เอกสารอ้างอิง

1. Beck, H., "Contribution to the Analysis of Coupled Shear Walls", Journal, American Concrete Institute, Detroit, Michigan., Vol. 59, 1962, pp. 1055-69.

2. Rosman, R., "Approximate Analysis of Shear Walls Subjected to Lateral Loads", Journal, American Concrete Institute, Detroit, Michigan, Vol. 61, 1964, pp. 717-732.

3. Coull, Alexander, and Puri, Ravi D., "Analysis of Pierced Shear Walls", Journal of the Structural Division, ASCE, Vol. 94, No. ST1, Proc. Paper 5710, January, 1968, pp. 71-82.

4. Sensmeier, Paul E., "An Analytical and Experimental Study of Shear Walls with Openings", Thesis presented to Purdue University, in Lafayette, Indiana, in June, 1967, in partial fulfillment of the requirements for the degree of Doctor of Philosophy.

5. Girijavallabhan, C.V., "Analysis of Shear Walls with Openings", Journal of Structural Division, ASCE, Vol. 95, October 1969, pp. 2093-2103.

6. Irons, B.M., "A Frontal Solution Program for Finite Element Analysis", Int. J. Num. Meth. Engng, 2, 5-32 (1970).

7. Johnson, C.P., "A Frontal-Based Solver with Multi-Level Substructuring" Proceedings, North Carolina State University, May 23-25, 1977, pp. 381-384.

8. Beer, G. and Haas, W., "A Partitioned Frontal Solver For Finite Element Analysis", Int. J. Num. Meth. Engng, Vol. 18, 1623-1654 (1982).

9. Natarajan, R., "Front Solution Program for Transmission Tower Analysis", Computers & Structures, Vol. 5, 1975. pp. 59-64.

10. Klans-Jürgen Bathe and Wilson, E.L., Numerical Methods in Finite Element Analysis, Prentice-Hall, Inc., New Jersey, 1976.

11. Cook, R.C., Concepts and Applications of Finite Element Analysis, 2nd Edition, John Wiley & Sons, Inc., 1981.

12. Przeminiecki, J.S., Matrix Structural Analysis of Substructures, American Institute of Aeronautics and Astronautics Journal, Vol. 1, 1963, pp. 138-147.

13. Zienkiewiez, O.C., The Finite Element Method, 3rd Edition, McGraw-Hill, New York, 1977.

14. Hinton, E. and Owen, D.R.J., Finite Element Programming, Academic Press, London, 1977.

15. Peterson, H. and Popov, E.P., "Substructuring and Equation System Solutions in Finite Element Analysis", Computers & Structures, Vol. 7, No. 2, 1977, pp. 197-206.

16. Pekau, O.A. and Huttelmaier, H.P., "A Versatile Panel Element for the Analysis of Shear Wall Structures", Computers & Structures, Vol. 12, 1980, pp. 349-359.

17. Tiong, G.H., "A Frontal Substructuring Scheme for Analysis of Unbranched Prismatic Structures", M. Eng. Thesis No. ST-80-6, Asian Institute of Technology, Bangkok, Thailand, March 1980.

18. Rajatabhothi, R., "Multi-Level Substructuring of Large Structural Systems for the Eigenproblem", Ph.D. Dissertation Present to the Faculty of Graduate School of University of Texas at Austin, August 1982.

19. Udomlurgchai, S., "Development of a General Purpose Finite Element Program on Microcomputer (Apple II)", M. Eng. Thesis No. ST-83-18, Asian Institute of Technology, Bangkok, Thailand, 1983.

20. ภาณุวัฒน์ คุรุรัตน์, "โปรแกรมไมโครคอมพิวเตอร์ซึ่งใช้วิธีฟรอนทัลในการวิเคราะห์โครงสร้างชนิดโครงระนาบ", วิทยานิพนธ์ปริญญามหาบัณฑิต ภาควิชาวิศวกรรมโยธา บัณฑิตวิทยาลัย จุฬาลงกรณ์มหาวิทยาลัย, 2527.

21. เริงเดชา รัชตโพธิ์, "โปรแกรมไมโครคอมพิวเตอร์สำหรับระบบไฟไนท์เอเลเมนท์โดยใช้วิธีโครงสร้างย่อย", รายงานวิจัย-คณะวิศวกรรมศาสตร์, 2526 (กำลังอยู่ในระหว่างดำเนินงาน).

22. การปิโตรเลี่ยมแห่งประเทศไทย, "การศึกษาผลกระทบอันจะมีต่อแนวท่อส่งก๊าซธรรมชาติบนบก", ภาควิชาวิศวกรรมโยธา คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย, พฤศจิกายน 2527.

