Dr. W.G. Soper Dahlgren, VA 22448
3	Commander Naval Weapons Center ATTN: Code 3835 Sewell B. Lundstrom M. Backman China Lake, CA 93555
1	Commander Naval Explosive Ordnance Disposal Facility ATTN: Lib Div Indian Head, MD 20604
1	Commander Naval Research Laboratory Washington, DC 20375
4	Commander Naval Research Laboratory ATTN: Mr. W.J. Ferguson Dr. C. Sanday Dr. H. Pusey Dr. F. Rosenthal Washington, DC 20375
3	AFATL (DLJM, W. Dittrich; D. Davis; DLDL) Eglin AFB, FL 32542

DISTRIBUTION LIST

No. of Copies	Organization
2	ADTC/DLJW (MAJ D. Matuska; LTC j. Osborn) Eglin AFB, FL 32542
2	AFWL/SUL Kirtland AFB, NM 87117
1	AFML/LLN (Dr. T. Nicholas) Wright-Patterson AFB, OH 45433
2	ASD (XROT, Gerald Bennett; ENFTV, Martin Lentz) Wright-Patterson AFB, OH 45433
1	U.S. Department of the Interior Bureau Mines Pittsburgh Technical Support Center ATTN: Dr. S.G. Sawyer Pittsburgh, PA 15213
1	Director Lawrence Livermore Laboratory ATTN: Dr. R.H. Toland, L-424 Livermore, CA 94550
1	Director Lawrence Livermore Laboratory ATTN: Dr. M.L. Wilkins Livermore, CA 94550
1	Director Lawrence Livermore Laboratory ATTN: Dr. R. Werne Livermore, CA 94550
1	Director Los Alamos Scientific Laboratory P.O. Box 1663 Los Alamos, NM 87544
1	Headquarters National Aeronautics and Space Administration Washington, DC 20546
1	Director National Aeronautics and Space Administration Manned Spacecraft Center ATTN: Library Houston, TX 77058
2	Aerospace Corporation ATTN: Mr. L. Rubin Mr. L.G. King 2350 E. El Segundo Blvd. El Segundo, CA 90009
1	Effects Technology, Inc. 5383 Hollister Avenue P.O. Box 30400 Santa Barbara, CA 93105
2	Falcon R&D Company Thor Facility ATTN: Mr. D. Malick Mr. J. Wilson 696 Fairmount Avenue Baltimore, MD 21204
2	Falcon R&D Company ATTN: L. Smith R. Miller 109 Inverness Drive, East Englewood, CO 80112
1	FMC Corporation Ordnance Engineering Division San Jose, CA 95114
1	General Electric Company Armament Systems Department Burlington, VT 05401
1	President General Research Corporation ATTN: Lib McLean, VA 22101
1	Goodier Aerospace Corp. 1210 Massillon Road Akron, OH 44315

DISTRIBUTION LIST

No. of Copies	Organization
3	Honeywell, Inc. Government and Aerospace Products Division ATTN: Mr. J. Blackburn Dr. G. Johnson Mr. R. Simpson 600 Second Street, NE Hopkins, MN 55343
1	Director Jet Propulsion Laboratory ATTN: Lib (TD) 4800 Oak Grove Drive Pasadena, CA 91103
1	Kaman Sciences Corporation ATTN: Dr. P. Snow 1500 Garden of the Gods Road Colorado Springs, CO 80933
1	Lockheed Corporation ATTN: Dr. C.E. Vivian Sunnyvale, CA 94087
1	Lockheed Huntsville ATTN: Dr. L.A. Picklesimer P.O. Box 1103 Huntsville, AL 35809
1	Materials Research Laboratory, Inc. 1 Science Road Glenwood, IL 60427
1	McDonnell Douglas Astronautics ATTN: Mail Station 21-2 Dr. J. Wall 5301 Bolsa Avenue Huntington Beach, CA 92647
1	Pacific Technical Corporation ATTN: Dr. P.K. Feldmann 460 Ward Drive Santa Barbara, CA 93105
2	Physics International Company ATTN: Dr. D. Orphal Dr. E.T. Moore 2700 Merced Street San Leandro, CA 94577
1	Rockwell International Automatics Missile Systems Division ATTN: Dr. Wayne T. Armburst 4300 E. 5th Avenue Columbus, OH 43216
3	Sandia Laboratories ATTN: Dr. W. Herrmann Dr. L. Bertholf Dr. J.W. Nunziato Albuquerque, NM 87115
1	Science Applications, Inc. 101 Continental Blvd, Suite 310 El Segundo, CA 90245
1	Science Applications, Inc. ATTN: G. Burghart 201 W. Dyer Rd (Unit B) Santa Ana, CA 92707
2	Systems, Science & Software ATTN: Dr. R. Sedgwick Ms. L. Hageman P.O. Box 1620 La Jolla, CA 92038
1	US Steel Corporation Research Center 125 Jamison Lane Monroeville, PA 15146
1	Drexel University Dept. of Mechanical Engineering ATTN: Dr. P.C. Chou 32nd and Chestnut Streets Philadelphia, PA 19104
1	Illinois Institute of Technology Dept of Mechanics & Aerospace Engineering ATTN: Dr. P.H. Francis Chicago, IL 60616
1	New Mexico Institute of Mining and Technology Terra Group Socorro, NM 87801