ภาคผนวก ก

รายละเอียดโปรแกรมคอมพิวเตอร์

```
1000  REM              ******************
1010  REM              ***   REMARK   ***
1020  REM              ******************
1030  REM    ::::::::::::::::::::::::::::::::::::::::::::::
1040  REM    ::   CONTROL INFORMATION IN THIS PROGRAM   ::
1050  REM    ::::::::::::::::::::::::::::::::::::::::::::::
1060  REM
1070  REM    PT          : LARGEST NODE NUMBER
1080  REM    EL          : NUMBER OF FINITE ELEMENTS
1090  REM    NF          : NUMBER OF FINITE ELEMENTS SAVE ON DISK
1100  REM    MN          : NUMBER OF NODES IN EACH ELEMENT ON LEVEL 1
1110  REM    JZ          : MAX. NUMBER OF NODES IN SUBSTRUCTURE ON
1120  REM                : ANY LEVEL
1130  REM    PS          : MAX. NUMBER OF ELEMENTS IN ANY LEVEL
1140  REM    NS          : MAX. NUMBER OF BOUNDARY CONSTRAINT NODES
1150  REM    BN          : NUMBER OF SUBSTRUCTURE IN LEVEL 1
1160  REM    KSUB        : NUMBER OF STEPS IN MULTI-LEVEL SUBSTRUCTURE
1170  REM    DM          : NUMBER OF MATERIAL TYPES
1180  REM    TE          : ELEMENT TYPE  -:  PLANE STRAIN  = 1
1190  REM                                -:  PLANE STRESS  = 2
1200  REM    LU          : MAX. NUMBER OF THE SAME SUBSTRUCTURE ON ANY
1210  REM                  LEVEL
1220  REM    E3          : NUMBER OF NODAL POINT LOADS
1230  REM    RT          : REFERENCE TEMPERATURE
1240  REM
1250  REM
1260  REM    ::::::::::::::::::::::::::::::::::::::::::::::
1270  REM    ::   SUBPROGRAM  INPUT DATA , PRINT DATA   ::
1280  REM    ::               CHANGE DATA , PLOT        ::
1290  REM    ::::::::::::::::::::::::::::::::::::::::::::::
1300  REM
1310  REM    DL          : NUMBER OF DISTRIBUTED LOADS
1320  REM    DN          : NUMBER OF NODES WHICH DISTRIBUTED LOAD IS
1330  REM                  ASSOCIATED
1340  REM    DOF         : NUMBER OF DOF. IN EACH NODE
1350  REM    IX          : ELEMENT TYPE NUMBER
1360  REM    JS          : NUMBER OF NODES IN ANY SUBSTRUCTURE
1370  REM    LB          : NUMBER OF ELEMENTS IN ANY SUBSTRUCTURE
1380  REM    NB          : NUMBER OF SUBSTRUCTURE IN ANY LEVEL
1390  REM    NE          : NUMBER OF NODE IN EACH ELEMENT TYPE
1400  REM    NP          : NUMBER OF BOUNDARY CONSTRAINT NODES IN ANY
1410  REM                  LEVEL
1420  REM    THC         : THICKNESS IN EACH ELEMENT
1430  REM    AP(I)       : ARRAY STORES COEF. OF TEMPERATURE OF
1440  REM                  MATERIAL TYPE
1450  REM    CD(I,J)     : ARRAY STORES NUMBER OF BOUNDARY CONSTRAINT
1460  REM                  NODES & BOUNDARY CONSTRAINT CODE
1470  REM    DEN(I)      : ARRAY STORES MASS DENSITY OF ELEMENT TYPE
1480  REM    DISN(I,J)   : ARRAY STORES NODE NUMBER WHICH DISTRIBUTED
1490  REM                  LOAD IS ASSOCIATED
1500  REM    ELN(I)      : ARRAY STORES ELEMENT NUMBER WHICH
```

```
1510  REM                 DISTRIBUTED LOAD IS ASSOCIATED
1520  REM    FS(I)      : ARRAY STORES NUMBER OF ELEMENTS IN ANY
1530  REM                 SUBSTRUCTURE
1540  REM    IQ(I)      : ARRAY STORES NODE NUMBER OF ELEMENTS
1550  REM    LO(I,J)    : ARRAY STORES NODE NUMBER WHICH NODAL POINT
1560  REM                 LOAD IS ASSOCIATED
1570  REM    ND(I)      : ARRAY STORES NUMBER OF DOF. IN EACH NODE
1580  REM    NP(I)      : ARRAY STORES NODE NUMBER OF STRUCTURE
1590  REM    PR(I)      : ARRAY STORES POISSON'S RATIO OF ELEMENT TYPE
1600  REM    R1(I,J)    : ARRAY STORES NODAL POINT LOADS IN X- Coor.
1610  REM    R2(I,J)    : ARRAY STORES NODAL POINT LOADS IN Y- Coor.
1620  REM    R3(I,J)    : ARRAY STORES NORMAL     DISTRIBUTED LOAD
1630  REM    R4(I,J)    : ARRAY STORES TANGENTIAL DISTRIBUTED LOAD
1640  REM    SM(I)      : ARRAY STORES SUBSTRUCTURE NODE NUMBER IN
1650  REM                 EACH SUBSTRUCTURE
1660  REM    SN(I,J)    : ARRAY STORES DISPLACEMENT IN X,Y- Coor.
1670  REM                 OF BOUNDARY CONSTRAINT NODES
1680  REM    T(I)       : ARRAY STORES NODAL TEMPERATURE
1690  REM    X(I)       : ARRAY STORES X- Coor. IN EACH NODE
1700  REM    Y(I)       : ARRAY STORES Y- Coor. IN EACH NODE
1710  REM    YM(I)      : ARRAY STORES YOUNG'S MODULUS OF MATERIAL TYPE
1720  REM
1730  REM
1740  REM    ::::::::::::::::::::::::::::::::::::::::::::::::
1750  REM    ::            SUBPROGRAM  STIFFNESS           ::
1760  REM    ::  THIS SUBPROGRAM PERFORMS STIFFNESS MATRIX  ::
1770  REM    ::       AND LOAD VECTOR OF EACH ELEMENT       ::
1780  REM    ::       SAVE DATA INTO FILE  " STIFF1 "       ::
1790  REM    ::::::::::::::::::::::::::::::::::::::::::::::::
1800  REM
1810  REM    NG         : NUMBER OF GAUSS QUADRATURE ORDER
1820  REM    B(I,J)     : ARRAY STORES STRAIN-DISPLACEMENT MATRIX
1830  REM    E(I,J)     : ARRAY STORES MATRIX OF ELASTIC CONSTANTS
1840  REM    EN(I)      : ARRAY STORES SHAPE FUNCTIONS OF
1850  REM                 ISOPARAMETRIC ELEMENT
1860  REM    ES(I)      : ARRAY STORES INITIAL SHEARING STRAINS AT NODES
1870  REM    EX(I)      : ARRAY STORES INITIAL NORMAL STRAIN AT NODES
1880  REM                 IN X-Coor.
1890  REM    EY(I)      : ARRAY STORES INITIAL NORMAL STRAIN AT NODES
1900  REM                 IN Y-Coor.
1910  REM    JAC(I,J)   : ARRAY STORES JACOBIAN MATRIX
1920  REM    NTE(I,J)   : ARRAY STORES DERIVATIVE OF SHAPE FUNCTION
1930  REM                 IN   - Coor.
1940  REM    NXI(I,J)   : ARRAY STORES DERIVATIVE OF SHAPE FUNCTION
1950  REM                 IN   - Coor.
1960  REM    P(I,J,K)   : ARRAY STORES LOAD VECTOR OF ELEMENT
1970  REM    PL(I,J)    : ARRAY STORES LOCATIONS OF GAUSS QUADRATURE
1980  REM    S(I,J)     : ARRAY STORES STIFFNESS MATRIX OF ELEMENT
1990  REM    WGT(I,J)   : ARRAY STORES WEIGHTS OF GAUSS QUADRATURE
2000  REM
```

```
2010  REM
2020  REM   ::::::::::::::::::::::::::::::::::::::::::::::::::
2030  REM                  SUBPROGRAM  FRNTIQ                 ::
2040  REM   ::   THIS SUBPROGRAM CONTROLS THE PREFRONT AT    ::
2050  REM   ::     THE SUBSTRUCTURE LEVEL AND THEN SETS      ::
2060  REM   ::   DESTINATION ARRAYS FOR FRONT  PROCESSING    ::
2070  REM   ::::::::::::::::::::::::::::::::::::::::::::::::::
2080  REM
2090  REM   B8         : MAX. NUMBER OF DOF. IN THE FRONT
2100  REM   D1         : NUMBER OF DOF. OF ELEMENT
2110  REM   M          : NUMBER OF ELIMINATED NODES IN A FRONT
2120  REM   N          : NUMBER OF           NODES IN A FRONT
2130  REM   QM         : NUMBER OF ELIMINATED DOF.  IN A FRONT
2140  REM   QN         : NUMBER OF           DOF.  IN A FRONT
2150  REM   U1         : NUMBER OF ELIMINATED NODES IN CURRENT FRONT
2160  REM   V1         : NUMBER OF           NODES IN CURRENT FRONT
2170  REM   V2         : NUMBER OF           NODES IN NEXT FRONT
2180  REM   AMOC(I)    : ARRAY STORES ACTIVE  NODES IN CURRENT FRONT
2190  REM   BMOC(I)    : ARRAY STORES ACTIVE  NODES IN NEXT FRONT
2200  REM   IB(I)      : ARRAY STORES NUMBER OF ELEMENTS THAT
2210  REM                CONNECTED AT NODES
2220  REM   IC(I)      : ARRAY STORES NUMBER OF ELEMENTS THAT
2230  REM                CONNECTED AT NODES IN EACH FRONT
2240  REM   IQ(I)      : ARRAY STORES NODE NUMBER OF ELEMENTS
2250  REM   LF(I)      : ARRAY STORES LOCATIONS OF INTERIOR DOF.IN
2260  REM                ELEMENT OR IN THE RETAINED PORTION OF FRONT
2270  REM   LF(I)      : ARRAY STORES BOUNDARY CONSTRAINTS CODE AT
2280  REM                (D1+I)LOCATIONS IN FINAL SUBSTRUCTURE LEVEL
2290  REM   MOC(I)     : ARRAY STORES ACTIVE  NODES IN FRONT
2300  REM
2310  REM
2320  REM   ::::::::::::::::::::::::::::::::::::::::::::::::::
2330  REM   ::              SUBPROGRAM  FRNTST               ::
2340  REM   ::   THIS SUBPROGRAM PERFORMS FRONT PROCESSING   ::
2350  REM   ::    FROM SUBSTRUCTURE LEVEL 1 TO LEVEL KSUB     ::
2360  REM   ::::::::::::::::::::::::::::::::::::::::::::::::::
2370  REM
2380  REM   A(I)       : ARRAY STORES STIFFNESS OF ELEMENTS
2390  REM   B(I,J)     : ARRAY STORES LOAD VECTOR OF ELEMENTS
2400  REM   NO(I)      : ARRAY STORES LOCATION OF DIAGONAL STIFFNESS
2410  REM                OF ELEMENT IN FRONT
2420  REM   S(I)       : ARRAY STORES STIFFNESS MATRIX AND LOAD
2430  REM                VECTOR OF ELEMENTS
2440  REM   SD(I)      : ARRAY STORES DISPLACEMENT IN X,Y- Coor. OF
2450  REM                BOUNDARY CONSTRAINT NODES
2460  REM
```

```
2470  REM
2480  REM   ::::::::::::::::::::::::::::::::::::::::::::::::::
2490  REM   ::     SUBPROGRAM BACKSUB,STRESS,PRINT RESULT     ::
2500  REM   ::   THIS SUBPROGRAM PERFORMS BACKSUBSTITUTION    ::
2510  REM   ::   TO FIND NODAL DISPLACEMENT,ELEMENT STRESS    ::
2520  REM   ::  PRINT NODAL DISPLACEMENT & ELEMENT STRESS     ::
2530  REM   ::::::::::::::::::::::::::::::::::::::::::::::::::
2540  REM
2550  REM   XC         : STORES X-Coor. OF ELEMENT AT GAUSS POINT
2560  REM                FOR 4-NODE ELEMENT
2570  REM   YC         : STORES Y-Coor. OF ELEMENT AT GAUSS POINT
2580  REM                FOR 4-NODE ELEMENT
2590  REM   A(I)       : ARRAY STORES REDUCED STIFFNESS OF ELEMENTS
2600  REM   E(I,J)     : ARRAY STORES DISPLACEMENT AT NODES FOR SAVE
2610  REM                ON DATA FILE " DISPL "
2620  REM   R(I)       : ARRAY STORES X-Coor. IN EACH NODE
2630  REM   SIG(I)     : ARRAY STORES STRESS AND PRINCIPAL STRESS
2640  REM                IN EACH ELEMENT
2650  REM   UX(I)      : ARRAY STORES NODAL DISPLACEMENT IN X-Coor.
2660  REM   UY(I)      : ARRAY STORES NODAL DISPLACEMENT IN Y-Coor.
2670  REM   W(I)       : ARRAY STORES STRESS AND PRINCIPAL STRESS OF
2680  REM                ELEMENT AT GAUSS POINT FOR 4-NODE ELEMENT
2690  REM   W(I,J)     : ARRAY STORES STRESS AND PRINCIPAL STRESS OF
2700  REM                ELEMENT AT GAUSS POINT FOR 8-NODE ELEMENT
2710  REM   X(I,J)     : ARRAY STORES REDUCED LOAD VECTOR
2720  REM   X(I,J)     : ARRAY STORES FRONT DISPLACEMENT
2730  REM   XC(I)      : ARRAY STORES X-Coor. OF ELEMENT AT GAUSS POINTS
2740  REM                FOR 8-NODE ELEMENT
2750  REM   Y(I,J)     : ARRAY STORES DISPLACEMENT AT NODES FOR READ
2760  REM                FROM DATA FILE " DISPL "
2770  REM   YC(I)      : ARRAY STORES Y-Coor. OF ELEMENT AT GAUSS POINTS
2780  REM                FOR 8-NODE ELEMENT
2790  REM   Z(I)       : ARRAY STORES Y-Coor. IN EACH NODE
```

```
1000  REM      ****************************************
1010  REM      ***                                  ***
1020  REM      ***                                  ***
1030  REM      *** A FRONTAL SUBSTRUCTURING SCHEME ***
1040  REM      ***                                  ***
1050  REM      ***   FOR ANALYSIS OF SHEAR WALLS    ***
1060  REM      ***                                  ***
1070  REM      ***        BY MICROCOMPUTER          ***
1080  REM      ***                                  ***
1090  REM      ***                                  ***
1100  REM      ****************************************
1110  REM
1120  REM    ::::::::::::::::::::::::::::::::::::::::::::::::::
1130  REM    :: THIS PROGRAM RUNS ON MICROCOMPUTER (Apple II ::
1140  REM    :: Apple IIe) FOR ANALYSIS OF SHEAR WALLS       ::
1150  REM    :: ELEMENT    : 4-POINT,8-POINT ISOPARAMETRIC   ::
1160  REM    ::              ELEMENT. LINEAR ELASTICITY      ::
1170  REM    :: LOAD       : NODAL POINT LOAD , DISTRIBUTED  ::
1180  REM    ::              LOAD , BODY FORCE , TEMPERATURE ::
1190  REM    ::              CHANGE                          ::
1200  REM    :: SUBPROGRAM : INPUT DATA , PRINT DATA , PLOT  ::
1210  REM    ::              CHANGE DATA , FRNTIQ , FRNTST   ::
1220  REM    ::              BACKSUB , STRESS , PRINT RESULT ::
1230  REM    ::::::::::::::::::::::::::::::::::::::::::::::::::
1240  REM
1250  HOME : REM   **  OPTIONS  **
1260  PRINT " ------------------------------------": PRINT
1270  PRINT "      COMMAND                NUMBER"
1280  PRINT "      -------                ------": PRINT
1290  PRINT "  - INPUT DATA                 1": PRINT
1300  PRINT "  - PRINT DATA                 2": PRINT
1310  PRINT "  - PLOT                       3": PRINT
1320  PRINT "  - CHANGE DATA                4": PRINT
1330  PRINT "  - EXECUTE                    5": PRINT
1340  PRINT "  - PRINT RESULT(Q4)           6": PRINT
1350  PRINT "  - PRINT RESULT(Q8)           7": PRINT
1360  PRINT "  - EXIT                       8": PRINT
1370  VTAB 22: INPUT "      Choose number....(1,8)  ";AA
1380  LET D$ =  CHR$ (4): IF AA > 8 THEN 1370
1390  ON AA GOTO 1400,1410,1420,1430,1440,1450,1460,1470
1400  PRINT D$;"RUN INPUT DATA,D1"
1410  PRINT D$;"RUN PRINT DATA,D1"
1420  PRINT D$;"RUN PLOT,D1"
1430  PRINT D$;"RUN CHANGE DATA,D1"
1440  PRINT D$;"RUN EXECUTE,D1"
1450  PRINT D$;"RUN PRINT RESULT(Q4),D1"
1460  PRINT D$;"RUN PRINT RESULT(Q8),D1"
1470  END
```

```
1000  HOME : REM   ** HEADING **
1010  GOTO 1030
1020  PRINT  TAB( 1)"****************************************": RETURN
1030  GOSUB 1020
1040  VTAB 4: PRINT  TAB( 5)"A FRONTAL SUBSTRUCTURING SCHEME": PRINT : PRINT : PRINT
     TAB( 7)"FOR ANALYSIS OF SHEAR WALLS": PRINT : PRINT : PRINT  TAB( 12)"BY MICRO
    COMPUTER"
1050  PRINT : GOSUB 1020
1060  VTAB 22: PRINT  TAB( 7)"Press RETURN key to continue";: INPUT "  ";A$
1070  PRINT  CHR$ (4);"RUN OPTIONS,D1"
```

```
1000  ONERR  GOTO 1420
1010  GOSUB 1020: GOTO 1060
1020  HOME : REM      **  INPUT DATA  **
1030  PRINT "--------------": PRINT
1040  PRINT "  INPUT DATA  ": PRINT
1050  PRINT "--------------": PRINT : RETURN
1060  VTAB 23: INPUT "Do you display DATA on monitor?(Y/N) ";B$: IF B$ <  > "Y" AND
    B$ <  > "N" THEN 1060
1070  IF B$ = "Y" THEN BB = 0: GOTO 1110
1080  VTAB 23: INPUT "Do you print   DATA on printer?(Y/N) ";B$: IF B$ <  > "Y" AND
    B$ <  > "N" THEN 1080
1090  IF B$ = "N" THEN  PRINT  CHR$ (4);"RUN OPTIONS,D1"
1100  IF B$ = "Y" THEN BB = 1: PRINT  CHR$ (4);"PR#1"
1110  GOSUB 1020: PRINT : PRINT : PRINT "  TITLE :   SHEAR WALL"
1120  PRINT : INPUT "LARGEST NODE NUMBER                = ";PT
1130  PRINT : INPUT "NUM. OF FINITE ELEMENTS            = ";EL
1140  PRINT : INPUT "NUM. OF FINITE ELEMENTS ON DISK    = ";NF
1150  PRINT : INPUT "NUM. OF NODES/ELEMENT IN LEVEL 1 = ";MN
1160  PRINT : INPUT "MAX. NUM. OF NODES IN SUBST.       = ";JZ
1170  PRINT : INPUT "MAX. NUM. OF ELEMENT IN ANY SUBST= ";PS
1180  PRINT : INPUT "MAX. NUM. OF BOUND. CON. NODES     = ";NS
1190  PRINT : INPUT "NUM. OF SUBSTRUCTURES IN LEVEL 1 = ";BN
1200  PRINT : INPUT "NUM. OF SUBSTRUCTURE LEVELS        = ";KSUB
1210  PRINT : INPUT "NUMBER OF MATERIAL TYPES           = ";DM
1220  PRINT : PRINT "ELEMENT TYPE : (PLANE STRAIN = 1)  ": INPUT "              : (P
    LANE STRESS = 2)  ";TE
1230  PRINT : INPUT "MAX. NUM. OF THE SAME SUBST.       = ";LU
1240  PRINT : INPUT "NUM. OF NODAL POINT LOADS          = ";E3
1250  PRINT : INPUT "REFERENCE TEMPERATURE              = ";RT
1260  PRINT : PRINT "DO THE ELEMENT IS THE SAME TYPE": PRINT : PRINT
1270  VTAB 23: INPUT "(IN SUBST. FLOOR) ? (Y/N) ....      ";B$
1280  IF B$ <  > "Y" AND B$ <  > "N" THEN 1270
1290  LET MT = 1: IF B$ = "Y" THEN MT = PS
1300  PRINT : PRINT : GOSUB 1400: IF B$ = "N" THEN 1110
1310  LET LV = 1:M1 = 2 * MN:XC = (M1 * (M1 + 1)) / 2 + (M1 * LU * MT):LQ = 4
1320  IF JZ <  = 0 AND MN < 2 * NS THEN JZ = 2 * NS
1330  IF JZ <  = 0 AND MN =  > 2 * NS THEN JZ = MN
1340  LET ZQ = 6 * JZ + 36: IF KSUB > 2 THEN ZQ = 9 * JZ + 36
1350  LET D$ =  CHR$ (4):ZA = ((MN * 2) * (MN * 2 + 1) + MN * 4 * LU * MT) * 10 + 3
    6:YT = (JZ + BN + 3 + (3 * NS)) * 6:X1 =  - 1:E3 = E3 + 1:LY = 0:MI = 2 * JZ: IF
    MN = 4 THEN 1370
1360  DIM NI(EL),NJ(EL),NK(EL),NL(EL)
1370  DIM IQ(MN),ND(PT),LO(E3,LV),R1(E3,LV),R2(E3,LV),FS(BN),CD(NS,3),SM(JZ),X(PT),
    Y(PT),FT(BN + 1)
1380  DIM T(PT),IX(EL),PR(DM),YM(DM),DEN(DM),AP(DM),JQ(MN),VV(BN),II(EL),IJ(EL),IK(
    EL),IL(EL),THC(EL),SN(NS,2)
1390  GOTO 1450
1400  VTAB 23: PRINT "Do you accept these data ? (Y/N)  ";: INPUT " ";B$: IF B$ <  >
    "Y" AND B$ <  > "N" THEN 1400
1410  RETURN
1420  CALL  - 998: RESUME
```

```
1430  VTAB 24: PRINT "Press RETURN key to continue ...  ";: INPUT " ";B$: RETURN
1440  PRINT  TAB( 1)"----------------------------------------": RETURN
1450  PRINT D$;"OPEN SUBST,D2": PRINT D$;"DELETE SUBST": PRINT D$;"CLOSE SUBST"
1460  PRINT D$;"OPEN FRONT,D2": PRINT D$;"DELETE FRONT": PRINT D$;"CLOSE FRONT"
1470  LET ZT =  - 1: GOSUB 3840
1480  REM      **  INPUT NODAL POINT COORDINATES  **
1490  HOME : PRINT : GOSUB 1440
1500  PRINT "  GLOBAL NODAL COORDINATE": PRINT : GOSUB 1440
1510  VTAB 8: PRINT "DO YOU WANT TO INPUT NODAL COOR.": PRINT
1520  INPUT "BY YOURSELF ?  (Y/N) ......   ";B$: IF B$ <  > "Y" AND B$ <  > "N" THEN
    1510
1530  IF B$ = "Y" THEN 1740
1540  VTAB 12: INPUT "WIDTH , HEIGHT OF THE SHEAR WALL= ";W,H
1550  VTAB 13: PRINT "                                  "
1560  VTAB 14: INPUT "NUM OF DIVISION IN WIDTH ,HEIGHT= ";SW,SH
1570  GOSUB 1400: IF B$ = "N" THEN 1540
1580  LET SW = SW + 1:SH = SH + 1
1590  VTAB 16: PRINT "DO YOU INPUT NODE NUMBER IN X-AXIS": PRINT
1600  INPUT "RESPECTIVELY ? (Y/N).......   ";B$: IF B$ <  > "Y" AND B$ <  > "N" THEN
    1590
1610  IF B$ = "N" THEN 1680
1620  FOR J = 1 TO SH:L = (J - 1) * SW + 1:X(L) = 0
1630  FOR I = 2 TO SW:L = (J - 1) * SW + I:X(L) = X(L - 1) + W / (SW - 1): NEXT I
1640  NEXT J:K = H / (SH - 1)
1650  FOR J = 1 TO SH
1660  FOR I = 1 TO SW:L = (J - 1) * SW + I:Y(L) = K * (J - 1): NEXT I
1670  NEXT J: GOTO 2070
1680  FOR J = 1 TO SW:L = (J - 1) * SH + 1:Y(L) = 0
1690  FOR I = 2 TO SH:L = (J - 1) * SH + I:Y(L) = Y(L - 1) + H / (SH - 1): NEXT I
1700  NEXT J:K = W / (SW - 1)
1710  FOR J = 1 TO SW
1720  FOR I = 1 TO SH:L = (J - 1) * SH + I:X(L) = K * (J - 1): NEXT I
1730  NEXT J: GOTO 2070
1740  HOME :W = 0:H = 0:L = 0:I = 0
1750  HOME : PRINT : GOSUB 1440
1760  VTAB 4: PRINT "     INPUT NODAL POINT COORDINATES": PRINT : GOSUB 1440
1770  IF BB = 1 THEN 1870
1780  IF L <  > 0 THEN 1830
1790  VTAB 9: PRINT "  NODE No.            = "
1800  VTAB 11: PRINT "  GENERATE INCREMENT  = "
1810  VTAB 13: PRINT "  X - Coordinate      = "
1820  VTAB 15: PRINT "  Y - Coordinate      = ": GOTO 1870
1830  VTAB 9: PRINT "  NODE No.            = ";L
1840  VTAB 11: PRINT "  GENERATE INCREMENT  = ";DZ(1)
1850  VTAB 13: PRINT "  X - Coordinate      = ";J
1860  VTAB 15: PRINT "  Y - Coordinate      = ";K
1870  VTAB 9: INPUT "  NODE No.            = ";I: IF I < 1 OR I > PT THEN 1870
1880  VTAB 11: INPUT "  GENERATE INCREMENT  = ";DZ(2): IF DZ(2) < 0 THEN 1880
1890  VTAB 13: INPUT "  X - Coordinate      = ";X(I):J = X(I)
1900  VTAB 15: INPUT "  Y - Coordinate      = ";Y(I):K = Y(I)
```

```
1910  GOSUB 1400: IF B$ = "N" THEN 1870
1920  IF I = 1 OR I <  = L OR DZ(I) = 0 THEN 2000
1930  LET DZ = (I - L) / DZ(I):M = DZ(I): IF DZ = 1 THEN M = DZ:DZ = DZ(I)
1940  LET DX = (X(I) - X(L)) / DZ:DY = (Y(I) - Y(L)) / DZ
1950  LET L = L + M: IF I = L THEN 2000
1960  IF L < I THEN 1990
1970  LET L = I(I):J = X(L):K = Y(L)
1980  VTAB 23: INPUT "  ERROR IN GENERATE NODE : REENTER ";B$: GOTO 1770
1990  LET X(L) = X(L - M) + DX:Y(L) = Y(L - M) + DY: GOTO 1950
2000  LET L = I:I(I) = I:DZ(I) = DZ(2): IF I < PT THEN 1770
2010  FOR I = 2 TO PT: IF X(I) = X(1) AND Y(I) = Y(1) THEN L = 0: GOTO 1980
2020  NEXT I
2030  FOR I = 1 TO PT: IF W < X(I) THEN W = X(I)
2040  IF H < Y(I) THEN H = Y(I)
2050  NEXT I
2060  REM      **  INPUT NODAL TEMPERATURES  **
2070  HOME : VTAB 7: GOSUB 1440
2080  VTAB 9: PRINT "     DO INITIAL STRAIN EFFECT FROM": IF BB = 1 THEN 2100
2090  VTAB 11: PRINT "     TEMPERATURE ? (Y/N) ......  ": PRINT : GOSUB 1440
2100  VTAB 11: INPUT "     TEMPERATURE ? (Y/N) ......  ";B$: IF B$ <  > "Y" AND B$ <
     > "N" THEN 2080
2110  FOR L = 1 TO PT:T(L) = RT: NEXT L:I = 0:L = 0: IF B$ = "N" THEN 2280
2120  HOME : PRINT : GOSUB 1440
2130  VTAB 4: PRINT "        INPUT NODAL TEMPERATURE"
2140  VTAB 6: PRINT "   DIFFERENT FROM REF. TEMP. & NODE";PT: PRINT : GOSUB 1440
2150  IF BB = 1 THEN 2210
2160  IF I <  > 0 THEN 2190
2170  VTAB 11: PRINT "  NODE No.          = "
2180  VTAB 13: PRINT "  TEMPERATURE       = ": GOTO 2210
2190  VTAB 11: PRINT "  NODE No.          = ";I
2200  VTAB 13: PRINT "  TEMPERATURE       = ";J
2210  VTAB 11: INPUT "  NODE No.          = ";I
2220  IF I < 1 OR I > PT THEN 2210
2230  VTAB 13: INPUT "  TEMPERATURE       = ";T(I):J = T(I)
2240  GOSUB 1400: IF B$ = "N" THEN 2210
2250  IF I < PT THEN 2150
2260  REM      **  INPUT DISPLACEMENT AT SUPPORT.  **
2270  REM      **  SAVE INTO DATA FILE " SUPSET "  **
2280  HOME : VTAB 7: GOSUB 1440
2290  VTAB 9: PRINT "   DO THE SUPPORT HAVE DISPLACEMENT ?": IF BB = 1 THEN 2310
2300  VTAB 11: PRINT "    (Y/N)    ...................    ": PRINT : GOSUB 1440
2310  VTAB 11: INPUT "    (Y/N)    ...................    ";B$: IF B$ <  > "Y" AND
    B$ <  > "N" THEN 2290
2320  FOR I = 1 TO NS:SN(I,1) = 0:SN(I,2) = 0: NEXT I
2330  IF B$ = "N" THEN 2470
2340  HOME : VTAB 2: GOSUB 1440
2350  VTAB 4: PRINT "        SUPPORT DISPLACEMENT": PRINT : GOSUB 1440
2360  FOR I = 1 TO NS
2370  VTAB 9: PRINT "NODE No.                  = ";CD(I,1): IF BB = 1 THEN 2430
2380  IF I <  > 1 THEN 2410
```

```
2390  VTAB 11: PRINT "DISPLACEMENT IN X-Coor.   = "
2400  VTAB 13: PRINT "DISPLACEMENT IN Y-Coor.   = ": GOTO 2430
2410  VTAB 11: PRINT "DISPLACEMENT IN X-Coor.   = ";J
2420  VTAB 13: PRINT "DISPLACEMENT IN Y-Coor.   = ";K
2430  VTAB 11: INPUT "DISPLACEMENT IN X-Coor.   = ";SN(I,1):J = SN(I,1)
2440  VTAB 13: INPUT "DISPLACEMENT IN Y-Coor.   = ";SN(I,2):K = SN(I,2)
2450  GOSUB 1400: IF B$ = "N" THEN 2430
2460  NEXT I
2470  PRINT D$;"OPEN SUPSET,D2": PRINT D$;"DELETE SUPSET"
2480  PRINT D$;"OPEN SUPSET": PRINT D$;"WRITE SUPSET"
2490  FOR I = 1 TO NS: PRINT SN(I,1): PRINT SN(I,2): NEXT I
2500  PRINT D$;"CLOSE SUPSET"
2510  FOR I = 1 TO EL:IX(I) = 0:THC(I) = 0: NEXT I
2520  FOR I = 1 TO BN:VV(I) = 0: NEXT I
2530  PRINT D$;"OPEN ELE1,D2": PRINT D$;"DELETE ELE1": PRINT D$;"CLOSE ELE1"
2540  PRINT D$;"OPEN ELE2,D2": PRINT D$;"DELETE ELE2": PRINT D$;"CLOSE ELE2"
2550  PRINT D$;"OPEN ELE3,D2": PRINT D$;"DELETE ELE3": PRINT D$;"CLOSE ELE3"
2560  IF E3 > 1 THEN  GOSUB 3330
2570  HOME : VTAB 7: GOSUB 1440
2580  VTAB 9: PRINT "     DO YOU INPUT DISTRIBUTED LOADS": IF BB = 1 THEN 2600
2590  VTAB 11: PRINT "      (Y/N)  .................    ": PRINT : GOSUB 1440
2600  VTAB 11: INPUT "      (Y/N)  .................    ";B$: IF B$ <  > "Y" AND B$
     <  > "N" THEN 2600
2610  LET DL = 0:DN = 0: IF B$ = "Y" THEN  GOSUB 3540
2620  REM      ** LOOP OVER EACH SUBSTRUCTURE  **
2630  FOR IT = 1 TO BN:VV = 1:LB = FT(IT): IF LB < 0 THEN 3060
2640  HOME : PRINT : GOSUB 1440
2650  FOR I = IT TO BN:J = I + 1: IF J > BN OR FT(J) > 0 THEN I = BN
2660  IF FT(J) < 0 THEN VV = VV + 1
2670  NEXT I
2680  PRINT "NUM. OF THE SAME SUBST. No.";IT;"  = ";VV:VV(IT) = VV: PRINT : GOSUB 1
    440: GOSUB 1430
2690  REM      ** LOOP OVER EACH ELEMENT  **
2700  HOME : FOR LZ = 1 TO LB:LY = LY + 1: VTAB 2: GOSUB 1440
2710  REM      **  INPUT NUMBER OF NODES , NODE NUMBERS AND PROPERTY  **
2720  REM      **  OF ELEMENTS. SAVE INTO DATA FILE " ELE1 "          **
2730  VTAB 4: PRINT "NUM. OF NODE &NODE NUM. OF ELE. No.";LY: PRINT : GOSUB 1440
2740  LET NE = MN: PRINT : PRINT "NUMBER OF ELEMENT NODE        = ";NE: IF BB = 1 THEN
    2820
2750  IF LZ <  > 1 THEN 2790
2760  VTAB 10: FOR L = 1 TO NE: PRINT "NODE NUMBER FOR ELEMENT      = ": NEXT L
2770  PRINT "TYPE OF ELEMENT                = "
2780  PRINT "THICKNESS OF ELEMENT           = ": GOTO 2820
2790  VTAB 10: FOR L = 1 TO NE: PRINT "NODE NUMBER FOR ELEMENT       = ";IQ(L): NEXT
    L
2800  PRINT "TYPE OF ELEMENT                = ";IX
2810  PRINT "THICKNESS OF ELEMENT           = ";THC
2820  VTAB 10: FOR L = 1 TO NE: INPUT "NODE NUMBER FOR ELEMENT       = ";IQ(L):JQ(L
     ) = IQ(L)
2830  IF IQ(L) < 1 OR IQ(L) > PT THEN 2820
2840  NEXT L
```

```
2850  INPUT "TYPE OF ELEMENT                = ";IX
2860  INPUT "THICKNESS OF ELEMENT           = ";THC
2870  GOSUB 1400: IF B$ = "N" THEN 2820
2880  PRINT D$;"OPEN ELE1,L";ZQ:X1 = X1 + 1:LW = X1: PRINT D$;"WRITE ELE1,R";LW
2890  PRINT NE
2900  FOR I = 1 TO NE: PRINT IQ(I): NEXT I
2910  PRINT D$;"CLOSE ELE1"
2920  PRINT D$;"OPEN ELE2,L";ZQ: PRINT D$;"WRITE ELE2,R";LW
2930  PRINT NE
2940  FOR I = 1 TO NE: PRINT IQ(I): NEXT I
2950  PRINT D$;"CLOSE ELE2": IF KSUB < 3 THEN 3000
2960  PRINT D$;"OPEN ELE3,L";ZQ: PRINT D$;"WRITE ELE3,R";LW
2970  PRINT NE
2980  FOR I = 1 TO NE: PRINT IQ(I): NEXT I
2990  PRINT D$;"CLOSE ELE3"
3000  FOR I = 1 TO VV: IF I = 1 THEN 3020
3010  FOR J = 1 TO NE:JQ(J) = JQ(J) - FT(IT + I - 1): NEXT J
3020  LET K = LY + (I - 1) * LB:II(K) = JQ(1):IJ(K) = JQ(2):IK(K) = JQ(3):IL(K) = J
      Q(4):IX(K) = IX:THC(K) = THC: IF MN = 4 THEN 3040
3030  LET NI(K) = JQ(5):NJ(K) = JQ(6):NK(K) = JQ(7):NL(K) = JQ(8)
3040  NEXT I
3050  NEXT LZ:LY = LY + (VV - 1) * LB
3060  NEXT IT: HOME
3070  REM      **  SAVE OTHER DATA INTO FILE " LOCATE "," SUM "  **
3080  REM      **  & " DAT1 " FOR USE THE NEXT SUBPROGRAM        **
3090  PRINT D$;"OPEN LOCATE,D2": PRINT D$;"DELETE LOCATE"
3100  PRINT D$;"OPEN LOCATE": PRINT D$;"WRITE LOCATE"
3110  FOR I = 1 TO PT: PRINT X(I): NEXT I
3120  FOR I = 1 TO PT: PRINT Y(I): NEXT I
3130  FOR I = 1 TO PT: PRINT T(I): NEXT I
3140  FOR I = 1 TO EL: PRINT IX(I): NEXT I
3150  PRINT D$;"CLOSE LOCATE"
3160  PRINT D$;"OPEN SUM,D2": PRINT D$;"DELETE SUM"
3170  PRINT D$;"OPEN SUM": PRINT D$;"WRITE SUM"
3180  PRINT E3: PRINT X1: PRINT DL: PRINT DN: IF DL = 0 THEN 3220
3190  FOR I = 1 TO DL: PRINT ELN(I)
3200  FOR J = 1 TO DN: PRINT DISN(I,J): PRINT R3(I,J): PRINT R4(I,J): NEXT J
3210  NEXT I
3220  FOR L = 1 TO PT: PRINT X(L): PRINT Y(L): PRINT T(L): NEXT L
3230  FOR I = 1 TO E3: PRINT LO(I,1): PRINT R1(I,1): PRINT R2(I,1): NEXT I
3240  FOR I = 1 TO BN: PRINT VV(I): PRINT FT(I): NEXT I
3250  FOR I = 1 TO EL: PRINT IX(I): PRINT THC(I): PRINT II(I): PRINT IJ(I): PRINT I
      K(I): PRINT IL(I): NEXT I: IF MN = 4 THEN 3270
3260  FOR I = 1 TO EL: PRINT NI(I): PRINT NJ(I): PRINT NK(I): PRINT NL(I): NEXT I
3270  PRINT D$;"CLOSE SUM": PRINT D$;"PR#0"
3280  PRINT D$;"OPEN DAT1,D2": PRINT D$;"DELETE DAT1"
3290  PRINT D$;"OPEN DAT1": PRINT D$;"WRITE DAT1"
3300  PRINT XC: PRINT PT: PRINT EL: PRINT MN: PRINT NF: PRINT MI: PRINT PS: PRINT N
      S: PRINT KSUB: PRINT JZ: PRINT ZT: PRINT ZQ: PRINT ZA: PRINT E3: PRINT YT: PRINT
      MT: PRINT LU: PRINT BN: PRINT TE
3310  PRINT D$;"CLOSE DAT1": POKE 216,0
3320  PRINT D$;"RUN OPTIONS,D1"
```

```
3330  LET LO(1,1) = 1:R1(1,1) = 0:R2(1,1) = 0
3340  REM      **  INPUT NODAL POINT LOADS AND NODE NUMBERS  **
3350  HOME : PRINT : GOSUB 1440
3360  VTAB 4: PRINT "INPUT NODAL POINT LOADS IN GLOBAL Coor.": PRINT : GOSUB 1440
3370  FOR I = 2 TO E3: IF BB = 1 THEN 3450
3380  IF I <  > 2 THEN 3420
3390  VTAB 9: PRINT "NODE No.                = "
3400  VTAB 11: PRINT "LOADING IN X-Coor.      = "
3410  VTAB 13: PRINT "LOADING IN Y-Coor.      = ": GOTO 3450
3420  VTAB 9: PRINT "NODE No.                = ";J
3430  VTAB 11: PRINT "LOADING IN X-Coor.      = ";K
3440  VTAB 13: PRINT "LOADING IN Y-Coor.      = ";L
3450  VTAB 9: INPUT "NODE No.                = ";LO(I,1):J = LO(I,1)
3460  IF J < 1 OR J > PT THEN 3450
3470  VTAB 11: INPUT "LOADING IN X-Coor.      = ";R1(I,1):K = R1(I,1)
3480  VTAB 13: INPUT "LOADING IN Y-Coor.      = ";R2(I,1):L = R2(I,1)
3490  GOSUB 1400: IF B$ = "N" THEN 3450
3500  NEXT I
3510  RETURN
3520  REM      **  INPUT NORMAL , TANGENTIAL DISTRIBUTED LOADS  **
3530  REM      **        AND NODE NUMBERS                       **
3540  HOME : PRINT : GOSUB 1440: IF BB = 1 THEN 3560
3550  VTAB 4: PRINT "NUM. OF DISTRIBUTED LOADS      = ": PRINT : GOSUB 1440
3560  VTAB 4: INPUT "NUM. OF DISTRIBUTED LOADS      = ";DL: PRINT : GOSUB 1440
3570  GOSUB 1400: IF B$ = "N" THEN 3560
3580  IF DL = 0 THEN 3820
3590  DIM ELN(DL),DISN(DL,3),R3(DL,3),R4(DL,3):DN = 2: IF MN = 8 THEN DN = 3
3600  HOME : FOR I = 1 TO DL: VTAB 2: GOSUB 1440
3610  VTAB 4: PRINT " DISTRIBUTED LOAD No.";I;" IN GLOBAL Coor.": PRINT : GOSUB 144
   0: IF BB = 1 THEN 3730
3620  IF I <  > 1 THEN 3680
3630  VTAB 9: PRINT "ELEMENT No.                = "
3640  FOR J = 1 TO DN:JJ = 7 + (J * 4)
3650  VTAB JJ: PRINT "NODE No.                  = "
3660  PRINT "NORMAL     DIST. LOAD      = "
3670  PRINT "TANGENTIAL DIST. LOAD      = ": NEXT J: GOTO 3730
3680  VTAB 9: PRINT "ELEMENT No.                = ";K
3690  FOR J = 1 TO DN:JJ = 7 + (J * 4)
3700  VTAB JJ: PRINT "NODE No.                  = ";L(J)
3710  PRINT "NORMAL     DIST. LOAD      = ";M(J)
3720  PRINT "TANGENTIAL DIST. LOAD      = ";N(J): NEXT J
3730  VTAB 9: INPUT "ELEMENT No.                = ";ELN(I):K = ELN(I)
3740  IF K < 1 OR K > EL THEN 3730
3750  FOR J = 1 TO DN:JJ = 7 + (J * 4)
3760  VTAB JJ: INPUT "NODE No.                  = ";DISN(I,J):L(J) = DISN(I,J)
3770  IF L(J) < 1 OR L(J) > PT THEN 3760
3780  INPUT "NORMAL     DIST. LOAD      = ";R3(I,J):M(J) = R3(I,J)
3790  INPUT "TANGENTIAL DIST. LOAD      = ";R4(I,J):N(J) = R4(I,J): NEXT J
3800  GOSUB 1400: IF B$ = "N" THEN 3730
3810  NEXT I
3820  RETURN
```

```
3830  REM      **  LOOP FROM SUBSTRUCTURING LEVEL 1 TO LEVEL KSUB  **
3840  FOR IM = 1 TO KSUB
3850  REM      **  INPUT SUBSTRUCTURE CONTROL DATA & BOUNDARY     **
3860  REM      **  CONSTRAINT NODES. SAVE INTO DATA FILE "SUBST"  **
3870  HOME : VTAB 4: GOSUB 1440
3880  PRINT "          INPUT CONTROL DATA": PRINT
3890  PRINT "           IN SUBST. LEV. ";IM: PRINT : GOSUB 1440: GOSUB 1430
3900  HOME : PRINT : GOSUB 1440
3910  VTAB 4: INPUT "   NUM. OF SUBSTRUCTURE          = ";NB: PRINT : GOSUB 1440
3920  IF NB > BN THEN 3910
3930  GOSUB 1400: IF B$ = "N" THEN 3910
3940  FOR I = 1 TO NB STEP 14:K = I + 13: IF K > NB THEN K = NB
3950  HOME : PRINT : GOSUB 1440
3960  VTAB 4: PRINT "   NUM. OF SUBSTRUCTURE          = ";NB: PRINT : GOSUB 1440
3970  LET L = 7: FOR J = I TO K:L = L + 1
3980  VTAB L: INPUT "NUM. OF ELEMENTS IN SUBST.       = ";FS(J): NEXT J
3990  GOSUB 1400: IF B$ = "N" THEN 3970
4000  NEXT I
4010  IF IM > 1 THEN 4040
4020  FOR I = 1 TO NB:FT(I) = FS(I): NEXT I
4030  REM      **  LOOP OVER EACH SUBSTRUCTURE  **
4040  FOR IT = 1 TO NB: IF FS(IT) < 0 THEN 4610
4050  HOME : VTAB 4: GOSUB 1440
4060  PRINT "          INPUT CONTROL DATA": PRINT
4070  PRINT "           IN SUBST. NO.  ";IT: PRINT : GOSUB 1440: GOSUB 1430
4080  HOME : PRINT : GOSUB 1440
4090  VTAB 4: INPUT "   NUM. OF SUBST. NODES           = ";JS
4100  VTAB 6: INPUT "   NUM. OF BOUN. CON. NODES       = ";NP: PRINT : GOSUB 1440
4110  IF JS > JZ OR NP > NS THEN 4090
4120  GOSUB 1400: IF B$ = "N" THEN 4090
4130  IF JS = 0 THEN 4230
4140  FOR I = 1 TO JS STEP 15:K = I + 14: IF K > JS THEN K = JS
4150  HOME : PRINT : GOSUB 1440
4160  VTAB 4: PRINT "   NUM. OF SUBST. NODES    = ";JS: PRINT : GOSUB 1440
4170  LET L = 7: FOR J = I TO K:L = L + 1
4180  VTAB L: INPUT "SUBST. NODE NUMBER          = ";SM(J)
4190  IF SM(J) < 1 OR SM(J) > PT THEN 4180
4200  NEXT J
4210  GOSUB 1400: IF B$ = "N" THEN 4170
4220  NEXT I
4230  IF NP = 0 THEN 4410
4240  HOME : PRINT : GOSUB 1440
4250  PRINT "  NUM. OF BOUN. CON. NODES     = ";NP: PRINT
4260  PRINT "    ( Free = 0   :   Fixed = 1 )": PRINT : GOSUB 1440: PRINT
4270  FOR I = 1 TO NP: IF BB = 1 THEN 4350
4280  IF I > 1 THEN 4320
4290  VTAB 11: PRINT "NODE NUMBER WITH BOUN. CONSTRAIN  = "
4300  VTAB 13: PRINT "B. C. IN X-Coor.                  = "
4310  VTAB 15: PRINT "B. C. IN Y-Coor.                  = ": GOTO 4350
```

```
4320  VTAB 11: PRINT "NODE NUMBER WITH BOUN. CONSTRAIN  = ";J
4330  VTAB 13: PRINT "B. C. IN X-Coor.                  = ";K
4340  VTAB 15: PRINT "B. C. IN Y-Coor.                  = ";L