DISTRIBUTION LIST

No. of Copies	Organization
2	Southwest Research Institute Dept of Mechanical Sciences ATTN: Dr. U. Lindholm Dr. W. Baker 8500 Culebra Road San Antonio, TX 78228
3	SRI International ATTN: Dr. L. Seaman Dr. D. Curran Dr. D. Shockey 333 Ravenswood Aveune Menlo Park, CA 94025
1	Stanford University Stanford Linear Accelerator Center ATTN: Eric Grosse, Numerical Analysis Consultant ALAC, P.O. Box 4349 Stanford, CA 94305
2	University of Arizona Civil Engineering Department ATTN: Dr. D.A. DaDeppo Dr. R. Richard Tucson, AZ 85721
4	University of California Los Alamos Scientific Laboratory ATTN: Dr. R. Karpp Dr. J. Dienes Dr. L. Germain Dr. B. Germain P.O. Box 808 Livermore, CA 94550
1	University of Dayton Research Institute ATTN: S.J. Bless Dayton, OH 45469
1	University of Denver Denver Research Institute ATTN: Mr. R.F. Recht 2390 South University Boulevard Denver, CO 80210
1	University of Illinois Aeronautical and Astronautical Engineering Department 101 Transportation Bldg. ATTN: Prof. A.R. Zak Urbana, IL 61801
1	University of Illinois Department of Mathematics ATTN: Dr. Evelyn Frank Urbana, IL 61801
1	University of Illinois Dept. of Mechanical Engineering ATTN: Prof Herman Krier 144 Meb, 1206 W. Green Street Urbana, IL 61801
2	University of Wisconsin-Madison Mathematics Research Center ATTN: Dr. John Nohel 610 Walnut Street Madison, WI 53706
1	Virginia Commonwealth University Dept. of Math Sciences 901 W. Franklin Richmond, VA 23284

Aberdeen Proving Ground

Dir, USAMSAA
ATTN: DRXSY-D
DRXSY-MP, H. Cohen
Cdr, USATECOM
ATTN: DRSTE-TO-F
Dir, USACSL, Bldg. E3516, EA
ATTN: DRDAR-CLB-PA
DRDAR-CL
DRDAR-CLD
DRDAR-CLB
DRDAR-CLY
DRDAR-CLN

USER EVALUATION OF REPORT

Please take a few minutes to answer the questions below; tear out this sheet, fold as indicated, staple or tape closed, and place in the mail. Your comments will provide us with information for improving future reports.

1. BRL Report Number _____

2. Does this report satisfy a need? (Comment on purpose, related project, or other area of interest for which report will be used.)

_____

_____

_____

3. How, specifically, is the report being used? (Information source, design data or procedure, management procedure, source of ideas, etc.) _____

_____

_____

4. Has the information in this report led to any quantitative savings as far as man-hours/contract dollars saved, operating costs avoided, efficiencies achieved, etc.? If so, please elaborate.

_____

_____

5. General Comments (Indicate what you think should be changed to make this report and future reports of this type more responsive to your needs, more usable, improve readability, etc.) _____

_____

_____

_____

6. If you would like to be contacted by the personnel who prepared this report to raise specific questions or discuss the topic, please fill in the following information.

Name: _____

Telephone Number: _____

Organization Address: _____

_____

_____