4350  VTAB 11: INPUT "NODE NUMBER WITH BOUN. CONSTRAIN  = ";CD(I,1):J = CD(I,1)
4360  IF J < 1 OR J > PT THEN 4350
4370  VTAB 13: INPUT "B. C. IN X-Coor.                  = ";CD(I,2):K = CD(I,2)
4380  VTAB 15: INPUT "B. C. IN Y-Coor.                  = ";CD(I,3):L = CD(I,3)
4390  GOSUB 1400: IF B$ = "N" THEN 4350
4400  NEXT I
4410  PRINT : PRINT D$;"OPEN SUBST,L";YT:ZT = ZT + 1: PRINT D$;"WRITE SUBST,R";ZT
4420  PRINT NB: PRINT JS: PRINT NP
4430  FOR I = 1 TO NB: PRINT FS(I): NEXT I
4440  IF JS = 0 THEN 4460
4450  FOR I = 1 TO JS: PRINT SM(I): NEXT I
4460  IF NP = 0 THEN 4500
4470  FOR I = 1 TO NP
4480  FOR J = 1 TO 3: PRINT CD(I,J): NEXT J
4490  NEXT I
4500  PRINT D$;"CLOSE SUBST"
4510  PRINT D$;"OPEN FRONT,L";YT: PRINT D$;"WRITE FRONT,R";ZT
4520  PRINT NB: PRINT JS: PRINT NP
4530  FOR I = 1 TO NB: PRINT FS(I): NEXT I
4540  IF JS = 0 THEN 4560
4550  FOR I = 1 TO JS: PRINT SM(I): NEXT I
4560  IF NP = 0 THEN 4600
4570  FOR I = 1 TO NP
4580  FOR J = 1 TO 3: PRINT CD(I,J): NEXT J
4590  NEXT I
4600  PRINT D$;"CLOSE FRONT"
4610  NEXT IT
4620  NEXT IM
4630  REM     **  INPUT MATERIAL PROPERTY. SAVE INTO DATA FILE "MAT"  **
4640  HOME : FOR I = 1 TO DM: VTAB 2: GOSUB 1440
4650  VTAB 4: PRINT "    MATERIAL TYPE   ..... ";I: PRINT : GOSUB 1440: IF BB = 1 THEN
   4750
4660  IF I > 1 THEN 4710
4670  VTAB 9: PRINT "YOUNG'S MODULUS         = "
4680  VTAB 11: PRINT "POISSON'S RATIO         = "
4690  VTAB 13: PRINT "MASS DENSITY            = "
4700  VTAB 15: PRINT "COEF. OF TEMPERATURE    = ": GOTO 4750
4710  VTAB 9: PRINT "YOUNG'S MODULUS         = ";J
4720  VTAB 11: PRINT "POISSON'S RATIO         = ";K
4730  VTAB 13: PRINT "MASS DENSITY            = ";L
4740  VTAB 15: PRINT "COEF. OF TEMPERATURE    = ";M
4750  VTAB 9: INPUT "YOUNG'S MODULUS         = ";YM(I):J = YM(I)
4760  VTAB 11: INPUT "POISSON'S RATIO         = ";PR(I):K = PR(I)
4770  VTAB 13: INPUT "MASS DENSITY            = ";DEN(I):L = DEN(I)
4780  VTAB 15: INPUT "COEF. OF TEMPERATURE    = ";AP(I):M = AP(I)
4790  GOSUB 1400: IF B$ = "N" THEN 4750
4800  NEXT I
```

```
4810  PRINT D$;"OPEN MAT,D2": PRINT D$;"DELETE MAT"
4820  PRINT D$;"OPEN MAT": PRINT D$;"WRITE MAT"
4830  PRINT DM: PRINT RT
4840  FOR I = 1 TO DM: PRINT YM(I): PRINT PR(I): PRINT DEN(I): PRINT AP(I): NEXT I
4850  PRINT D$;"CLOSE MAT"
4860  RETURN
```

```
1000  HOME :D$ =  CHR$ (4): VTAB 4: GOTO 1040: REM      ** PLOT **
1010  PRINT "----------": PRINT
1020  PRINT "   PLOT   ": PRINT
1030  PRINT "----------": PRINT : RETURN
1040  GOSUB 1010: PRINT D$;"OPEN DAT1,D2": PRINT D$;"READ DAT1"
1050  INPUT U,PT,EL,U,U,U,U,U,U,U,U,U,U,U,U,U,U,BN,U
1060  PRINT D$;"CLOSE DAT1"
1070  DIM X(PT),Y(PT),II(EL),IJ(EL),IK(EL),IL(EL)
1080  PRINT D$;"OPEN SUM,D2": PRINT D$;"READ SUM"
1090  INPUT E3,XR,DL,DN: IF DL = 0 THEN 1130
1100  FOR I = 1 TO DL: INPUT U
1110  FOR J = 1 TO DN: INPUT U,U,U: NEXT J
1120  NEXT I
1130  FOR I = 1 TO PT: INPUT X(I),Y(I),U: NEXT I
1140  FOR I = 1 TO E3: INPUT U,U,U: NEXT I
1150  FOR I = 1 TO BN: INPUT U,U: NEXT I
1160  FOR I = 1 TO EL: INPUT U,U,II(I),IJ(I),IK(I),IL(I): NEXT I
1170  PRINT D$;"CLOSE SUM"
1180  HOME : GOSUB 1010: VTAB 24: PRINT  TAB( 4)"Do you PLOT on monitor ?(Y/N) ";: INPUT
    "";B$: IF B$ <  > "Y" AND B$ <  > "N" THEN 1180
1190  IF B$ = "N" THEN 1210
1200  GOSUB 1260: TEXT
1210  HOME : GOSUB 1010: VTAB 24: PRINT  TAB( 4)"Do you PLOT on printer ?(Y/N) ";: INPUT
    "";B$: IF B$ <  > "Y" AND B$ <  > "N" THEN 1210
1220  IF B$ = "N" THEN 1250
1230  PRINT D$;"PR#1": GOSUB 1260
1240  PRINT  CHR$ (17): PRINT D$;"PR#0": TEXT : HOME
1250  PRINT D$;"RUN OPTIONS,D1"
1260  HOME : HCOLOR= 7: HGR :P = 0:Q = 0:R = 0:S = 0
1270  FOR I = 1 TO PT
1280  IF X(I) > P THEN P = X(I)
1290  IF X(I) < Q THEN Q = X(I)
1300  IF Y(I) > R THEN R = Y(I)
1310  IF Y(I) < S THEN S = Y(I)
1320  NEXT I:X = P - Q:Y = R - S:Z = X: IF Y > Z THEN Z = Y
1330  LET H = 150 / Z:P = 0:R = 0
1340  IF Q < 0 THEN P =  - Q
1350  IF S < 0 THEN R =  - S
1360  FOR N = 1 TO EL:I = II(N):J = IJ(N):K = IK(N):L = IL(N)
1370  LET X1 = (X(I) + P) * H + 20:X2 = (X(J) + P) * H + 20:X3 = (X(K) + P) * H + 2
    0:X4 = (X(L) + P) * H + 20
1380  LET Y1 = 155 - (Y(I) + R) * H:Y2 = 155 - (Y(J) + R) * H:Y3 = 155 - (Y(K) + R)
     * H:Y4 = 155 - (Y(L) + R) * H
1390  HPLOT X1,Y1 TO X2,Y2 TO X3,Y3 TO X4,Y4 TO X1,Y1: NEXT N
1400  RETURN
```

```
1000  HOME :D$ =  CHR$ (4): VTAB 4: GOTO 1040: REM      ** PRINT DATA **
1010  PRINT "-----------------": PRINT
1020  PRINT "   PRINT  DATA   ": PRINT
1030  PRINT "-----------------": PRINT : RETURN
1040  GOSUB 1010: PRINT D$;"OPEN DAT1,D2": PRINT D$;"READ DAT1"
1050  INPUT XC,PT,EL,MN,NF,MI,PS,NS,KSUB,JZ,ZT,ZQ,ZA,E3,YT,MT,LU,BN,TE
1060  PRINT D$;"CLOSE DAT1":LV = 1:M1 = 2 * MN: IF MN = 4 THEN 1080
1070  DIM NI(EL),NJ(EL),NK(EL),NL(EL)
1080  DIM IQ(MN),LO(E3,LV),R1(E3,LV),R2(E3,LV),FS(BN),CD(NS,3),SM(JZ),X(PT),Y(PT),F
      T(BN)
1090  DIM T(PT),IX(EL),JQ(MN),VV(BN),II(EL),IJ(EL),IK(EL),IL(EL),THC(EL),SN(NS,2)
1100  PRINT D$;"OPEN MAT,D2": PRINT D$;"READ MAT"
1110  INPUT DM,RT
1120  DIM YM(DM),PR(DM),DEN(DM),AP(DM)
1130  FOR I = 1 TO DM: INPUT YM(I),PR(I),DEN(I),AP(I): NEXT I
1140  PRINT D$;"CLOSE MAT"
1150  PRINT D$;"OPEN SUM,D2": PRINT D$;"READ SUM"
1160  INPUT E3,XR,DL,DN: IF DL = 0 THEN 1210
1170  DIM ELN(DL),DISN(DL,3),R3(DL,3),R4(DL,3)
1180  FOR I = 1 TO DL: INPUT ELN(I)
1190  FOR J = 1 TO DN: INPUT DISN(I,J),R3(I,J),R4(I,J): NEXT J
1200  NEXT I
1210  FOR L = 1 TO PT: INPUT X(L),Y(L),T(L): NEXT L
1220  FOR I = 1 TO E3: INPUT LO(I,1),R1(I,1),R2(I,1): NEXT I
1230  FOR I = 1 TO BN: INPUT VV(I),FT(I): NEXT I
1240  FOR I = 1 TO EL: INPUT IX(I),THC(I),II(I),IJ(I),IK(I),IL(I): NEXT I: IF MN =
      4 THEN 1260
1250  FOR I = 1 TO EL: INPUT NI(I),NJ(I),NK(I),NL(I): NEXT I
1260  PRINT D$;"CLOSE SUM"
1270  PRINT D$;"OPEN SUPSET,D2": PRINT D$;"READ SUPSET"
1280  FOR I = 1 TO NS: INPUT SN(I,1),SN(I,2): NEXT I
1290  PRINT D$;"CLOSE SUPSET"
1300  FOR I = 1 TO PT
1310  LET X(I) =  INT (X(I) * 1000 + 0.5) / 1000:Y(I) =  INT (Y(I) * 1000 + 0.5) /
      1000:T(I) =  INT (T(I) * 1000 + 0.5) / 1000: NEXT I
1320  VTAB 23: INPUT "Do you display DATA on monitor?(Y/N) ";B$: IF B$ < > "Y" AND
      B$ < > "N" THEN 1320
1330  GOSUB 1590: IF B$ = "N" THEN 1350
1340  HOME : GOSUB 1010: GOSUB 1400: HOME : GOSUB 1010
1350  VTAB 23: INPUT "Do you print DATA on printer ?(Y/N) ";B$: IF B$ < > "Y" AND
      B$ < > "N" THEN 1350
1360  IF B$ = "N" THEN 1390
1370  PRINT D$;"PR#1": POKE 1657,80:CP = 1: GOSUB 1010: GOSUB 1400
1380  PRINT D$;"PR#0"
1390  PRINT D$;"RUN OPTIONS,D1"
1400  VTAB 8: PRINT "LARGEST NODE NUMBER              = ";PT
1410  VTAB 10: PRINT "NUMBER OF FINITE ELEMENTS        = ";EL
1420  VTAB 12: PRINT "NUMBER OF FINITE ELEMENTS ON DISK= ";NF
1430  VTAB 14: PRINT "NUM. OF NODES/ELEMENT IN LEVEL 1 = ";MN
1440  VTAB 16: PRINT "MAX. NUM. OF NODES IN SUBST.     = ";JZ
1450  VTAB 18: PRINT "MAX. NUM. OF ELEMENT IN ANY LEVEL= ";PS
1460  VTAB 20: PRINT "MAX. NUM. OF BOUND. CON. NODES   = ";NS
```

```
1470  IF CP = 0 THEN  GOSUB 1600: HOME : GOSUB 1010
1480  VTAB 8: PRINT "NUM. OF SUBSTRUCTURES IN LEVEL 1 = ";BN
1490  VTAB 10: PRINT "NUM. OF SUBSTRUCTURE LEVELS      = ";KSUB
1500  VTAB 12: PRINT "NUMBER OF MATERIAL TYPES         = ";DM
1510  VTAB 14: PRINT "ELEMENT TYPE  (1 OR 2).....     = ";TE
1520  VTAB 16: PRINT "MAX. NUM. OF THE SAME SUBST.     = ";LU
1530  VTAB 18: PRINT "MAX. NUM. OF NODAL POINT LOADS   = ";E3 - 1
1540  VTAB 20: PRINT "REFERENCE TEMPERATURE            = ";RT
1550  LET Y$ = "Y": IF MT = 1 THEN Y$ = "N"
1560  VTAB 22: PRINT "THE ELEMENTS ARE THE SAME TYPE   = ";Y$
1570  IF CP = 0 THEN  GOSUB 1600
1580  GOTO 1620
1590  VTAB 23: PRINT "                                              ": RETURN
1600  VTAB 24: PRINT "Press RETURN key to continue .... ";: INPUT "  ";B$: RETURN
1610  PRINT  TAB( 1)"----------------------------------------": RETURN
1620  LET ZS =  - 1: GOSUB 2170
1630  HOME :J = 9
1640  FOR I = 1 TO PT:J = J + 1: IF J > 22 AND CP = 0 THEN  GOSUB 1600:J = 10
1650  IF J > 22 AND CP = 1 THEN J = 10: GOTO 1700
1660  IF J <  > 10 THEN 1700
1670  HOME : GOSUB 1610
1680  VTAB 3: PRINT " GLOBAL NODAL COORDINATE & TEMPERATURE": PRINT : GOSUB 1610
1690  VTAB 7: PRINT "NODE     X-Coor.     Y-Coor.    TEMP.": PRINT "----     ------
   -     -------    -----": PRINT
1700  VTAB J: PRINT  SPC( 4 -  LEN ( STR$ (I)));I;: PRINT  SPC( 11 -  LEN ( STR$ (X
   (I)));X(I);: PRINT  SPC( 12 -  LEN ( STR$ (Y(I)));Y(I);: PRINT  SPC( 8 -  LEN
   ( STR$ (T(I)));T(I): NEXT I
1710  IF CP = 0 THEN  GOSUB 1600
1720  HOME :J = 9:K = 1: IF MN = 8 THEN K = 2
1730  FOR I = 1 TO EL:J = J + K: IF J > 21 AND CP = 0 THEN  GOSUB 1600:J = 9 + K
1740  IF J > 21 AND CP = 1 THEN J = 9 + K: GOTO 1790
1750  IF J <  > 9 + K THEN 1790
1760  HOME : GOSUB 1610
1770  VTAB 3: PRINT  TAB( 8)"ELEMENT NODES & TYPE": PRINT : GOSUB 1610
1780  VTAB 7: PRINT "ELE. No.  I    J    K    L   MAT.  THC.": PRINT "-------- ---
    ---  ---  ---  ----  ----": PRINT
1790  VTAB J + 1 - K: PRINT  SPC( 6 -  LEN ( STR$ (I)));I;: PRINT  SPC( 6 -  LEN ( STR$
   (II(I)));II(I);: PRINT  SPC( 5 -  LEN ( STR$ (IJ(I)));IJ(I);: PRINT  SPC( 5 -
    LEN ( STR$ (IK(I)));IK(I);
1800  PRINT  SPC( 5 -  LEN ( STR$ (IL(I)));IL(I);: PRINT  SPC( 4 -  LEN ( STR$ (IX
   (I)));IX(I);: PRINT  SPC( 6 -  LEN ( STR$ (THC(I)));THC(I): IF MN = 4 THEN 18
   20
1810  VTAB J: PRINT  SPC( 12 -  LEN ( STR$ (NI(I)));NI(I);: PRINT  SPC( 5 -  LEN (
    STR$ (NJ(I)));NJ(I);: PRINT  SPC( 5 -  LEN ( STR$ (NK(I)));NK(I);: PRINT  SPC(
   5 -  LEN ( STR$ (NL(I)));NL(I)
1820  NEXT I
1830  IF CP = 0 THEN  GOSUB 1600
1840  HOME :J = 9: IF E3 = 1 THEN 1930
1850  FOR I = 2 TO E3:J = J + 1: IF J > 21 AND CP = 0 THEN  GOSUB 1600:J = 10
1860  IF J > 21 AND CP = 1 THEN J = 10: GOTO 1910
1870  IF J <  > 10 THEN 1910
```

```
1880  HOME : GOSUB 1610
1890  VTAB 3: PRINT "   NODAL POINT LOADS IN GLOBAL Coor.": PRINT : GOSUB 1610
1900  VTAB 7: PRINT "   NODE      FX      FY": PRINT "   ----      ----    ----": PRINT

1910  VTAB J: PRINT  SPC( 6 -  LEN ( STR$ (LO(I,1)));LO(I,1);: PRINT  SPC( 9 -  LEN
    ( STR$ (R1(I,1)));R1(I,1);: PRINT  SPC( 8 -  LEN ( STR$ (R2(I,1)));R2(I,1): NEXT
    I
1920  IF CP = 0 THEN  GOSUB 1600
1930  IF DL = 0 THEN 2080
1940  HOME :K = 7
1950  FOR I = 1 TO DL:K = K + 4: IF K > 19 AND CP = 0 THEN  GOSUB 1600:K = 11
1960  IF K > 19 AND CP = 1 THEN K = 11: GOTO 2030
1970  IF K <  > 11 THEN 2030
1980  HOME : GOSUB 1610
1990  PRINT "   NUM. OF DISTRIBUTED LOADS    = ";DL: PRINT : GOSUB 1610
2000  VTAB 7: PRINT "DIST. LOAD  ELE    NODE No.&DIST. LOAD"
2010  VTAB 8: PRINT "    No.     No.      I      J      K  "
2020  VTAB 9: PRINT "----------  ---    -------------------"
2030  VTAB K: PRINT  SPC( 6 -  LEN ( STR$ (I)));I;: PRINT  SPC( 9 -  LEN ( STR$ (EL
    N(I)));ELN(I);: FOR J = 1 TO DN: PRINT  SPC( 7 -  LEN ( STR$ (DISN(I,J)));DIS
    N(I,J);: NEXT J: PRINT
2040  VTAB K + 1: PRINT "NORMAL     LOAD";: FOR J = 1 TO DN: PRINT  SPC( 7 -  LEN (
     STR$ (R3(I,J)));R3(I,J);: NEXT J: PRINT
2050  VTAB K + 2: PRINT "TANGENTIAL LOAD";: FOR J = 1 TO DN: PRINT  SPC( 7 -  LEN (
     STR$ (R4(I,J)));R4(I,J);: NEXT J: PRINT
2060  NEXT I
2070  IF CP = 0 THEN  GOSUB 1600
2080  HOME :J = 1: GOSUB 1610
2090  FOR IT = 1 TO BN
2100  LET LB = FT(IT):VV = VV(IT): IF LB < 0 THEN 2140
2110  LET J = J + 2: IF J > 21 AND CP = 0 THEN  GOSUB 1600:J = 3: HOME : GOSUB 1610

2120  IF J > 21 AND CP = 1 THEN J = 3
2130  VTAB J: PRINT "NUM. OF THE SAME SUBST. No.";IT;"  = ";VV
2140  NEXT IT
2150  IF CP = 0 THEN  GOSUB 1600
2160  RETURN
2170  FOR IM = 1 TO KSUB: GOTO 2280
2180  PRINT D$;"OPEN SUBST,L";YT;",D2":ZS = ZS + 1: PRINT D$;"READ SUBST,R";ZS
2190  INPUT NB,JS,NP
2200  FOR I = 1 TO NB: INPUT FS(I): NEXT I
2210  IF JS = 0 THEN 2230
2220  FOR I = 1 TO JS: INPUT SM(I): NEXT I
2230  IF NP = 0 THEN 2270
2240  FOR I = 1 TO NP
2250  FOR J = 1 TO 3: INPUT CD(I,J): NEXT J
2260  NEXT I
2270  PRINT D$;"CLOSE SUBST": RETURN
2280  GOSUB 2180
2290  HOME :J = 7
```

```
2300  FOR I = 1 TO NB:J = J + 1: IF J > 21 AND CP = 0 THEN  GOSUB 1600:J = 8
2310  IF J > 21 AND CP = 1 THEN J = 8: GOTO 2350
2320  IF J <  > 8 THEN 2350
2330  HOME : PRINT : GOSUB 1610
2340  VTAB 4: PRINT "NUM. OF SUBST. IN LEV. ";IM;"         = ";NB: PRINT : GOSUB 16
   10
2350  VTAB J: PRINT "NUM. OF ELEMENTS IN SUBST.       = ";FS(I): NEXT I
2360  IF CP = 0 THEN  GOSUB 1600
2370  FOR IT = 1 TO NB
2380  IF FS(IT) > 0 AND IT > 1 THEN  GOSUB 2180
2390  IF FS(IT) < 0 THEN 2600
2400  HOME : PRINT : GOSUB 1610
2410  PRINT : PRINT "NUM. OF NODES OF SUBSTRUCTURE No.";IT;" = ";JS
2420  PRINT : PRINT "NUM. OF BOUN. CON. NODES IN  LEV.";IM;" = ";NP
2430  IF CP = 0 THEN  GOSUB 1600
2440  IF JS = 0 THEN 2490
2450  HOME : PRINT : GOSUB 1610:J = 3
2460  FOR I = 1 TO JS:J = J + 1: IF J > 21 AND CP = 0 THEN  GOSUB 1600:J = 4: HOME
   : PRINT : GOSUB 1610
2470  IF J > 21 AND CP = 1 THEN J = 4
2480  VTAB J: PRINT "SUBST. NODE NUMBER OF NODE ";: PRINT  SPC( 4 -  LEN ( STR$ (I)
   ));I;"  = ";SM(I): NEXT I
2490  IF NP = 0 THEN 2590
2500  HOME :J = 9
2510  FOR I = 1 TO NP:J = J + 1: IF J > 21 AND CP = 0 THEN  GOSUB 1600:J = 10
2520  IF J > 21 AND CP = 1 THEN J = 10: GOTO 2570
2530  IF J <  > 10 THEN 2570
2540  HOME : PRINT : GOSUB 1610
2550  VTAB 3: PRINT  TAB( 5)"BOUNDARY CONSTRAIN AT JOINT": PRINT : GOSUB 1610
2560  VTAB 7: PRINT "  NODE    X    Y    X-Displ.  Y-Displ.": PRINT "  ----   ---
     ---   -------- --------"
2570  VTAB J: PRINT  SPC( 6 -  LEN ( STR$ (CD(I,1))));CD(I,1);: PRINT  SPC( 6 -  LEN
   ( STR$ (CD(I,2))));CD(I,2);: PRINT  SPC( 5 -  LEN ( STR$ (CD(I,3))));CD(I,3);
2580  PRINT  SPC( 7 -  LEN ( STR$ (SN(I,1))));SN(I,1);: PRINT  SPC( 10 -  LEN ( STR$
   (SN(I,2))));SN(I,2): NEXT I
2590  IF CP = 0 THEN  GOSUB 1600
2600  NEXT IT
2610  NEXT IM
2620  FOR I = 1 TO DM: HOME : GOSUB 1610
2630  PRINT  TAB( 3)"MATERIAL TYPE  ..... ";I: PRINT : GOSUB 1610
2640  PRINT "YOUNG'S MODULUS        = ";YM(I)
2650  PRINT "POISSON'S RATIO        = ";PR(I)
2660  PRINT "MASS DENSITY           = ";DEN(I)
2670  PRINT "COEF. OF TEMPERATURE   = ";AP(I)
2680  IF CP = 0 THEN  GOSUB 1600
2690  NEXT I
2700  RETURN
```

```
1000  ONERR  GOTO 1670
1010  HOME :D$ =  CHR$ (4): GOTO 1050: REM    **  CHANGE DATA  **
1020  PRINT "-----------------": PRINT
1030  PRINT "   CHANGE DATA   ": PRINT
1040  PRINT "-----------------": PRINT : RETURN
1050  GOSUB 1020: PRINT D$;"OPEN DAT1,D2": PRINT D$;"READ DAT1"
1060  INPUT XC,PT,EL,MN,NF,MI,PS,NS,KSUB,JZ,ZT,Z0,ZA,E3,YT,MT,LU,BN,TE
1070  PRINT D$;"CLOSE DAT1"
1080  PRINT D$;"OPEN MAT,D2": PRINT D$;"READ MAT"
1090  INPUT DM,RT
1100  DIM YM(15),PR(15),DEN(15),AP(15)
1110  FOR I = 1 TO DM: INPUT YM(I),PR(I),DEN(I),AP(I): NEXT I
1120  PRINT D$;"CLOSE MAT": PRINT
1130  LET U1 = PT:V1 = PT:U2 = EL:V2 = EL:U3 = JZ:V3 = JZ:U4 = BN:V4 = BN:U5 = NS:V
    5 = NS:U6 = E3:V6 = E3:U7 = DM:V7 = DM:U8 = MN:V8 = MN
1140  VTAB 8: PRINT "LARGEST NODE NUMBER                = ";PT
1150  VTAB 10: PRINT "NUMBER OF FINITE ELEMENTS          = ";EL
1160  VTAB 12: PRINT "NUMBER OF FINITE ELEMENTS ON DISK= ";NF
1170  VTAB 14: PRINT "NUM. OF NODES/ELEMENT IN LEVEL 1 = ";MN
1180  VTAB 16: PRINT "MAX. NUM. OF NODES IN SUBST.       = ";JZ
1190  VTAB 18: PRINT "MAX. NUM. OF ELEMENT IN ANY LEVEL= ";PS
1200  VTAB 20: PRINT "MAX. NUM. OF BOUND. CON. NODES     = ";NS
1210  GOSUB 1630: IF B$ = "N" THEN 1290
1220  VTAB 8: INPUT "LARGEST NODE NUMBER                = ";PT
1230  VTAB 10: INPUT "NUMBER OF FINITE ELEMENTS          = ";EL
1240  VTAB 12: INPUT "NUMBER OF FINITE ELEMENTS ON DISK= ";NF
1250  VTAB 14: INPUT "NUM. OF NODES/ELEMENT IN LEVEL 1 = ";MN
1260  VTAB 16: INPUT "MAX. NUM. OF NODES IN SUBST        = ";JZ
1270  VTAB 18: INPUT "MAX. NUM. OF ELEMENT IN ANY LEVEL= ";PS
1280  VTAB 20: INPUT "MAX. NUM. OF BOUND. CON. NODES     = ";NS
1290  HOME : GOSUB 1020
1300  VTAB 8: PRINT "NUM. OF SUBSTRUCTURES IN LEVEL 1 = ";BN
1310  VTAB 10: PRINT "NUM. OF SUBSTRUCTURE LEVELS        = ";KSUB
1320  VTAB 12: PRINT "NUMBER OF MATERIAL TYPES           = ";DM
1330  VTAB 14: PRINT "ELEMENT TYPE  (1 OR 2).....       = ";TE
1340  VTAB 16: PRINT "MAX. NUM. OF THE SAME SUBST.       = ";LU
1350  VTAB 18: PRINT "MAX. NUM. OF NODAL POINT LOADS     = ";E3 - 1
1360  VTAB 20: PRINT "REFERENCE TEMPERATURE              = ";RT
1370  LET Y$ = "Y": IF MT = 1 THEN Y$ = "N"
1380  VTAB 22: PRINT "THE ELEMENTS ARE THE SAME TYPE     = ";Y$
1390  GOSUB 1630: IF B$ = "N" THEN 1490
1400  VTAB 8: INPUT "NUM. OF SUBSTRUCTURES IN LEVEL 1 = ";BN
1410  VTAB 10: INPUT "NUM. OF SUBSTRUCTURE LEVELS        = ";KSUB
1420  VTAB 12: INPUT "NUMBER OF MATERIAL TYPES           = ";DM
1430  VTAB 14: INPUT "ELEMENT TYPE  (1 OR 2).....       = ";TE
1440  VTAB 16: INPUT "MAX. NUM. OF THE SAME SUBST.       = ";LU
1450  VTAB 18: INPUT "MAX. NUM. OF NODAL POINT LOADS     = ";E3
1460  VTAB 20: INPUT "REFERENCE TEMPERATURE              = ";RT
1470  VTAB 22: INPUT "THE ELEMENTS ARE THE SAME TYPE     = ";Y$: IF Y$ <  > "Y" AND Y
    $ <  > "N" THEN 1470
```

```
1480  LET E3 = E3 + 1:MT = 1: IF Y$ = "Y" THEN MT = PS
1490  LET LV = 1:M1 = 2 * MN:X1 =  - 1:LY = 0
1500  LET ZO = 6 * JZ + 36:YR = (JZ + BN + 3 + (3 * NS)) * 6:ZA = ((MN * 2) * (MN *
    2 + 1) + MN * 4 * LU * MT) * 10 + 36: IF KSUB > 2 THEN ZO = 9 * JZ + 36
1510  IF U1 < PT THEN U1 = PT
1520  IF U2 < EL THEN U2 = EL
1530  IF U3 < JZ THEN U3 = JZ
1540  IF U4 < BN THEN U4 = BN
1550  IF U5 < NS THEN U5 = NS
1560  IF U6 < E3 THEN U6 = E3
1570  IF U8 < MN THEN U8 = MN
1580  DIM IQ(U8),LO(U6,LV),R1(U6,LV),R2(U6,LV),FS(U4),CD(U5,3),SM(U3),X(U1),Y(U1),F
    T(U4 + 1)
1590  DIM T(U1),IX(U2),JQ(U8),VV(U4),II(U2),IJ(U2),IK(U2),IL(U2),THC(U2),SN(U5,2)
1600  IF MN = 4 THEN 1700
1610  DIM NI(U2),NJ(U2),NK(U2),NL(U2)
1620  GOTO 1700
1630  VTAB 23: PRINT "                                                    "
1640  VTAB 23: INPUT "Do you want to change DATA ?(Y/N)  ";B$: IF B$ <  > "Y" AND B
    $ <  > "N" THEN 1640
1650  RETURN
1660  VTAB 23: PRINT "                                                    ": RETURN
1670  CALL  - 998: RESUME
1680  VTAB 23: INPUT "Press RETURN key to continue ..... ";B$: RETURN
1690  PRINT  TAB( 1)"----------------------------------------": RETURN
1700  PRINT D$;"OPEN SUM,D2": PRINT D$;"READ SUM"
1710  INPUT V6,XR,DL,DN: IF DL = 0 THEN 1760
1720  DIM ELN(DL),DISN(DL,3),R3(DL,3),R4(DL,3)
1730  FOR I = 1 TO DL: INPUT ELN(I)
1740  FOR J = 1 TO DN: INPUT DISN(I,J),R3(I,J),R4(I,J): NEXT J
1750  NEXT I
1760  FOR L = 1 TO V1: INPUT X(L),Y(L),T(L): NEXT L
1770  FOR I = 1 TO V6: INPUT LO(I,1),R1(I,1),R2(I,1): NEXT I
1780  FOR I = 1 TO V4: INPUT VV(I),FT(I): NEXT I
1790  FOR I = 1 TO V2: INPUT IX(I),THC(I),II(I),IJ(I),IK(I),IL(I): NEXT I: IF V8 =
    4 THEN 1810
1800  FOR I = 1 TO V2: INPUT NI(I),NJ(I),NK(I),NL(I): NEXT I
1810  PRINT D$;"CLOSE SUM"
1820  LET ZS =  - 1:ZR =  - 1: GOSUB 3540
1830  HOME : VTAB 7: GOSUB 1690
1840  PRINT " GLOBAL NODAL COORDINATE & TEMPERATURE": PRINT
1850  PRINT  TAB( 7)" YOU SEE FROM PRINT DATA": PRINT : GOSUB 1690
1860  GOSUB 1630:W = 0:H = 0: IF B$ = "N" THEN 2080
1870  HOME : VTAB 7: GOSUB 1690
1880  PRINT "    INPUT GLOBAL NODAL Coor. & TEMP.": PRINT
1890  PRINT "     FOR MISTAKE NODES & NODE No. ";PT: PRINT : GOSUB 1690: GOSUB 1680
```

```
1900  HOME : PRINT : GOSUB 1690
1910  VTAB 4: PRINT " GLOBAL NODAL COORDINATE & TEMPERATURE": PRINT : GOSUB 1690
1920  VTAB 9: PRINT "NODE No.              =              "
1930  VTAB 11: PRINT "X-Coor.              =              "
1940  VTAB 13: PRINT "Y-Coor.              =              "
1950  VTAB 15: PRINT "TEMPERATURE          =              "
1960  VTAB 9: INPUT "NODE No.             = ";I: IF I < 1 OR I > PT THEN 1960
1970  VTAB 11: PRINT "X-Coor.              = ";X(I)
1980  VTAB 13: PRINT "Y-Coor.              = ";Y(I)
1990  VTAB 15: PRINT "TEMPERATURE          = ";T(I)
2000  GOSUB 1630: IF B$ = "N" THEN 2040
2010  VTAB 11: INPUT "X-Coor.              = ";X(I)
2020  VTAB 13: INPUT "Y-Coor.              = ";Y(I)
2030  VTAB 15: INPUT "TEMPERATURE          = ";T(I)
2040  IF I < PT THEN 1920
2050  FOR I = 1 TO PT: IF W < X(I) THEN W = X(I)
2060  IF H < Y(I) THEN H = Y(I)
2070  NEXT I
2080  HOME : VTAB 7: GOSUB 1690
2090  VTAB 9: PRINT "   DO YOU CHANGE SUPPORT DISPLACEMENT"
2100  VTAB 11: PRINT "   (Y/N)  .....................   ": PRINT : GOSUB 1690
2110  VTAB 11: INPUT "   (Y/N)  .....................   ";B$: IF B$ <  > "Y" AND
    B$ <  > "N" THEN 2110
2120  IF B$ = "N" THEN 2300
2130  PRINT D$;"OPEN SUPSET,D2": PRINT D$;"READ SUPSET"
2140  FOR I = 1 TO V5: INPUT SN(I,1),SN(I,2): NEXT I
2150  PRINT D$;"CLOSE SUPSET"
2160  HOME : PRINT : GOSUB 1690
2170  VTAB 4: PRINT "        SUPPORT DISPLACEMENT": PRINT : GOSUB 1690
2180  FOR I = 1 TO NS
2190  VTAB 9: PRINT "NODE No.                  = ";CD(I,1)
2200  VTAB 11: PRINT "DISPLACEMENT IN X-Coor.   = ";SN(I,1)
2210  VTAB 13: PRINT "DISPLACEMENT IN Y-Coor.   = ";SN(I,2)
2220  GOSUB 1630: IF B$ = "N" THEN 2250
2230  VTAB 11: INPUT "DISPLACEMENT IN X-Coor.   = ";SN(I1,1)
2240  VTAB 13: INPUT "DISPLACEMENT IN Y-Coor.   = ";SN(I1,2)
2250  NEXT I
2260  PRINT D$;"OPEN SUPSET,D2": PRINT D$;"DELETE SUPSET"
2270  PRINT D$;"OPEN SUPSET": PRINT D$;"WRITE SUPSET"
2280  FOR I = 1 TO NS: PRINT SN(I,1): PRINT SN(I,2): NEXT I
2290  PRINT D$;"CLOSE SUPSET"
2300  IF E3 > 1 THEN  GOSUB 2980
2310  LET DF = 0: GOSUB 3140
2320  HOME : VTAB 7: GOSUB 1690
2330  PRINT  TAB( 9)" ELEMENT NODES & TYPE": PRINT
2340  PRINT  TAB( 7)" YOU SEE FROM PRINT DATA": PRINT : GOSUB 1690
2350  GOSUB 1630: IF B$ = "N" THEN X1 = XR: GOTO 2740
2360  FOR IT = 1 TO BN:VV = 1: HOME : PRINT : GOSUB 1690
2370  LET LB = FT(IT): IF LB < 0 THEN 2730
```

```
2380  FOR I = IT TO BN:J = I + 1: IF J > BN OR FT(J) > 0 THEN I = BN
2390  IF FT(J) < 0 THEN VV = VV + 1
2400  NEXT I
2410  PRINT "NUM. OF THE SAME SUBST. No.";IT;"  = ";VV:VV(IT) = VV: PRINT : GOSUB 1
    690: GOSUB 1680
2420  HOME : FOR LZ = 1 TO LB:LY = LY + 1:IX = IX(LY):THC = THC(LY):X1 = X1 + 1:LW =
    X1
2430  IF X1 > XR THEN 2480
2440  PRINT D$;"OPEN ELE2,L";ZO: PRINT D$;"READ ELE2,R";LW
2450  INPUT NE
2460  FOR I = 1 TO NE: INPUT IQ(I): NEXT I
2470  PRINT D$;"CLOSE ELE2":NE = MN
2480  VTAB 2: GOSUB 1690
2490  VTAB 4: PRINT "NUM. OF NODE &NODE NUM. OF ELE. No.";LY: PRINT : GOSUB 1690
2500  VTAB 9: PRINT "NUMBER OF ELEMENT NODE        = ";NE
2510  FOR L = 1 TO NE: PRINT "NODE NUMBER FOR ELEMENT       = ";IQ(L): NEXT L
2520  PRINT "TYPE OF ELEMENT                = ";IX
2530  PRINT "THICKNESS OF ELEMENT           = ";THC
2540  GOSUB 1630: IF B$ = "N" THEN 2720
2550  FOR I = 1 TO NE:J = 9 + I
2560  VTAB J: INPUT "NODE NUMBER FOR ELEMENT       = ";IQ(I):JQ(I) = IQ(I): NEXT I
2570  INPUT "TYPE OF ELEMENT                = ";IX:IX(LY) = IX
2580  INPUT "THICKNESS OF ELEMENT           = ";THC:THC(LY) = THC
2590  PRINT D$;"OPEN ELE1,L";ZO:LW = X1: PRINT D$;"WRITE ELE1,R";LW
2600  PRINT NE
2610  FOR I = 1 TO NE: PRINT IQ(I): NEXT I
2620  PRINT D$;"CLOSE ELE1": IF KSUB < 3 THEN 2670
2630  PRINT D$;"OPEN ELE3,L";ZO: PRINT D$;"WRITE ELE3,R";LW
2640  PRINT NE
2650  FOR I = 1 TO NE: PRINT IQ(I): NEXT I
2660  PRINT D$;"CLOSE ELE3"
2670  FOR I = 1 TO VV: IF I = 1 THEN 2690
2680  FOR J = 1 TO NE:JQ(J) = JQ(J) - FT(IT + I - 1): NEXT J
2690  LET K = LY + (I - 1) * LB:II(K) = JQ(1):IJ(K) = JQ(2):IK(K) = JQ(3):IL(K) = J
    Q(4):IX(K) = IX:THC(K) = THC: IF MN = 4 THEN 2710
2700  LET NI(K) = JQ(5):NJ(K) = JQ(6):NK(K) = JQ(7):NL(K) = JQ(8)
2710  NEXT I
2720  NEXT LZ:LY = LY + (VV - 1) * LB
2730  NEXT IT:XC = (M1 * (M1 + 1)) / 2 + (M1 * LU * MT):ZA = ((MN * 2) * (MN * 2 +
    1) + MN * 4 * LU * MT) * 10 + 36: HOME
2740  PRINT D$;"OPEN LOCATE,D2": PRINT D$;"DELETE LOCATE"
2750  PRINT D$;"OPEN LOCATE": PRINT D$;"WRITE LOCATE"
2760  FOR I = 1 TO PT: PRINT X(I): NEXT I
2770  FOR I = 1 TO PT: PRINT Y(I): NEXT I
2780  FOR I = 1 TO PT: PRINT T(I): NEXT I
2790  FOR I = 1 TO EL: PRINT IX(I): NEXT I
2800  PRINT D$;"CLOSE LOCATE"
```

```
2810  PRINT D$;"OPEN SUM,D2": PRINT D$;"DELETE SUM"
2820  PRINT D$;"OPEN SUM": PRINT D$;"WRITE SUM"
2830  PRINT E3: PRINT X1: PRINT DF: PRINT DN: IF DF = 0 THEN 2870
2840  FOR I = 1 TO DF: PRINT EEN(I)
2850  FOR J = 1 TO DN: PRINT DP(I,J): PRINT R5(I,J): PRINT R6(I,J): NEXT J
2860  NEXT I
2870  FOR L = 1 TO PT: PRINT X(L): PRINT Y(L): PRINT T(L): NEXT L
2880  FOR I = 1 TO E3: PRINT LO(I,1): PRINT R1(I,1): PRINT R2(I,1): NEXT I
2890  FOR I = 1 TO BN: PRINT VV(I): PRINT FT(I): NEXT I
2900  FOR I = 1 TO EL: PRINT IX(I): PRINT THC(I): PRINT II(I): PRINT IJ(I): PRINT I
   K(I): PRINT IL(I): NEXT I: IF MN = 4 THEN 2920
2910  FOR I = 1 TO EL: PRINT NI(I): PRINT NJ(I): PRINT NK(I): PRINT NL(I): NEXT I
2920  PRINT D$;"CLOSE SUM"
2930  PRINT D$;"OPEN DAT1,D2": PRINT D$;"DELETE DAT1"
2940  PRINT D$;"OPEN DAT1": PRINT D$;"WRITE DAT1"
2950  PRINT XC: PRINT PT: PRINT EL: PRINT MN: PRINT NF: PRINT MI: PRINT PS: PRINT N
   S: PRINT KSUB: PRINT JZ: PRINT ZR: PRINT ZO: PRINT ZA: PRINT E3: PRINT YR: PRINT
   MT: PRINT LU: PRINT BN: PRINT TE
2960  PRINT D$;"CLOSE DAT1": POKE 216,0
2970  PRINT D$;"RUN OPTIONS,D1"
2980  HOME : VTAB 7: GOSUB 1690
2990  VTAB 9: PRINT "    DO YOU CHANGE NODAL POINT LOADS"
3000  VTAB 11: PRINT "    (Y/N) .....    ": PRINT : GOSUB 1690
3010  VTAB 11: INPUT "    (Y/N) .....    ";B$: IF B$ <  > "Y" AND B$
   <  > "N" THEN 3010
3020  IF B$ = "N" THEN  RETURN
3030  HOME : FOR I = 2 TO E3: VTAB 2: GOSUB 1690
3040  PRINT "NODAL POINT LOAD No. = ";I - 1: PRINT : GOSUB 1690
3050  VTAB 9: PRINT "NODE No.            = ";LO(I,1)
3060  VTAB 11: PRINT "LOADING IN X-Coor.   = ";R1(I,1)
3070  VTAB 13: PRINT "LOADING IN Y-Coor.   = ";R2(I,1)
3080  GOSUB 1630: IF B$ = "N" THEN 3120
3090  VTAB 9: INPUT "NODE No.            = ";LO(I,1)
3100  VTAB 11: INPUT "LOADING IN X-Coor.   = ";R1(I,1)
3110  VTAB 13: INPUT "LOADING IN Y-Coor.   = ";R2(I,1)
3120  NEXT I
3130  RETURN
3140  HOME : VTAB 7: GOSUB 1690
3150  VTAB 9: PRINT "    DO YOU CHANGE DISTRIBUTED LOADS"
3160  VTAB 11: PRINT "    (Y/N) .....    ": PRINT : GOSUB 1690
3170  VTAB 11: INPUT "    (Y/N) .....    ";B$: IF B$ <  > "Y" AND B$
   <  > "N" THEN 3170
3180  IF B$ = "Y" THEN 3230
3190  LET DF = DL: IF DL = 0 THEN  RETURN
3200  FOR I = 1 TO DF:EEN(I) = ELN(I)
3210  FOR J = 1 TO DN:DP(I,J) = DISN(I,J):R5(I,J) = R3(I,J):R6(I,J) = R4(I,J): NEXT
   J
3220  NEXT I: RETURN
3230  HOME : PRINT : GOSUB 1690
3240  VTAB 4: PRINT "NUM. OF DISTRIBUTED LOADS    = ";DL: PRINT : GOSUB 1690
```

```
3250  GOSUB 1630: IF B$ = "N" THEN 3300
3260  VTAB 4: INPUT "NUM. OF DISTRIBUTED LOADS    = ";P$
3270  LET DF = DL: IF P$ <  > "" THEN DF =  VAL (P$)
3280  VTAB 4: PRINT "NUM. OF DISTRIBUTED LOADS    = ";DF: IF DF = 0 THEN  RETURN
3290  GOTO 3310
3300  LET DF = DL: IF DL = 0 THEN  RETURN
3310  DIM EEN(DF),DP(DF,3),R5(DF,3),R6(DF,3):DN = 2: IF MN = 8 THEN DN = 3
3320  HOME : FOR I = 1 TO DF: VTAB 2: GOSUB 1690
3330  PRINT "DISTRIBUTED LOAD No.         = ";I: PRINT : GOSUB 1690: IF I > DL THEN
    3410
3340  VTAB 9: PRINT "ELEMENT No.                  = ";ELN(I):EEN(I) = ELN(I)
3350  FOR J = 1 TO DN:J1 = 7 + (J * 4)
3360  VTAB J1: PRINT "NODE No.                    = ";DISN(I,J):DP(I,J) = DISN(I,J
    )
3370  PRINT "NORMAL     DIST. LOAD        = ";R3(I,J):R5(I,J) = R3(I,J)
3380  PRINT "TANGENTIAL DIST. LOAD        = ";R4(I,J):R6(I,J) = R4(I,J): NEXT J
3390  GOSUB 1630: IF B$ = "N" THEN 3520
3400  GOTO 3470
3410  VTAB 9: PRINT "ELEMENT No.                  = ";0:EEN(I) = 0
3420  FOR J = 1 TO DN:J1 = 7 + (J * 4)
3430  VTAB J1: PRINT "NODE No.                    = ";0:DP(I,J) = 0
3440  PRINT "NORMAL     DIST. LOAD        = ";0:R5(I,J) = 0
3450  PRINT "TANGENTIAL DIST. LOAD        = ";0:R6(I,J) = 0: NEXT J
3460  GOSUB 1630: IF B$ = "N" THEN 3520
3470  VTAB 9: INPUT "ELEMENT No.                  = ";EEN(I): IF EEN(I) < 1 OR EEN(
    I) > EL THEN 3470
3480  FOR J = 1 TO DN:J1 = 7 + (J * 4)
3490  VTAB J1: INPUT "NODE No.                    = ";DP(I,J): IF DP(I,J) < 1 OR D
    P(I,J) > PT THEN 3490
3500  INPUT "NORMAL     DIST. LOAD        = ";R5(I,J)
3510  INPUT "TANGENTIAL DIST. LOAD        = ";R6(I,J): NEXT J
3520  NEXT I
3530  RETURN
3540  FOR IM = 1 TO KSUB: GOTO 3820
3550  LET ZS = ZS + 1: IF ZS > ZT THEN ZS = ZS - 1
3560  PRINT D$;"OPEN FRONT,L";YT;",D2": PRINT D$;"READ FRONT,R";ZS
3570  INPUT NB,JS,NP
3580  FOR I = 1 TO NB: INPUT FS(I): NEXT I
3590  IF JS = 0 THEN 3610
3600  FOR I = 1 TO JS: INPUT SM(I): NEXT I
3610  IF NP = 0 THEN 3650
3620  FOR I = 1 TO NP
3630  FOR J = 1 TO 3: INPUT CD(I,J): NEXT J
3640  NEXT I
3650  PRINT D$;"CLOSE FRONT"
3660  RETURN
3670  HOME : PRINT : GOSUB 1690
3680  PRINT "NUM. OF SUBST. IN LEV. ";IM;"        = ";NB: PRINT : GOSUB 1690: RETURN
```

```
3690  HOME : PRINT : GOSUB 1690
3700  VTAB 4: PRINT "NUM. OF SUBST. IN LEV. ";IM;"        = ";NB: PRINT : GOSUB 16
   90
3710  GOSUB 1630: IF B$ = "N" THEN 3740
3720  VTAB 4: INPUT "NUM. OF SUBST.                    = ";NB
3730  VTAB 4: PRINT "NUM. OF SUBST. IN LEV. ";IM;"         = ";NB
3740  FOR I = 1 TO NB:J = J + 1: IF J = 8 THEN  GOSUB 3670
3750  VTAB J: PRINT "NUM. OF ELEMENTS IN SUBST.       = ";FS(I): IF J < 22 AND I <
   NB THEN 3810
3760  GOSUB 1630: IF B$ = "N" THEN 3800
3770  LET I2 = J - 8:J = 7
3780  FOR I1 = I - I2 TO I:J = J + 1
3790  VTAB J: INPUT "NUM. OF ELEMENTS IN SUBST.       = ";FS(I1): NEXT I1
3800  LET J = 7
3810  NEXT I: RETURN
3820  GOSUB 3550:J = 7: GOSUB 3690
3830  IF IM > 1 THEN 3850
3840  FOR I = 1 TO NB:FT(I) = FS(I): NEXT I
3850  FOR IT = 1 TO NB
3860  IF FS(IT) > 0 AND IT > 1 THEN  GOSUB 3550:J = 7: GOSUB 3690
3870  IF FS(IT) < 0 THEN 4280
3880  HOME : PRINT : GOSUB 1690
3890  VTAB 4: PRINT "NUM. OF NODES OF SUBSTRUCTURE No.";IT;" = ";JS
3900  VTAB 6: PRINT "NUM. OF BOUN. CON. NODES IN  LEV.";IM;" = ";NP: PRINT : GOSUB
   1690
3910  GOSUB 1630: IF B$ = "N" THEN 3940
3920  VTAB 4: INPUT "NUM. OF NODES OF SUBSTRUCTURE      = ";JS
3930  VTAB 6: INPUT "NUM. OF BOUN. CON. NODES           = ";NP
3940  LET J = 7: IF JS = 0 THEN 4050
3950  FOR I = 1 TO JS:J = J + 1: IF J > 8 THEN 3980
3960  HOME : PRINT : GOSUB 1690
3970  VTAB 4: PRINT "   NUM. OF SUBST.NODES     = ";JS: PRINT : GOSUB 1690
3980  VTAB J: PRINT "SUBST. NODE NUMBER         = ";SM(I): IF J < 22 AND I < JS THEN
   4040
3990  GOSUB 1630: IF B$ = "N" THEN 4030
4000  LET I2 = J - 8:J = 7
4010  FOR I1 = I - I2 TO I:J = J + 1
4020  VTAB J: INPUT "SUBST. NODE NUMBER         = ";SM(I1): NEXT I1
4030  LET J = 7
4040  NEXT I
4050  IF NP = 0 THEN 4180
4060  HOME : PRINT : GOSUB 1690
4070  VTAB 4: PRINT "  NUM. OF BOUN. CON. NODES      = ";NP
4080  VTAB 6: PRINT "    ( Free = 0   :   Fixed = 1 )": PRINT : GOSUB 1690
4090  FOR I = 1 TO NP
4100  VTAB 11: PRINT "NODE NUMBER WITH BOUN. CONSTRAIN= ";CD(I,1)
4110  VTAB 13: PRINT "B. C. IN X-Coor.                  = ";CD(I,2)
4120  VTAB 15: PRINT "B. C. IN Y-Coor.                  = ";CD(I,3)
4130  GOSUB 1630: IF B$ = "N" THEN 4170
```

```
4140  VTAB 11: INPUT "NODE NUMBER WITH BOUN. CONSTRAIN= ";CD(I1,1)
4150  VTAB 13: INPUT "B. C. IN X-Coor.                  = ";CD(I1,2)
4160  VTAB 15: INPUT "B. C. IN Y-Coor.                  = ";CD(I1,3)
4170  NEXT I
4180  PRINT : PRINT D$;"OPEN SUBST,L";YR:ZR = ZR + 1: PRINT D$;"WRITE SUBST,R";ZR
4190  PRINT NB: PRINT JS: PRINT NP
4200  FOR I = 1 TO NB: PRINT FS(I): NEXT I
4210  IF JS = 0 THEN 4230
4220  FOR I = 1 TO JS: PRINT SM(I): NEXT I
4230  IF NP = 0 THEN 4270
4240  FOR I = 1 TO NP
4250  FOR J = 1 TO 3: PRINT CD(I,J): NEXT J
4260  NEXT I
4270  PRINT D$;"CLOSE SUBST"
4280  NEXT IT
4290  NEXT IM
4300  HOME : FOR I = 1 TO DM: VTAB 2: GOSUB 1690
4310  PRINT  TAB( 4)"MATERIAL TYPE   ..... ";I: PRINT : GOSUB 1690
4320  VTAB 9: PRINT "YOUNG'S MODULUS         = ";YM(I)
4330  VTAB 11: PRINT "POISSON'S RATIO         = ";PR(I)
4340  VTAB 13: PRINT "MASS DENSITY            = ";DEN(I)
4350  VTAB 15: PRINT "COEF. OF TEMPERATURE    = ";AP(I)
4360  GOSUB 1630: IF B$ = "N" THEN 4410
4370  VTAB 9: INPUT "YOUNG'S MODULUS         = ";YM(I)
4380  VTAB 11: INPUT "POISSON'S RATIO         = ";PR(I)
4390  VTAB 13: INPUT "MASS DENSITY            = ";DEN(I)
4400  VTAB 15: INPUT "COEF. OF TEMPERATURE    = ";AP(I)
4410  NEXT I
4420  PRINT D$;"OPEN MAT,D2": PRINT D$;"DELETE MAT"
4430  PRINT D$;"OPEN MAT": PRINT D$;"WRITE MAT"
4440  PRINT DM: PRINT RT
4450  FOR I = 1 TO DM: PRINT YM(I): PRINT PR(I): PRINT DEN(I): PRINT AP(I): NEXT I
4460  PRINT D$;"CLOSE MAT"
4470  RETURN
```

```
1000  HOME :D$ =  CHR$ (4): GOTO 1110: REM     ** STIFFNESS **
1010  PRINT "---------------": PRINT
1020  PRINT "   STIFFNESS   ": PRINT
1030  PRINT "---------------": PRINT : RETURN
1040  REM
1050  REM     ::::::::::::::::::::::::::::::::::::::::::::::::
1060  REM     :: THIS SUBPROGRAM PERFORMS STIFFNESS MATRIX ::
1070  REM     ::      AND LOAD VECTOR OF EACH ELEMENT      ::
1080  REM     ::      SAVE DATA INTO FILE " STIFF1 "       ::
1090  REM     ::::::::::::::::::::::::::::::::::::::::::::::::
1100  REM
1110  GOSUB 1010: PRINT D$;"OPEN DAT1,D2": PRINT D$;"READ DAT1"
1120  INPUT XC,PT,EL,MN,NF,MI,PS,NS,KSUB,JZ,ZT,ZQ,ZA,E3,YT,MT,LU,BN,TE
1130  PRINT D$;"CLOSE DAT1"
1140  PRINT D$;"OPEN MAT,D2": PRINT D$;"READ MAT"
1150  INPUT DM,RT
1160  DIM YM(DM),PR(DM),DEN(DM),AP(DM)
1170  FOR I = 1 TO DM: INPUT YM(I),PR(I),DEN(I),AP(I): NEXT I
1180  PRINT D$;"CLOSE MAT"
1190  LET LV = 1:M1 = 2 * MN:NG = 2:LY = 0:LC = 1:X1 =  - 1:X2 =  - 1:XK = 0:J5 = 0
    : IF MN = 4 THEN 1210
1200  DIM NI(EL),NJ(EL),NK(EL),NL(EL)
1210  DIM IQ(MN),S(M1,M1),ND(PT),LO(E3,LV),R1(E3,LV),R2(E3,LV),X(PT),Y(PT),P(M1,LU,
    MT),FT(BN),T(PT),IX(EL),JQ(MN),VV(BN),THC(EL)
1220  DIM PL(3,3),WGT(3,3),XL(MN),YL(MN),EX(MN),EY(MN),ES(MN),EN(MN),JAC(2,2),B(3,1
    7),E(3,3),XI(4),ET(4),BTE(M1,3),II(EL),IJ(EL),IK(EL),IL(EL),NXI(MN),NTE(MN)
1230  GOTO 1290
1240  VTAB 24: PRINT "  Do you accept these data ? (Y/N)  ";: INPUT " ";B$: RETURN

1250  PRINT  TAB( 1)"----------------------------------------": RETURN
1260  REM     ::::::::::::::::::::::::::::::::::::::::::::::::
1270  REM     :: READ POINTS AND WEIGHTS FOR GAUSS QUADRATURE ::
1280  REM     ::::::::::::::::::::::::::::::::::::::::::::::::
1290  FOR I = 1 TO 3
1300  FOR J = 1 TO 3: READ PL(I,J): NEXT J
1310  NEXT I
1320  DATA   0,-.577350269190,-.774596669241,0,.577350269190,0,0,0,.774596669241
1330  FOR I = 1 TO 3
1340  FOR J = 1 TO 3: READ WGT(I,J): NEXT J
1350  NEXT I
1360  DATA   2,1,.555555555556,0,1,.888888888889,0,0,.555555555556
1370  FOR I = 1 TO 4: READ ET(I): NEXT I
1380  DATA   -1,-1,1,1
1390  FOR I = 1 TO 4: READ XI(I): NEXT I
1400  DATA   -1,1,1,-1
1410  FOR I = 1 TO PT:ND(I) = 0: NEXT I
```

```
1420  PRINT D$;"OPEN SUM,D2": PRINT D$;"READ SUM"
1430  INPUT E3,XR,DL,DN: IF DL = 0 THEN 1480
1440  DIM ELN(DL),DISN(DL,3),R3(DL,3),R4(DL,3),R5(DL,3),R6(DL,3)
1450  FOR I = 1 TO DL: INPUT ELN(I)
1460  FOR J = 1 TO DN: INPUT DISN(I,J),R3(I,J),R4(I,J): NEXT J
1470  NEXT I
1480  FOR L = 1 TO PT: INPUT X(L),Y(L),T(L): NEXT L
1490  FOR I = 1 TO E3: INPUT LO(I,1),R1(I,1),R2(I,1): NEXT I
1500  FOR I = 1 TO BN: INPUT VV(I),FT(I): NEXT I
1510  FOR I = 1 TO EL: INPUT IX(I),THC(I),II(I),IJ(I),IK(I),IL(I): NEXT I: IF MN =
    4 THEN 1530
1520  FOR I = 1 TO EL: INPUT NI(I),NJ(I),NK(I),NL(I): NEXT I
1530  PRINT D$;"CLOSE SUM"
1540  PRINT D$;"OPEN STIFF1,D2": PRINT D$;"DELETE STIFF1": PRINT D$;"CLOSE STIFF1"
1550  PRINT D$;"OPEN ELE2,D2": PRINT D$;"DELETE ELE2": PRINT D$;"CLOSE ELE2"
1560  PRINT D$;"OPEN SUM,D2": PRINT D$;"DELETE SUM": PRINT D$;"CLOSE SUM"
1570  PRINT D$;"OPEN DIST,D2": PRINT D$;"DELETE DIST": PRINT D$;"CLOSE DIST"
1580  PRINT D$;"OPEN STRESS,D2": PRINT D$;"DELETE STRESS": PRINT D$;"CLOSE STRESS"
1590  IF E3 < 2 THEN 1650
1600  FOR I = 1 TO EL:J = II(I):ND(J) = ND(J) + 1:J = IJ(I):ND(J) = ND(J) + 1:J = I
    K(I):ND(J) = ND(J) + 1:J = IL(I):ND(J) = ND(J) + 1: NEXT I: IF MN = 4 THEN 1620

1610  FOR I = 1 TO EL:J = NI(I):ND(J) = ND(J) + 1:J = NJ(I):ND(J) = ND(J) + 1:J = N
    K(I):ND(J) = ND(J) + 1:J = NL(I):ND(J) = ND(J) + 1: NEXT I
1620  FOR J = 2 TO E3:I = LO(J,1): IF ND(I) = 0 THEN 1640
1630  LET R1(J,1) = R1(J,1) / ND(I):R2(J,1) = R2(J,1) / ND(I)
1640  NEXT J
1650  IF DL <  > 0 THEN  GOSUB 3400
1660  FOR I = 1 TO PT:ND(I) = 2: NEXT I
1670  REM     ::::::::::::::::::::::::::::::::::
1680  REM     :: LOOP OVER EACH SUBSTRUCTURE ::
1690  REM     ::::::::::::::::::::::::::::::::::
1700  FOR IT = 1 TO BN
1710  LET LB = FT(IT): IF LB < 0 THEN 3310
1720  LET VV = VV(IT)
1730  REM     ::::::::::::::::::::::::::::
1740  REM     :: LOOP OVER EACH ELEMENT ::
1750  REM     ::::::::::::::::::::::::::::
1760  FOR LZ = 1 TO LB:LY = LY + 1:IX = IX(LY):THC = THC(LY)
1770  PRINT D$;"OPEN ELE1,L";ZQ:X1 = X1 + 1:LW = X1: PRINT D$;"READ ELE1,R"LW
1780  INPUT NE
1790  FOR I = 1 TO NE: INPUT IQ(I): NEXT I
1800  PRINT D$;"CLOSE ELE1"
1810  REM     ::::::::::::::::::::::::::::::::::::::::::::::
1820  REM     :: FIND MATRIX OF ELASTIC STIFFNESS FOR ::
1830  REM     :: PLANE STRESS AND PLANE STRAIN        ::
1840  REM     ::::::::::::::::::::::::::::::::::::::::::::::
1850  IF TE = 2 THEN 1880
1860  LET F = YM(IX) / (1 + PR(IX)):G = F * PR(IX) / (1 - 2 * PR(IX)):H = F + G
```

```
1870  LET E(1,1) = H:E(1,2) = G:E(1,3) = 0:E(2,1) = G:E(2,2) = H:E(2,3) = 0:E(3,1) =
   0:E(3,2) = 0:E(3,3) = F / 2: GOTO 1900
1880  LET F = YM(IX) / (1 + PR(IX)):G = YM(IX) / (1 - PR(IX)):H = F * G / YM(IX)
1890  LET E(1,1) = H:E(1,2) = H * PR(IX):E(1,3) = 0:E(2,1) = H * PR(IX):E(2,2) = H:
   E(2,3) = 0:E(3,1) = 0:E(3,2) = 0:E(3,3) = F / 2
1900  HOME : GOSUB 1010: VTAB 8: GOSUB 1250
1910  PRINT "STIFFNESS & LOAD VECTOR OF ELE. No.";LY: PRINT : GOSUB 1250
1920  PRINT : PRINT  TAB( 16)"";: FLASH : PRINT "RUNNING": NORMAL
1930  FOR I = 1 TO NE:L = IQ(I):XL(I) = X(L):YL(I) = Y(L): NEXT I
1940  LET AREA = (XL(2) - XL(1)) * (YL(3) - YL(2)):M = 0
1950  FOR L = 1 TO NE:EX(L) = 0:EY(L) = 0:ES(L) = 0: NEXT L
1960  FOR I = 1 TO NE:J = IQ(I):M = M + ND(J): NEXT I: IF MT > 1 AND LZ < LB THEN 2
   030
1970  REM     ::::::::::::::::::::::::::::::::::::::::::::::::
1980  REM     :: INITIALIZE STIFFNESS MATRIX AND LOAD VECTOR ::
1990  REM     ::::::::::::::::::::::::::::::::::::::::::::::::
2000  FOR I = 1 TO M
2010  FOR J = I TO M:S(I,J) = 0: NEXT J
2020  NEXT I
2030  IF MT > 1 AND LZ > 1 THEN 2090
2040  FOR K = 1 TO MT
2050  FOR J = 1 TO LU
2060  FOR I = 1 TO M:P(I,J,K) = 0: NEXT I
2070  NEXT J
2080  NEXT K
2090  IF E3 <  = 1 THEN 2230
2100  REM     ::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
2110  REM     :: SAVE NODAL POINT LOADS INTO LOAD VECTORS OF ELEMENT ::
2120  REM     ::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
2130  FOR J1 = 2 TO E3
2140  FOR I = 1 TO NE:JQ(I) = IQ(I): NEXT I
2150  FOR J2 = 1 TO VV
2160  FOR J = 1 TO NE: IF J2 > 1 THEN JQ(J) = JQ(J) - FT(IT + J2 - 1)
2170  IF LO(J1,LV) <  > JQ(J) THEN 2200
2180  LET I = 1: IF MT > 1 THEN I = LZ
2190  LET L = 2 * J:K = L - 1:P(K,J2,I) = P(K,J2,I) + R1(J1,LV):P(L,J2,I) = P(L,J2,
   I) + R2(J1,LV)
2200  NEXT J
2210  NEXT J2
2220  NEXT J1
2230  IF DL = 0 THEN 2410
2240  REM     ::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
2250  REM     :: ADD DISTRIBUTED LOADS INTO LOAD VECTORS OF ELEMENT ::
2260  REM     ::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
2270  FOR J1 = 1 TO DL:J4 = J5 + LZ
2280  FOR I = 1 TO NE:JQ(I) = IQ(I): NEXT I
```

```
2290  FOR J2 = 1 TO VV: IF J2 > 1 THEN J4 = J4 + LB
2300  FOR J3 = 1 TO NE: IF J2 > 1 THEN JQ(J3) = JQ(J3) - FT(IT + J2 - 1)
2310  FOR J = 1 TO DN: IF ELN(J1) <  > J4 OR DISN(J1,J) <  > JQ(J3) THEN 2340
2320  LET I = 1: IF MT > 1 THEN I = LZ
2330  LET L = 2 * J3:K = L - 1:P(K,J2,I) = P(K,J2,I) + R5(J1,J):P(L,J2,I) = P(L,J2,
   I) + R6(J1,J)
2340  NEXT J
2350  NEXT J3
2360  NEXT J2
2370  NEXT J1
2380  REM     ::::::::::::::::::::::::::::::::::::::::::::::::
2390  REM     :: ENTER LOOPS FOR AREA NUMERICAL INTEGRATION ::
2400  REM     ::::::::::::::::::::::::::::::::::::::::::::::::
2410  FOR NA = 1 TO NG:PX = PL(NA,NG)
2420  FOR NB = 1 TO NG:PE = PL(NB,NG): IF MN = 8 THEN 2480
2430  REM     ::::::::::::::::::::::::::::::::::::::::::::::::::::
2440  REM     :: FIND THE SHAPE FUNCTIONS AND SHAPE FUNCTION ::
2450  REM     :: DERIVATIVES FOR 4-NODES OR 8-NODES ELEMENT  ::
2460  REM     ::::::::::::::::::::::::::::::::::::::::::::::::::::
2470  FOR L = 1 TO 4:ADUM = (1 + XI(L) * PX) / 4:BDUM = (1 + ET(L) * PE) / 4:EN(L) =
   4 * ADUM * BDUM:NXI(L) = XI(L) * BDUM:NTE(L) = ET(L) * ADUM: NEXT L: GOTO 2580
2480  LET EN(1) = (1 - PX) * (1 - PE) * ( - PX - PE - 1) / 4:EN(2) = (1 + PX) * (1 -
   PE) * (PX - PE - 1) / 4:EN(3) = (1 + PX) * (1 + PE) * (PX + PE - 1) / 4:EN(4) =
   (1 - PX) * (1 + PE) * ( - PX + PE - 1) / 4
2490  LET EN(5) = (1 - PX ^ 2) * (1 - PE) / 2:EN(6) = (1 + PX) * (1 - PE ^ 2) / 2:E
   N(7) = (1 - PX ^ 2) * (1 + PE) / 2:EN(8) = (1 - PX) * (1 - PE ^ 2) / 2
2500  LET NXI(1) = (PE + 2 * PX - 2 * PX * PE - PE ^ 2) / 4:NXI(2) = ( - PE + 2 * P
   X - 2 * PX * PE + PE ^ 2) / 4:NXI(3) = (PE + 2 * PX + 2 * PX * PE + PE ^ 2) / 4
   :NXI(4) = ( - PE + 2 * PX + 2 * PX * PE - PE ^ 2) / 4
2510  LET NXI(5) = ( - PX + PX * PE):NXI(6) = (1 - PE ^ 2) / 2:NXI(7) = ( - PX - PX
    * PE):NXI(8) = ( - 1 + PE ^ 2) / 2
2520  LET NTE(1) = (PX + 2 * PE - 2 * PX * PE - PX ^ 2) / 4:NTE(2) = ( - PX + 2 * P
   E + 2 * PX * PE - PX ^ 2) / 4:NTE(3) = (PX + 2 * PE + 2 * PX * PE + PX ^ 2) / 4
   :NTE(4) = ( - PX + 2 * PE - 2 * PX * PE + PX ^ 2) / 4
2530  LET NTE(5) = ( - 1 + PX ^ 2) / 2:NTE(6) = ( - PE - PX * PE):NTE(7) = (1 - PX ^
   2) / 2:NTE(8) = ( - PE + PX * PE)
2540  REM     ::::::::::::::::::::::::::::::::::::
2550  REM     :: INITIALIZE JACOBIAN MATRIX AND ::
2560  REM     :: STRAIN-DISPLACEMENT MATRIX B   ::
2570  REM     ::::::::::::::::::::::::::::::::::::
2580  FOR I = 1 TO 2
2590  FOR J = 1 TO 2:JAC(I,J) = 0: NEXT J
2600  NEXT I
2610  FOR I = 1 TO 3
2620  FOR J = 1 TO 17:B(I,J) = 0: NEXT J
2630  NEXT I
```

```
2640  REM     ::::::::::::::::::::::::::::::::::::::::::::::
2650  REM     :: FIND JACOBIAN MATRIX AND ITS DETERMINANT ::
2660  REM     ::::::::::::::::::::::::::::::::::::::::::::::
2670  FOR L = 1 TO NE:JAC(1,1) = JAC(1,1) + NXI(L) * XL(L):JAC(1,2) = JAC(1,2) + NX
   I(L) * YL(L):JAC(2,1) = JAC(2,1) + NTE(L) * XL(L):JAC(2,2) = JAC(2,2) + NTE(L) *
   YL(L): NEXT L
2680  LET DJ = JAC(1,1) * JAC(2,2) - JAC(2,1) * JAC(1,2):ADUM = JAC(1,1) / DJ
2690  LET JAC(1,1) = JAC(2,2) / DJ:JAC(1,2) =  - JAC(1,2) / DJ:JAC(2,1) =  - JAC(2,
   1) / DJ:JAC(2,2) = ADUM
2700  FOR J = 1 TO NE:L = 2 * J:K = L - 1
2710  REM     ::::::::::::::::::::::::::::::::::::::
2720  REM     :: FORM STRAIN-DISPLACEMENT MATRIX B ::
2730  REM     ::::::::::::::::::::::::::::::::::::::
2740  LET B(1,K) = JAC(1,1) * NXI(J) + JAC(1,2) * NTE(J):B(2,L) = JAC(2,1) * NXI(J)
    + JAC(2,2) * NTE(J):B(3,K) = B(2,L):B(3,L) = B(1,K): NEXT J
2750  LET TK = 0
2760  FOR L = 1 TO NE:TK = TK + EN(L) * THC: NEXT L
2770  LET BY =  - DEN(IX):BX = 0
2780  LET DV = WGT(NA,NG) * WGT(NB,NG) * TK * DJ
2790  FOR J = 1 TO NE:L = 2 * J:K = L - 1
2800  FOR N = 1 TO 3:BTE(K,N) = B(1,K) * E(1,N) + B(3,K) * E(3,N):BTE(L,N) = B(2,L)
    * E(2,N) + B(3,L) * E(3,N): NEXT N
2810  LET N = 1: IF MT > 1 THEN N = LZ
2820  REM     ::::::::::::::::::::::::::::::::::::::::::::::::
2830  REM     :: ADD BODY FORCES TO LOAD VECTORS OF ELEMENT ::
2840  REM     ::::::::::::::::::::::::::::::::::::::::::::::::
2850  FOR I = 1 TO VV:P(K,I,N) = P(K,I,N) + EN(J) * BX * DV:P(L,I,N) = P(L,I,N) + E
   N(J) * BY * DV: NEXT I
2860  NEXT J
2870  LET K = 1: IF MT > 1 THEN K = LZ
2880  REM     ::::::::::::::::::::::::::::::::::::::::::::::::::::::::
2890  REM     :: ADD INITIAL THERMAL STRAINS TO LOAD VECTORS OF ::
2900  REM     :: ELEMENT. CALCULATE ELEMENT STIFFNESS          ::
2910  REM     ::::::::::::::::::::::::::::::::::::::::::::::::::::::::
2920  FOR I = 1 TO NE:JQ(I) = IQ(I): NEXT I
2930  FOR II = 1 TO VV
2940  FOR I = 1 TO 3:B(I,17) = 0: NEXT I
2950  FOR L = 1 TO NE: IF II > 1 THEN JQ(L) = JQ(L) - FT(IT + II - 1)
2960  LET I = JQ(L): IF TE = 1 THEN 2980
2970  LET EX(L) = AP(IX) * (T(I) - RT):EY(L) = AP(IX) * (T(I) - RT):ES(L) = 0: GOTO
   2990
2980  LET EX(L) = (1 + PR(IX)) * AP(IX) * (T(I) - RT):EY(L) = (1 + PR(IX)) * AP(IX)
    * (T(I) - RT):ES(L) = 0
2990  LET B(1,17) = B(1,17) + EN(L) * EX(L):B(2,17) = B(2,17) + EN(L) * EY(L):B(3,1
   7) = B(3,17) + EN(L) * ES(L): NEXT L
3000  FOR NR = 1 TO M
3010  FOR J = 1 TO 3:P(NR,II,K) = P(NR,II,K) + BTE(NR,J) * B(J,17) * DV: NEXT J
3020  NEXT NR
3030  NEXT II
```

```
3040  IF MT > 1 AND LZ < LB THEN 3100
3050  FOR NR = 1 TO M
3060  FOR NC = NR TO M:DUM = 0
3070  FOR J = 1 TO 3:DUM = DUM + BTE(NR,J) * B(J,NC): NEXT J
3080  LET S(NR,NC) = S(NR,NC) + DUM * DV: NEXT NC
3090  NEXT NR
3100  NEXT NB
3110  NEXT NA
3120  IF MT > 1 AND LZ < LB THEN 3300
3130  LET NW = (M * (M + 1)) / 2 + (M * VV * LB):X2 = X2 + 1:LW = X2
3140  IF MT = 1 THEN NW = (M * (M + 1)) / 2 + (M * VV)
3150  IF XK < NW THEN XK = NW
3160  PRINT D$;"OPEN STIFF1,L";ZA: PRINT D$;"WRITE STIFF1,R";LW
3170  PRINT LC: PRINT LC: PRINT M: PRINT M: PRINT NW: PRINT VV
3180  FOR I = 1 TO M
3190  FOR J = I TO M: PRINT S(I,J): NEXT J
3200  NEXT I: IF MT = 1 THEN 3260
3210  FOR K = 1 TO LB
3220  FOR I = 1 TO M
3230  FOR J = 1 TO VV: PRINT P(I,J,K): NEXT J
3240  NEXT I
3250  NEXT K: GOTO 3290
3260  FOR I = 1 TO M
3270  FOR J = 1 TO VV: PRINT P(I,J,1): NEXT J
3280  NEXT I
3290  PRINT D$;"CLOSE STIFF1"
3300  NEXT LZ:LY = LY + (VV - 1) * LB:J5 = J5 + (VV * LB)
3310  NEXT IT: IF XK = XC THEN 3360
3320  PRINT D$;"OPEN DAT1,D2": PRINT D$;"DELETE DAT1"
3330  PRINT D$;"OPEN DAT1": PRINT D$;"WRITE DAT1"
3340  PRINT XK: PRINT PT: PRINT EL: PRINT MN: PRINT NF: PRINT MI: PRINT PS: PRINT N
    S: PRINT KSUB: PRINT JZ: PRINT ZT: PRINT ZQ: PRINT ZA: PRINT E3: PRINT YT: PRINT
    MT: PRINT LU: PRINT BN: PRINT TE
3350  PRINT D$;"CLOSE DAT1"
3360  PRINT D$;"RUN FRNTIQ,D1"
3370  REM     ::::::::::::::::::::::::::::::::::::::
3380  REM     :: LOOP OVER EACH DISTRIBUTED LOAD ::
3390  REM     ::::::::::::::::::::::::::::::::::::::
3400  FOR IE = 1 TO DL:PE =  - 1
3410  FOR J = 1 TO DN:R5(IE,J) = 0:R6(IE,J) = 0: NEXT J
3420  REM     ::::::::::::::::::::::::::::::::::::::::::::::::::::
3430  REM     :: ENTER LOOPS FOR LINEAR NUMERICAL INTEGRATION ::
3440  REM     ::::::::::::::::::::::::::::::::::::::::::::::::::::
3450  FOR NA = 1 TO NG:PX = PL(NA,NG): IF MN = 8 THEN 3510
```

```
3460  REM     ::::::::::::::::::::::::::::::::::::::::::::::::
3470  REM     :: FIND THE SHAPE FUNCTIONS AND SHAPE FUNCTION ::
3480  REM     :: DERIVATIVES FOR 4-NODES OR 8-NODES ELEMENT  ::
3490  REM     ::::::::::::::::::::::::::::::::::::::::::::::::
3500  FOR L = 1 TO 2:ADUM = (1 + XI(L) * PX) / 4:BDUM = (1 + ET(L) * PE) / 4:EN(L) =
    4 * ADUM * BDUM:NXI(L) = XI(L) * BDUM: NEXT L: GOTO 3570
3510  LET EN(1) = (1 - PX) * (1 - PE) * ( - PX - PE - 1) / 4:EN(2) = (1 - PX ^ 2) *
    (1 - PE) / 2:EN(3) = (1 + PX) * (1 - PE) * (PX - PE - 1) / 4
3520  LET NXI(1) = (PE + 2 * PX - 2 * PX * PE - PE ^ 2) / 4:NXI(2) = ( - PX + PX *
    PE):NXI(3) = ( - PE + 2 * PX - 2 * PX * PE + PE ^ 2) / 4
3530  REM     ::::::::::::::::::::::::::::::::::::::
3540  REM     :: CALCULATE EQUIVALENT NODAL LOADS ::
3550  REM     :: FROM DISTRIBUTED LOADS           ::
3560  REM     ::::::::::::::::::::::::::::::::::::::
3570  FOR I = 1 TO 2:PG(I) = 0:DG(I) = 0: NEXT I
3580  FOR J = 1 TO DN:PG(1) = PG(1) + EN(J) * R3(IE,J):PG(2) = PG(2) + EN(J) * R4(I
    E,J):L = DISN(IE,J):DG(1) = DG(1) + NXI(J) * X(L):DG(2) = DG(2) + NXI(J) * Y(L)
    : NEXT J
3590  LET DV = WGT(NA,NG):DX = DG(1) * PG(2) - DG(2) * PG(1):DY = DG(1) * PG(1) + D
    G(2) * PG(2)
3600  FOR I = 1 TO DN:R5(IE,I) = R5(IE,I) + EN(I) * DX * DV:R6(IE,I) = R6(IE,I) + E
    N(I) * DY * DV: NEXT I
3610  NEXT NA
3620  NEXT IE
3630  RETURN
```

```
1000  HOME :D$ =  CHR$ (4): GOTO 1110: REM    ** FRNTIQ  **
1010  PRINT "----------": PRINT
1020  PRINT "  FRNTIQ  ": PRINT
1030  PRINT "----------": PRINT : RETURN
1040  REM
1050  REM     ::::::::::::::::::::::::::::::::::::::::::::::::::
1060  REM     :: THIS SUBPROGRAM CONTROLS THE PREFRONT AT THE ::
1070  REM     :: SUBSTRUCTURE LEVEL AND THEN SETS DESTINATION ::
1080  REM     :: ARRAYS FOR FRONT PROCESSING                  ::
1090  REM     ::::::::::::::::::::::::::::::::::::::::::::::::::
1100  REM
1110  GOSUB 1010: PRINT D$;"OPEN DAT1,D2": PRINT D$;"READ DAT1"
1120  INPUT XC,PT,EL,MN,NF,MI,PS,NS,KSUB,JZ,ZT,ZQ,ZA,E3,YT,MT,LU,BN,TE
1130  PRINT D$;"CLOSE DAT1"
1140  LET CM = 5000:TM = (3 * PT) + (2 * JZ) + (7 * NS) + BN + 7:GT =  INT ((5000 -
    TM) / 10)
1150  DIM ND(PT),IB(PT),IC(PT),IQ(JZ),AMOC(GT),BMOC(GT),LQ(GT),SM(JZ),FS(BN),CD(NS,
    3),DUM(7),MOC(GT),VH(GT),SN(NS,2),SD(2 * NS),LF(5 * GT + 7)
1160  PRINT D$;"OPEN SUPSET,D2": PRINT D$;"READ SUPSET"
1170  FOR I = 1 TO NS: INPUT SN(I,1),SN(I,2): NEXT I
1180  PRINT D$;"CLOSE SUPSET"
1190  FOR I = 1 TO 2 * NS:SD(I) = 0: NEXT I
1200  FOR I = 1 TO PT:ND(I) = 2: NEXT I: GOTO 1220
1210  PRINT "---------------------------------------": RETURN
1220  PRINT D$;"OPEN IQ1,D1": PRINT D$;"DELETE IQ1": PRINT D$;"CLOSE IQ1"
1230  PRINT D$;"OPEN FRONT,D2": PRINT D$;"DELETE FRONT": PRINT D$;"CLOSE FRONT"
1240  LET L1$ = "ELE1":L5$ = "ELE2":FI = 1:X1 =  - 1:X3 =  - 1:X5 =  - 1:X6 =  - 1:
    X7 =  - 1:ZT =  - 1:SD = 0:ZU = (4 * JZ * 6) + 42: IF KSUB > 2 THEN ZU = (5 * J
    Z * 6) + 42
1250  REM     ::::::::::::::::::::::::::::::::::::::::::::::::::
1260  REM     :: LOOP FROM SUBSTRUCTURE LEVEL 1 TO LEVEL KSUB ::
1270  REM     ::::::::::::::::::::::::::::::::::::::::::::::::::
1280  FOR IM = 1 TO KSUB: HOME : GOSUB 1010: VTAB 8: GOSUB 1210: PRINT
1290  PRINT  TAB( 5)"SUBSTRUCTURING IN LEVEL ";IM: PRINT : GOSUB 1210
1300  PRINT : PRINT : PRINT  TAB( 16)"";: FLASH : PRINT "RUNNING": NORMAL
1310  LET SP = 0:SF = 0:S3 = 0:S4 = 0: GOTO 1420
1320  LET ZT = ZT + 1: PRINT D$;"OPEN SUBST,L";YT;",D2": PRINT D$;"READ SUBST,R";ZT

1330  INPUT NB,JS,NP
1340  FOR I = 1 TO NB: INPUT FS(I): NEXT I
1350  IF JS = 0 THEN 1370
1360  FOR I = 1 TO JS: INPUT SM(I): NEXT I
1370  IF NP = 0 THEN 1410
1380  FOR I = 1 TO NP
1390  FOR J = 1 TO 3: INPUT CD(I,J): NEXT J
1400  NEXT I
```

```
1410  PRINT D$;"CLOSE SUBST": RETURN
1420  GOSUB 1320
1430  LET T$ = "IQ1":VC = FI:VD = NB:VE = NB:VF = NB:X3 = X3 + 1:LW = X3:ZK = ZU:J =
    1
1440  FOR I = 1 TO VF:VH(I) = FS(I): NEXT I: GOSUB 3690
1450  IF S3 = 1 THEN F0 = 1
1460  REM     ::::::::::::::::::::::::::::::::::
1470  REM     :: LOOP OVER EACH SUBSTRUCTURE ::
1480  REM     ::::::::::::::::::::::::::::::::::
1490  FOR IT = 1 TO NB:LB = FS(IT): IF LB > 0 AND IT > 1 THEN  GOSUB 1320
1500  IF LB < 0 THEN 1520
1510  GOTO 1530
1520  FOR I = 1 TO JS:SM(I) = SM(I) +  INT ( ABS (LB)): NEXT I
1530  IF LB < 1 THEN 2900
1540  IF JS < 1 THEN 1590
1550  REM     ::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
1560  REM     :: SET NEGATIVE VALVES  ND ARRAY FOR SUBSTRUCTURE NODES ::
1570  REM     ::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
1580  FOR I = 1 TO JS:L = SM(I):ND(L) =  - ND(L): NEXT I
1590  LET XD = 0:XF = 0:XE = 0:EX = 0:XM = 0:PQ = 0
1600  FOR I = 1 TO PT:IB(I) = 0: NEXT I
1610  REM     ::::::::::::::::::::::::::::
1620  REM     :: LOOP OVER EACH ELEMENT ::
1630  REM     ::::::::::::::::::::::::::::
1640  FOR LZ = 1 TO LB:T$ = L1$:X1 = X1 + 1:LW = X1: GOSUB 3170
1650  FOR I = 1 TO VF:IQ(I) = VH(I): NEXT I
1660  LET NE = VF
1670  FOR I = 1 TO NE:K = IQ(I):IB(K) = IB(K) + 1: NEXT I
1680  NEXT LZ
1690  FOR I = 1 TO PT:IC(I) = IB(I): NEXT I
1700  REM     ::::::::::::::::::::::::::::::::::::::::::::::::
1710  REM     :: LOOP OVER EACH ELEMENT CREATES  MOC ARRAY ::
1720  REM     :: WHICH CONTAINS ACTIVE NODES IN A FRONT    ::
1730  REM     ::::::::::::::::::::::::::::::::::::::::::::::::
1740  FOR LZ = 1 TO LB:KZ = LB - LZ:X6 = X6 + 1:LW = X6:NG = 1: GOSUB 3220
1750  IF GA < N THEN GA = N
1760  NEXT LZ
1770  LET MQ = QM:NQ = QN
1780  FOR I = 1 TO GA:MOC(I) = AMOC(I): NEXT I
1790  LET LW = X6 + 1:NG = 0:KZ = 0: GOSUB 3220
1800  FOR I = 1 TO GA:AMOC(I) = MOC(I): NEXT I
1810  LET X6 = X6 - LB:IG = 2:LEV = 1:AP = 0:BP = 0:FC = XF
1820  LET T$ = "IQ1":VC = FC:VD = LEV:VE = CB:VF = 13:X3 = X3 + 1:LW = X3:ZK = ZU:J
    = 1: GOSUB 3210
1830  GOSUB 3690
1840  LET ALV = 1:T$ = "FRONT":LW = 0:ZK = ZQ:J = 2: GOSUB 3730
1850  LET UA = VC:PE = VD:U1 = VE:V1 = VF
1860  FOR I = 1 TO V1:AMOC(I) = VH(I): NEXT I
```

```
1870  REM     ::::::::::::::::::::::::::::
1880  REM     :: LOOP OVER EACH ELEMENT ::
1890  REM     ::::::::::::::::::::::::::::
1900  FOR LZ = 1 TO LB:T$ = "FRONT":LW = LZ:ZK = ZQ:J = 2: GOSUB 3730
1910  LET UB = VC:EP = VD:U2 = VE:V2 = VF
1920  FOR I = 1 TO V2:BMOC(I) = VH(I): NEXT I
1930  LET T$ = L1$:X6 = X6 + 1:LW = X6: GOSUB 3170
1940  LET NE = VF:JG = 0
1950  FOR I = 1 TO NE:IQ(I) = VH(I): NEXT I
1960  IF UA < 0 OR UB < 0 THEN JG = 1
1970  IF LZ =  > LB - 1 THEN JG = 1
1980  IF JG <  > 0 THEN 2200
1990  REM     ::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
2000  REM     :: FOR LEVEL-1 FRONT. NODES IN CURRENT FRONT & NEXT FRONT ::
2010  REM     :: (AMOC & BMOC ARRAY) KEPT IN THE SAME LOCATIONS AS FAR  ::
2020  REM     :: AS POSSIBLE FOR MINIMIZE INTERCHANGING OF COEFFICIENTS ::
2030  REM     ::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
2040  LET H1 = U1 + 1:H2 = U2 + 1
2050  IF H1 =  > H2 THEN IX = H1
2060  IF H2 =  > H1 THEN IX = H2
2070  IF V1 <  = V2 THEN IY = V1
2080  IF V2 <  = V1 THEN IY = V2
2090  IF IX > IY THEN 2200
2100  FOR I = IX TO IY:K = AMOC(I)
2110  FOR J = H2 TO V2: IF BMOC(J) = K THEN 2130
2120  NEXT J: GOTO 2140
2130  LET BMOC(J) = BMOC(I):BMOC(I) = K
2140  NEXT I
2150  REM     ::::::::::::::::::::::::::::::::::::::::::::::::::::::::
2160  REM     :: FIND DOF. SIZE OF ELEMENT NODES INTO FRONT,      ::
2170  REM     :: DOF. SIZE OF ELIMINATED NODES IN CURRENT FRONT,  ::
2180  REM     :: DOF. SIZE OF NODES IN CURRENT FRONT & NEXT FRONT ::
2190  REM     ::::::::::::::::::::::::::::::::::::::::::::::::::::::::
2200  LET NUM = 0: IF NE = 0 THEN 2220
2210  FOR I = 1 TO NE:KK = IQ(I):NUM = NUM +  INT ( ABS (ND(KK))): NEXT I
2220  LET D1 = NUM:NUM = 0: IF U1 = 0 THEN 2240
2230  FOR I = 1 TO U1:KK = AMOC(I):NUM = NUM +  INT ( ABS (ND(KK))): NEXT I
2240  LET QM = NUM:QK = QM + 1:NUM = 0: IF V1 = 0 THEN 2260
2250  FOR I = 1 TO V1:KK = AMOC(I):NUM = NUM +  INT ( ABS (ND(KK))): NEXT I
2260  LET QN = NUM:NUM = 0: IF V2 = 0 THEN 2280
2270  FOR I = 1 TO V2:KK = BMOC(I):NUM = NUM +  INT ( ABS (ND(KK))): NEXT I
2280  LET QR = NUM: IF QM <  = 0 THEN 2460
2290  REM     ::::::::::::::::::::::::::::::::::::::::
2300  REM     :: SAVES BOUNDARY CONSTRIANT CODE IN ::
2310  REM     :: (D1+I) LOCATIONS OF  LF ARRAY     ::
2320  REM     ::::::::::::::::::::::::::::::::::::::::
2330  LET GI = 2
2340  FOR I = 1 TO QM:LF(D1 + I) = 0: NEXT I
2350  IF NP < 1 THEN 2460
```

```
2360  FOR K = 1 TO U1
2370  FOR J = 1 TO NP: IF AMOC(K) <  > CD(J,1) THEN 2440
2380  LET LY = AMOC(K):NUM = 0: IF K = 0 THEN 2400
2390  FOR I = 1 TO K:KK = AMOC(I):NUM = NUM +  INT ( ABS (ND(KK))): NEXT I
2400  LET Q2 = NUM:Q1 = Q2 -  INT ( ABS (ND(LY))) + 1:L = 1
2410  FOR X = Q1 TO Q2:L = L + 1:LF(D1 + X) = CD(J,L): IF LF(D1 + X) = 1 THEN SD =
    SD + 1:SD(SD) = SN(J,L - 1)
2420  NEXT X
2430  GOTO 2450
2440  NEXT J
2450  NEXT K
2460  IF B8 < QN THEN B8 = QN
2470  REM     ::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
2480  REM     :: CONVERTS NODAL LOCATIONS INTO DOF. LOCATIONS IN A FRONT ::
2490  REM     ::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
2500  LET LK = 0:GX = 1:Y1 = NE:Y2 = V1:L = LK
2510  LET LQ(1) = 1: IF Y2 = 1 THEN 2530
2520  FOR I = 2 TO Y2:M = AMOC(I - 1):LQ(I) = LQ(I - 1) +  INT ( ABS (ND(M))): NEXT
    I
2530  FOR I = GX TO Y1:M = IQ(I)
2540  FOR J = 1 TO Y2: IF M = AMOC(J) THEN 2560
2550  NEXT J
2560  LET N =  INT ( ABS (ND(M))) - 1:M = LQ(J):N = M + N
2570  FOR X = M TO N:L = L + 1:LF(L) = X: NEXT X
2580  NEXT I
2590  LET LK = L:LK = LK + QM:KN = U1 + 1: IF KN > V1 THEN 2690
2600  LET GX = KN:Y1 = V1:Y2 = V2:L = LK
2610  LET LQ(1) = 1: IF Y2 = 1 THEN 2630
2620  FOR I = 2 TO Y2:M = BMOC(I - 1):LQ(I) = LQ(I - 1) +  INT ( ABS (ND(M))): NEXT
    I
2630  FOR I = GX TO Y1:M = AMOC(I)
2640  FOR J = 1 TO Y2: IF M = BMOC(J) THEN 2660
2650  NEXT J
2660  LET N =  INT ( ABS (ND(M))) - 1:M = LQ(J):N = M + N
2670  FOR X = M TO N:L = L + 1:LF(L) = X: NEXT X
2680  NEXT I:LK = L
2690  LET Y1 = D1 + 1:Y2 = D1 + QN:LI = LK:CT = LI:T$ = "IQ1":X3 = X3 + 1:LN = X3:G
    I = 1:ZK = ZU:J = 1
2700  REM     ::::::::::::::::::::::::::::::::::::::::::::::::
2710  REM     :: SAVES DATA IN FILE "IQ1" FOR IN ASSEMBLING ::
2720  REM     :: AND UPDATING COEFFICIENTS IN A FRONT       ::
2730  REM     ::::::::::::::::::::::::::::::::::::::::::::::::
2740  LET DUM(1) = QN:DUM(2) = QR:DUM(3) = PE:DUM(4) = EP:DUM(5) = TI:DUM(6) = TJ:D
    UM(7) = D1
2750  FOR I = 1 TO 7:CT = CT + 1:LF(CT) = DUM(I): NEXT I
2760  PRINT D$;"OPEN";T$;",L";ZK;",D";J: PRINT D$;"WRITE";T$;",R";LN
2770  PRINT LK: PRINT QM: PRINT QK: PRINT CT
2780  FOR I = 1 TO CT: PRINT LF(I): NEXT I
2790  PRINT D$;"CLOSE";T$
```

```
2800  REM     ::::::::::::::::::::::::::::::::::::::::::::::::::::::
2810  REM     :: UPDATES THE NODE NUMBER BETWEEN  AMOC & BMOC ARRAY ::
2820  REM     ::::::::::::::::::::::::::::::::::::::::::::::::::::::
2830  FOR I = 1 TO V2:AMOC(I) = BMOC(I): NEXT I
2840  LET U1 = U2:V1 = V2:UA = UB:PE = EP
2850  IF GS < CT THEN GS = CT
2860  NEXT LZ
2870  LET T$ = "IQ1":VC = FC:VD = LEV:VE = CB:VF = 13:X3 = X3 + 1:LW = X3:ZK = ZU:J
     = 1: GOSUB 3210
2880  GOSUB 3690
2890  FOR I = 1 TO PT:ND(I) =  INT ( ABS (ND(I))): NEXT I
2900  IF JS < 1 THEN 2970
2910  LET T$ = L5$:X5 = X5 + 1:LW = X5
2920  PRINT D$;"OPEN";T$;",L";ZQ;",D2": PRINT D$;"WRITE";T$;",R";LW
2930  PRINT JS
2940  FOR I = 1 TO JS: PRINT SM(I): NEXT I
2950  PRINT D$;"CLOSE";T$
2960  LET X6 = X1:X7 =  - 1
2970  NEXT IT
2980  LET T$ = L1$:L1$ = L5$:L5$ = T$
2990  LET X1 =  - 1:X5 =  - 1:X6 =  - 1:X7 =  - 1
3000  LET T$ = "IQ1":VC = FI:VD = NB:VE = NB:VF = NB:X3 = X3 + 1:LW = X3:ZK = ZU:J =
    1
3010  FOR I = 1 TO VF:VH(I) = FS(I): NEXT I: GOSUB 3690
3020  NEXT IM
3030  LET TM = TM + (5 * GA) + GS: HOME : GOSUB 1010
3040  VTAB 8: PRINT  TAB( 4)"USED MEMORY IN FRNTIQ     = ";TM
3050  PRINT D$;"OPEN NEWSUP,D2": PRINT D$;"DELETE NEWSUP"
3060  PRINT D$;"OPEN NEWSUP": PRINT D$;"WRITE NEWSUP"
3070  FOR I = 1 TO 2 * NS: PRINT SD(I): NEXT I
3080  PRINT D$;"CLOSE NEWSUP"
3090  PRINT D$;"OPEN DAT2,D2": PRINT D$;"DELETE DAT2"
3100  PRINT D$;"OPEN DAT2": PRINT D$;"WRITE DAT2"
3110  PRINT GS: PRINT S2: PRINT S4: PRINT ZU: PRINT B8
3120  PRINT D$;"CLOSE DAT2": IF KSUB < 3 THEN 3140
3130  PRINT D$;"OPEN ELE1,D2": PRINT D$;"DELETE ELE1": PRINT D$;"CLOSE ELE1"
3140  PRINT D$;"OPEN ELE2,D2": PRINT D$;"DELETE ELE2": PRINT D$;"CLOSE ELE2"
3150  PRINT D$;"OPEN FRONT,D2": PRINT D$;"DELETE FRONT": PRINT D$;"CLOSE FRONT"
3160  PRINT D$;"RUN FRNTST,D1"
3170  PRINT D$;"OPEN";T$;",L";ZQ;",D2": PRINT D$;"READ";T$;",R";LW
3180  INPUT VF
3190  FOR I = 1 TO VF: INPUT VH(I): NEXT I
3200  PRINT D$;"CLOSE";T$: RETURN
3210  LET VH(1) = XD:VH(2) = XF:VH(3) = XE:VH(4) = EX:VH(5) = PL:VH(6) = PQ:VH(7) =
    MQ:VH(8) = NQ:VH(9) = XM:VH(10) = MO:VH(11) = AP:VH(12) = BP:VH(13) = MAY: RETURN

3220  LET PE = 0: IF NG < 1 THEN 3300
3230  LET T$ = L1$: GOSUB 3170
3240  FOR I = 1 TO VF:IQ(I) = VH(I): NEXT I
3250  LET NE = VF
```

```
3260  FOR I = 1 TO NE:K = IQ(I)
3270  IF ND(K) > 0 THEN IC(K) = IC(K) - 1
3280  IF ND(K) < 0 THEN IC(K) = IC(K) + 1
3290  NEXT I
3300  LET M = 0:N = 0
3310  FOR I = 1 TO PT: IF IB(I) = IC(I) THEN 3390
3320  IF IC(I) <  > 0 THEN 3380
3330  REM     ::::::::::::::::::::::::::::::::::::::::
3340  REM     :: THE ELIMINATED NODES CONTAINS THE ::
3350  REM     :: FIRST  M LOCATIONS IN MOC ARRAY   ::
3360  REM     ::::::::::::::::::::::::::::::::::::::::
3370  LET IB(I) = 0:M = M + 1:MOC(M) = I: GOTO 3390
3380  LET N = N + 1:LQ(N) = I
3390  NEXT I
3400  REM     ::::::::::::::::::::::::::::::::::::::::::::::::
3410  REM     :: THE RETAINED NODES CONTAINS THE REMAINING ::
3420  REM     :: (N-M) LOCATIONS IN  MOC ARRAY             ::
3430  REM     ::::::::::::::::::::::::::::::::::::::::::::::::
3440  FOR I = 1 TO N:MOC(I + M) = LQ(I): NEXT I
3450  LET N = M + N:NUM = 0: IF M = 0 THEN 3470
3460  FOR I = 1 TO M:KK = MOC(I):NUM = NUM +  INT ( ABS (ND(KK))): NEXT I
3470  LET QM = NUM:NUM = 0: IF N = 0 THEN 3490
3480  FOR I = 1 TO N:KK = MOC(I):NUM = NUM +  INT ( ABS (ND(KK))): NEXT I
3490  LET QN = NUM:NUM = 0: IF NE = 0 THEN 3510
3500  FOR I = 1 TO NE:KK = IQ(I):NUM = NUM +  INT ( ABS (ND(KK))): NEXT I
3510  LET D1 = NUM:IN = QM * QN - (QM * (QM - 1)) / 2:IE = IN * FI:GE = IE + (QN -
   QM) * FI
3520  IF PE > EX THEN EX = PE
3530  IF D1 > XD THEN XD = D1
3540  IF QN > XF THEN XF = QN
3550  IF IE > XE THEN XE = IE
3560  IF GE > N2 THEN N2 = GE
3570  IF M = N THEN 3610
3580  IF KZ > 0 THEN 3610
3590  LET J = M + 1
3600  FOR I = J TO N:MOC(I) = SM(I - M): NEXT I
3610  LET ANF = (QN * (QN + 1)) / 2 * FI: IF N <  > 0 THEN 3660
3620  FOR I = 1 TO GT:MOC(I) = 0: NEXT I
3630  REM     ::::::::::::::::::::::::::::::::::::::::::::::
3640  REM     :: SAVES  MOC ARRAY ON DATA FILE  "FRONT" ::
3650  REM     ::::::::::::::::::::::::::::::::::::::::::::::
3660  LET T$ = "FRONT":VC = ANF:VD = PE:VE = M:VF = N:X7 = X7 + 1:LN = X7:ZK = ZQ:J
    = 2
3670  FOR I = 1 TO VF:VH(I) = MOC(I): NEXT I: GOSUB 3690
3680  RETURN
3690  PRINT D$;"OPEN";T$;",L";ZK;",D";J: PRINT D$;"WRITE";T$;",R";LN
3700  PRINT VC: PRINT VD: PRINT VE: PRINT VF
3710  FOR I = 1 TO VF: PRINT VH(I): NEXT I
3720  PRINT D$;"CLOSE";T$: RETURN
3730  PRINT D$;"OPEN";T$;",L";ZK;",D";J: PRINT D$;"READ";T$;",R";LN
3740  INPUT VC,VD,VE,VF
3750  FOR I = 1 TO VF: INPUT VH(I): NEXT I
3760  PRINT D$;"CLOSE";T$: RETURN
```

```
1000  HOME :D$ =  CHR$ (4): GOTO 1100: REM    **  FRNTST  **
1010  PRINT "----------": PRINT
1020  PRINT "  FRNTST  ": PRINT
1030  PRINT "----------": PRINT : RETURN
1040  REM
1050  REM     ::::::::::::::::::::::::::::::::::::::::::::::::
1060  REM     :: THIS SUBPROGRAM PERFORMS FRONT PROCESSING ::
1070  REM     :: FROM SUBSTRUCTURING LEVEL 1 TO LEVEL KSUB ::
1080  REM     ::::::::::::::::::::::::::::::::::::::::::::::::
1090  REM
1100  GOSUB 1010: PRINT D$;"OPEN DAT1,D2": PRINT D$;"READ DAT1"
1110  INPUT XC,PT,EL,MN,NF,MI,PS,NS,KSUB,JZ,ZT,ZQ,ZA,E3,YT,MT,LU,BN,TE
1120  PRINT D$;"CLOSE DAT1"
1130  PRINT D$;"OPEN DAT2,D2": PRINT D$;"READ DAT2"
1140  INPUT GS,S2,S4,ZU,B8
1150  PRINT D$;"CLOSE DAT2"
1160  LET ZG = ZQ:ZK = ZQ:B9 = B8 * (B8 + 1) / 2 + B8 * LU:B3 = JZ * (2 * JZ + 1) +
    (2 * JZ * LU):CM = 5500
1170  LET GB = JZ: IF KSUB > 2 THEN GB = 1.5 * JZ
1180  IF GB < 13 THEN GB = 13
1190  IF XC > B9 THEN B9 = XC
1200  IF XC > B3 THEN B3 = XC
1210  LET XB = B9 + (B8 * LU):XW = XB:TM = EL + BN + B3 + B9 + (2 * B8) + NF + (B8 *
    LU) + (KSUB * BN) + (2 * NS) + 7 + GB + GS:GB = GB + CM - TM
1220  VTAB 8: PRINT  TAB( 4)"USED MEMORY IN FRNTST     = ";TM: IF TM < CM THEN 1250

1230  PRINT : PRINT  TAB( 4)"CORE MOMERY               = ";CM
1240  PRINT : PRINT  TAB( 4)"IT IS OVER CORE MEMORY    = ";TM - CM: PRINT : GOSUB 1
    270: END
1250  DIM FS(BN),S(B3),DUM(7),B(B8,LU),VH(GB),AO(B8),A(B9),ZP(EL),ZB(NF),NO(B8),ZT(
    KSUB * BN),SD(2 * NS),LF(GS)
1260  GOTO 1280
1270  PRINT "----------------------------------------": RETURN
1280  LET L16$ = "STIFF2":L2$ = "STIFF1"
1290  PRINT D$;"OPEN QM,D1": PRINT D$;"DELETE QM": PRINT D$;"CLOSE QM"
1300  PRINT D$;"OPEN QM,D2": PRINT D$;"DELETE QM": PRINT D$;"CLOSE QM"
1310  PRINT D$;"OPEN STIFF2,D2": PRINT D$;"DELETE STIFF2": PRINT D$;"CLOSE STIFF2"
1320  PRINT D$;"OPEN NEWSUP,D2": PRINT D$;"READ NEWSUP"
1330  FOR I = 1 TO 2 * NS: INPUT SD(I): NEXT I
1340  PRINT D$;"CLOSE NEWSUP":SD = 0
1350  LET X1 =  - 1:X2 =  - 1:X3 =  - 1:X4 =  - 1:X5 =  - 1:X6 =  - 1:X7 =  - 1:X8 =
     - 1:XY =  - 1:XZ =  - 1:C2 =  - 10:C3 = 0
1360  FOR I = 0 TO NF:ZB(I) = ZA: NEXT I
1370  REM     ::::::::::::::::::::::::::::::::::::::::::::::::::::::
1380  REM     :: LOOP FROM SUBSTRUCTURING LEVEL 1 TO LEVEL KSUB ::
1390  REM     ::::::::::::::::::::::::::::::::::::::::::::::::::::::
1400  FOR IM = 1 TO KSUB: HOME : GOSUB 1010: VTAB 8: GOSUB 1270: PRINT
1410  PRINT  TAB( 5)"SUBSTRUCTURING IN LEVEL ";IM: PRINT : GOSUB 1270
1420  PRINT : PRINT : PRINT  TAB( 16)"";: FLASH : PRINT "RUNNING": NORMAL
1430  LET T$ = "IQ1":X3 = X3 + 1:LW = X3:ZK = ZU: GOSUB 2700
1440  LET FI = VC:NB = VF:ZS = 0
```

```
1450  FOR I = 1 TO VF:FS(I) = VH(I): NEXT I
1460  REM     ::::::::::::::::::::::::::::::::::
1470  REM     :: LOOP OVER EACH SUBSTRUCTURE ::
1480  REM     ::::::::::::::::::::::::::::::::::
1490  FOR IT = 1 TO NB:LB = FS(IT): IF LB < 1 THEN 2530
1500  FOR I = 1 TO B9:A(I) = 0: NEXT I
1510  FOR I = 1 TO B8
1520  FOR J = 1 TO LU:B(I,J) = 0: NEXT J
1530  NEXT I:T$ = "IQ1":X3 = X3 + 1:LW = X3:ZK = ZU: GOSUB 2700
1540  GOSUB 2690
1550  LET FC = VC:LEV = VD:CB = VE
1560  REM     ::::::::::::::::::::::::::::
1570  REM     :: LOOP OVER EACH ELEMENT ::
1580  REM     ::::::::::::::::::::::::::::
1590  FOR LZ = 1 TO LB
1600  IF IM = 2 AND LZ > 1 THEN X2 = X2 - 1
1610  LET T$ = "IQ1":GI = 2:X3 = X3 + 1:LW = X3:ZK = ZU
1620  PRINT D$;"OPEN";T$;",L";ZK;",D1": PRINT D$;"READ";T$;",R";LW
1630  INPUT LD,QM,QK,CT
1640  FOR I = 1 TO CT: INPUT LF(I): NEXT I
1650  PRINT D$;"CLOSE";T$
1660  LET CT = CT - 7:M = CT
1670  FOR I = 1 TO 7:M = M + 1:DUM(I) = LF(M): NEXT I
1680  LET QN = DUM(1):QR = DUM(2):PE = DUM(3):EP = DUM(4):TI = DUM(5):TJ = DUM(6):D
   1 = DUM(7)
1690  LET T$ = "IQ1":XZ = XZ + 1:LW = XZ
1700  PRINT D$;"OPEN ";T$;",L";ZK;",D1": PRINT D$;"WRITE";T$;",R";LW
1710  PRINT QM: PRINT D1: PRINT QN: PRINT CT
1720  FOR I = 1 TO CT: PRINT LF(I): NEXT I
1730  PRINT D$;"CLOSE";T$
1740  LET VTR = 0:UDE = 0:SC = 0
1750  IF QN =  > QR AND QN > FC THEN VTR = 1
1760  IF QR =  > QN AND QR > FC THEN VTR = 1
1770  IF LZ = LB THEN UDE = 1
1780  IF VTR = 0 AND UDE = 0 THEN SC = 1
1790  REM     ::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
1800  REM     :: CALCULATES THE NUMBER OF ROWS OF COEFFICIENTS IN FRONT ::
1810  REM     ::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
1820  LET H3 = 1: GOSUB 2860
1830  IF H2 = 0 THEN 1850
1840  LET ZS = 2:JR = 0:IR = 1:JR = QN
1850  LET DO = 0:DI = 0: IF MR > 0 THEN DO = 1
1860  IF DO = 1 AND FI = 1 THEN DI = 1
1870  REM     ::::::::::::::::::::::::::::::::::::::::::::::::::
1880  REM     :: ASSEMBLES STIFFNESS MATRIX & LOAD VECTOR ::
1890  REM     :: WITH NEW ELEMENT                          ::
1900  REM     ::::::::::::::::::::::::::::::::::::::::::::::::::
1910  LET T$ = L2$: GOSUB 2970
```

```
1920  REM     ::::::::::::::::::::::::::::::::::::::::::::
1930  REM     :: USE MODIFY THE STIFFNESS COEFFICIENTS ::
1940  REM     :: IN A FRONT FOR BOUNDARY CONSTRAINTS   ::
1950  REM     ::::::::::::::::::::::::::::::::::::::::::::
1960  IF H2 = 1 AND MR > 0 THEN  GOSUB 2740
1970  REM     ::::::::::::::::::::::::::::::::::::::::::::::::::
1980  REM     :: REDUCES DOF. OF NODES WHICH WAS ELIMINATED ::
1990  REM     :: FROM FRONT BY GAUSS ELIMINATION            ::
2000  REM     ::::::::::::::::::::::::::::::::::::::::::::::::::
2010  IF DI = 1 THEN  GOSUB 3800
2020  IF H2 = 1 THEN IR = KR
2030  LET H2 = 0: IF MR = 0 THEN 2140
2040  REM     ::::::::::::::::::::::::::::::::::::::::::::::
2050  REM     :: SAVES DOF. OF NODES WHICH WAS ELIMINATED ::
2060  REM     :: FROM FRONT FOR USE IN BACKSUB           ::
2070  REM     ::::::::::::::::::::::::::::::::::::::::::::::
2080  LET T$ = "DM": GOSUB 3500
2090  LET IE = MR * FC - (MR * (MR - 1)) / 2
2100  FOR I = 1 TO IE:A(I) = 0: NEXT I
2110  FOR J = 1 TO LV
2120  FOR I = 1 TO MR:B(I,J) = 0: NEXT I
2130  NEXT J
2140  IF S7 = 0 THEN 2250
2150  REM     ::::::::::::::::::::::::::::::::::::::::::::
2160  REM     :: SAVES DOF. OF SUBSTRUCTURE NODES FROM ::
2170  REM     :: THE LAST REDUCED FRONT FOR USE IN THE ::
2180  REM     :: NEXT SUBSTRUCTURING LEVEL             ::
2190  REM     ::::::::::::::::::::::::::::::::::::::::::::
2200  LET T$ = L16$:XY = XY + 1:LW = XY:ZB(LW) = (JZ * (2 * JZ + 1) + (2 * JZ * LU)
    ) * 20:ZK = ZB(LW):IE = MR * QN - (MR * (MR - 1)) / 2
2210  LET GN = FC: IF FC > QN THEN GN = QN
2220  FOR I = 1 TO GN:AO(I) = NO(I): NEXT I: GOSUB 3230
2230  FOR I = 1 TO GN:NO(I) = AO(I): NEXT I
2240  IF S7 = 1 THEN XY = LW
2250  IF SC <  > 1 THEN 2470
2260  LET K = 0: IF MR < 1 THEN 2280
2270  FOR I = 1 TO MR:LF(D1 + I) = 0: NEXT I
2280  IF QN =  > QR THEN 2310
2290  LET J = QN + 1
2300  FOR I = J TO QR:LF(D1 + I) = 0: NEXT I
2310  IF QN =  > QR THEN EB = QN
2320  IF QR > QN THEN EB = QR
2330  REM     ::::::::::::::::::::::::::::::::::::::::::::::::::::
2340  REM     :: INTERCHANGES ROWS & COLUMNS IN CURRENT FRONT ::
2350  REM     :: TO ACCOMMODATE IN NEXT FRONT                ::
2360  REM     ::::::::::::::::::::::::::::::::::::::::::::::::::::
2370  FOR M1 = QK TO QN
2380  LET N1 = LF(D1 + M1): IF N1 = M1 OR N1 = 0 THEN 2460
2390  LET LF(D1 + M1) = LF(D1 + N1):LF(D1 + N1) = N1
2400  IF M1 <  = N1 THEN M = M1
```

```
2410  IF N1 < M1 THEN M = N1
2420  IF M1 =  > N1 THEN M = M1
2430  IF N1 > M1 THEN M = N1
2440  GOSUB 3690
2450  GOTO 2380
2460  NEXT M1
2470  NEXT LZ
2480  FOR I = 1 TO B9:A(I) = 0: NEXT I
2490  FOR I = 1 TO B3:S(I) = 0: NEXT I
2500  LET T$ = "IQ1":X3 = X3 + 1:LW = X3:ZK = ZU: GOSUB 2700
2510  GOSUB 2690
2520  LET FC = VC:LEV = VD:CB = VE
2530  NEXT IT
2540  LET T$ = L16$:L16$ = L2$:L2$ = T$:XY =  - 1:X2 =  - 1:C2 =  - 10:C3 = 0
2550  PRINT D$;"OPEN";L16$;",D2": PRINT D$;"DELETE";L16$: PRINT D$;"CLOSE";L16$
2560  LET T$ = "IQ1":X3 = X3 + 1:LW = X3:ZK = ZU: GOSUB 2700
2570  LET FI = VC:NB = VF
2580  FOR I = 1 TO VF:FS(I) = VH(I): NEXT I
2590  NEXT IM
2600  PRINT D$;"OPEN DAT3,D2": PRINT D$;"DELETE DAT3"
2610  PRINT D$;"OPEN DAT3": PRINT D$;"WRITE DAT3"
2620  PRINT X4: PRINT ZP: PRINT ZQ: PRINT XZ: PRINT LV: PRINT X8
2630  FOR I = 0 TO X4: PRINT ZP(I): NEXT I
2640  FOR I = 0 TO X8: PRINT ZT(I): NEXT I
2650  PRINT D$;"CLOSE DAT3"
2660  PRINT D$;"OPEN STIFF1,D2": PRINT D$;"DELETE STIFF1": PRINT D$;"CLOSE STIFF1"
2670  PRINT D$;"OPEN STIFF2,D2": PRINT D$;"DELETE STIFF2": PRINT D$;"CLOSE STIFF2"
2680  PRINT D$;"RUN BACKSUB,D1"
2690  LET XD = VH(1):XF = VH(2):XE = VH(3):EX = VH(4):PL = VH(5):PQ = VH(6):MQ = VH
    (7):NQ = VH(8):XM = VH(9):MO = VH(10):AP = VH(11):BP = VH(12):MAY = VH(13): RETURN

2700  PRINT D$;"OPEN";T$;",L";ZK;",D1": PRINT D$;"READ";T$;",R";LW
2710  INPUT VC,VD,VE,VF
2720  FOR I = 1 TO VF: INPUT VH(I): NEXT I
2730  PRINT D$;"CLOSE";T$: RETURN
2740  FOR I = 1 TO MR:N = NO(I) + I
2750  IF LF(D1 + I) <  > 1 THEN 2770
2760  LET A(N) = 1.E + 35:SD = SD + 1: IF SD(SD) <  > 0 THEN B(I,LV) = SD(SD) + 1.E
    + 35
2770  NEXT I: RETURN
2780  LET U1 = U2 + 1:U2 = U1:UW = UU:US = (UQ - U1 + 1) * UI
2790  FOR I = U1 TO UM:UW = UW + US: IF UW > UT THEN 2810
2800  LET U2 = U2 + 1:US = US - UI: NEXT I: GOTO 2820
2810  LET UW = UW - US
2820  LET UW = UW - UU:U2 = U2 - 1: RETURN
2830  LET NO(1) = 0: IF N = 1 THEN  RETURN
2840  LET L = N
2850  FOR I = 2 TO N:L = L - 1:NO(I) = NO(I - 1) + L: NEXT I: RETURN
```

```
2860  LET H2 = 1:MR = QM: IF MR = 0 THEN 2890
2870  LET MR = 0:U1 = IU:U2 = MR:UM = QM:UQ = QN:UI = FI:UU = 0:UW = JQ:UT = XW: GOSUB
   2780
2880  LET IU = U1:MR = U2:QM = UM:QN = UQ:FI = UI:JQ = UW:XW = UT
2890  LET S7 = 0: IF UDE = 1 AND MR = QM AND IM < KSUB THEN S7 = 1
2900  REM     ::::::::::::::::::::::::::::::::::::::::::::::::::
2910  REM     :: GENERATES A POINTER  NO ARRAY FOR IDENTIFYING ::
2920  REM     :: THE DIAGONAL ELEMENT LOCATIONS OF STIFFNESS   ::
2930  REM     :: IN FRONT                                      ::
2940  REM     ::::::::::::::::::::::::::::::::::::::::::::::::::
2950  LET N = FC: GOSUB 2830
2960  LET FC = N:KR = MR + 1: RETURN
2970  IF IM = 1 AND LZ > 1 AND MT > 1 THEN X2 = X2 - 1
2980  IF KSUB = 2 AND IM = KSUB AND LZ > 1 AND C3 = LZ - 1 THEN X2 = X2 + 1
2990  LET X2 = X2 + 1:LW = X2:ZA = ZB(LW): IF C2 = X2 THEN 3040
3000  PRINT D$;"OPEN";T$;",L";ZA;",D2": PRINT D$;"READ";T$;",R";LW
3010  INPUT ZLB,I1,I2,MD,WN,GLV
3020  FOR I = 1 TO WN: INPUT S(I): NEXT I
3030  PRINT D$;"CLOSE";T$:C2 = X2: IF KSUB = 2 AND IM = KSUB AND GLV <  > LB THEN C
   3 = C3 + GLV
3040  LET LV = GLV:LX = 0:J = WN - (I2 + 1) * LV
3050  IF IM = 1 AND MT > 1 THEN J = WN - (I2 + 1) * LV - (LB - 1) * I2 * LV
3060  FOR M = I1 TO I2:LM = LF(M):J = J + LV
3070  IF IM = 1 AND MT > 1 THEN 3120
3080  IF IM = 1 OR LV = 1 THEN 3110
3090  IF KSUB = 2 AND IM = KSUB AND GLV <  > LB THEN B(LM,1) = B(LM,1) + S(J + LZ +
   GLV - C3): GOTO 3140
3100  LET K = 1:B(LM,K) = B(LM,K) + S(J + LZ): GOTO 3140
3110  FOR K = 1 TO LV:B(LM,K) = B(LM,K) + S(J + K): NEXT K: GOTO 3140
3120  LET L = (LZ - 1) * LV * I2
3130  FOR K = 1 TO LV:B(LM,K) = B(LM,K) + S(J + L + K): NEXT K
3140  FOR N = M TO MD:LN = LF(N)
3150  IF LM <  = LN THEN I = LM
3160  IF LN <  = LM THEN I = LN
3170  IF LM =  > LN THEN K = NO(I) + LM
3180  IF LN =  > LM THEN K = NO(I) + LN
3190  LET LX = LX + 1:L = LX:A(K) = A(K) + S(L): NEXT N
3200  NEXT M
3210  IF IM > 1 THEN LV = 1
3220  RETURN
3230  LET MX = QN - MR: IF QN =  > FC THEN 3350
3240  LET NR = FC - MR:AO(1) = 0: IF QN = 1 THEN 3270
3250  LET L = QN
3260  FOR I = 2 TO QN:L = L - 1:AO(I) = AO(I - 1) + L: NEXT I
3270  IF MX < 2 THEN 3350
3280  REM     ::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
3290  REM     :: PACK THE STIFFNESS IN ARRAY  A  FOR SAVE ON DATA FILE ::
3300  REM     ::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
3310  LET L = QN
```

```
3320  FOR I = 2 TO QN:VO = NO(I)
3330  FOR J = I TO QN:L = L + 1:A(L) = A(VO + J): NEXT J
3340  NEXT I
3350  LET CI = 1:JJ = IR - 1:U2 = JJ:UM = JR:UQ = NQ:UU = 6:UT = XB:UI = FI: GOSUB
    2780
3360  LET II = U1:JJ = U2:NY = UW:IX = II - MR:KX = JJ - MR: IF IX = 1 THEN XL = 0
3370  LET XL = XL + 1:K = 0
3380  FOR I = IE + 1 TO B9:K = K + 1:A(K) = A(I): NEXT I
3390  FOR I = MR + 1 TO GN
3400  FOR J = 1 TO LV:NY = NY + 1:A(NY) = B(I,J): NEXT J
3410  NEXT I
3420  PRINT D$;"OPEN";T$;",L";ZK;",D2": PRINT D$;"WRITE";T$;",R";LW
3430  PRINT XL: PRINT IX: PRINT KI: PRINT MX: PRINT NY: PRINT LV
3440  FOR K = 1 TO NY: PRINT A(K): NEXT K
3450  PRINT D$;"CLOSE";T$
3460  FOR KJ = 1 TO B9:A(KJ) = 0: NEXT KJ
3470  FOR I = MR + 1 TO B8
3480  FOR J = 1 TO LV:B(I,J) = 0: NEXT J
3490  NEXT I: RETURN
3500  LET IE = MR * QN - (MR * (MR - 1)) / 2:L = 0
3510  FOR I = 1 TO MR:VO = NO(I)
3520  FOR J = I TO QN:L = L + 1:A(L) = A(VO + J): NEXT J
3530  NEXT I
3540  LET EB = IE * FB:EJ = IE
3550  IF IM > 1 THEN 3590
3560  LET X4 = X4 + 1:LW = X4:J = 2:ZP(LW) = (EJ + (MR * LV)) * 20: IF LW <  = 0 THEN
    ZP = ZP(LW): GOTO 3620
3570  IF ZP(LW) < ZP(LW - 1) THEN ZP(LW) = ZP(LW - 1)
3580  LET ZP = ZP(LW): GOTO 3620
3590  LET X8 = X8 + 1:LW = X8:J = 1:ZT(LW) = (EJ + (MR * LV)) * 20: IF LW <  = 0 THEN
    ZP = ZT(LW): GOTO 3620
3600  IF ZT(LW) < ZT(LW - 1) THEN ZT(LW) = ZT(LW - 1)
3610  LET ZP = ZT(LW)
3620  PRINT D$;"OPEN";T$;",L";ZP;",D";J: PRINT D$;"WRITE";T$;",R";LW
3630  PRINT MR: PRINT QN: PRINT EJ: PRINT LV
3640  FOR I = 1 TO EJ: PRINT A(I): NEXT I
3650  FOR I = 1 TO MR
3660  FOR J = 1 TO LV: PRINT B(I,J): NEXT J
3670  NEXT I
3680  PRINT D$;"CLOSE";T$: RETURN
3690  LET HM = NO(M):HN = NO(N):HC = A(HM + M):A(HM + M) = A(HN + N):A(HN + N) = HC

3700  FOR J = 1 TO LV:HC = B(M,J):B(M,J) = B(N,J):B(N,J) = HC: NEXT J
3710  IF M = 1 THEN 3740
3720  LET IY = M - 1
3730  FOR I = 1 TO IY:HIS = NO(I):HC = A(HIS + M):A(HIS + M) = A(HIS + N):A(HIS + N
    ) = HC: NEXT I
3740  LET F1 = M + 1: IF F1 = N THEN 3770
3750  LET F2 = N - 1
```

```
3760  FOR J = F1 TO F2:FS = NO(J):HC = A(HM + J):A(HM + J) = A(FS + N):A(FS + N) =
    HC: NEXT J
3770  IF N = EB THEN  RETURN
3780  LET F1 = N + 1
3790  FOR J = F1 TO EB:HC = A(HM + J):A(HM + J) = A(HN + J):A(HN + J) = HC: NEXT J:
     RETURN
3800  FOR I = ZS TO JR: IF MR <  = I - 1 THEN J8 = MR
3810  IF I - 1 < MR THEN J8 = I - 1
3820  LET IK = NO(I)
3830  FOR J = 1 TO J8:JK = NO(J):P =  - A(JK + I) / A(JK + J)
3840  FOR K = 1 TO LV:B(I,K) = B(I,K) + B(J,K) * P: NEXT K
3850  FOR K = I TO ON:A(IK + K) = A(IK + K) + A(JK + K) * P: NEXT K
3860  NEXT J
3870  NEXT I: RETURN
```

```
1000  HOME :D$ =  CHR$ (4): GOTO 1100: REM    ** BACKSUB **
1010  PRINT "------------": PRINT
1020  PRINT "   BACKSUB   ": PRINT
1030  PRINT "------------": PRINT : RETURN
1040  REM
1050  REM     ::::::::::::::::::::::::::::::::::::::::::::::::
1060  REM     :: THIS SUBPROGRAM PERFORMS BACKSUBSTITUTION ::
1070  REM     :: TO FIND NODAL DISPLACEMENT                ::
1080  REM     ::::::::::::::::::::::::::::::::::::::::::::::::
1090  REM
1100  GOSUB 1010: PRINT D$;"OPEN DAT1,D2": PRINT D$;"READ DAT1"
1110  INPUT XC,PT,EL,MN,NF,MI,PS,NS,KSUB,JZ,ZT,ZQ,ZA,E3,YT,MT,LU,BN,TE
1120  PRINT D$;"CLOSE DAT1"
1130  PRINT D$;"OPEN DAT2": PRINT D$;"READ DAT2"
1140  INPUT GS,S2,S4,ZU,B8
1150  PRINT D$;"CLOSE DAT2"
1160  DIM ZP(EL),ZT(KSUB * BN)
1170  PRINT D$;"OPEN DAT3": PRINT D$;"READ DAT3"
1180  INPUT X4,ZP,ZQ,XZ,LV,X8
1190  FOR I = 0 TO X4: INPUT ZP(I): NEXT I
1200  FOR I = 0 TO X8: INPUT ZT(I): NEXT I
1210  PRINT D$;"CLOSE DAT3"
1220  LET X4 = X4 + 1:ZR =  - 1:ZS =  - 1:ZT = ZT + 1:XZ = XZ + 1:X1 =  - 1:X8 = X8
      + 1:B3 = JZ * (2 * JZ + 1) + (2 * JZ * BN):C9 = (B8 - 2 * JZ) * (B8 + BN): IF
    B3 < C9 THEN B3 = C9
1230  LET TM = B3 + (B8 * BN * 2) + (2 * JZ * BN) + JZ + (2 * EL) + (2 * KSUB * BN)
      + BN + (NS * 3):CPU = 6000: IF TM < CPU THEN 1270
1240  VTAB 8: PRINT  TAB( 4)"USED MEMORY IN BACKSUB      = ";TM
1250  PRINT : PRINT  TAB( 4)"CORE MEMORY                 = ";CPU
1260  PRINT : PRINT  TAB( 4)"IT IS OVER CORE MEMORY      = ";TM - CPU: PRINT : GOSUB
    1290: END
1270  DIM A(B3),X(B8,BN),E(2 * JZ,BN),SM(JZ),FS(BN),CD(NS,3),Y(B8,BN),ZD(EL + KSUB *
    BN),LF(GS)
1280  PRINT D$;"OPEN DISPL,D2": PRINT D$;"DELETE DISPL": PRINT D$;"CLOSE DISPL": GOTO
    1300
1290  PRINT "--------------------------------------": RETURN
1300  DEF  FN WH(I) = QN * I - (I * (I - 1)) / 2 + 1
1310  REM     ::::::::::::::::::::::::::::::::::::::::::::::::::::
1320  REM     :: LOOP FROM SUBSTRUCTURING LEVEL 1 TO LEVEL KSUB ::
1330  REM     ::::::::::::::::::::::::::::::::::::::::::::::::::::
1340  FOR IM = KSUB TO 1 STEP  - 1: HOME : GOSUB 1010: VTAB 8: GOSUB 1290: PRINT
1350  PRINT  TAB( 5)"SUBSTRUCTURING IN LEVEL ";IM: PRINT : GOSUB 1290
1360  PRINT : PRINT : PRINT  TAB( 16)"";: FLASH : PRINT "RUNNING": NORMAL :PG = 0:C
    3 = 0: GOTO 1480
1370  LET ZT = ZT - 1
1380  PRINT D$;"OPEN SUBST,L";YT;",D2": PRINT D$;"READ SUBST,R";ZT
1390  INPUT NB,JS,NP
1400  FOR I = 1 TO NB: INPUT FS(I): NEXT I
```

```
1410  IF JS = 0 THEN 1430
1420  FOR I = 1 TO JS: INPUT SM(I): NEXT I
1430  IF NP = 0 THEN 1470
1440  FOR I = 1 TO NP
1450  FOR J = 1 TO 3: INPUT CD(I,J): NEXT J
1460  NEXT I
1470  PRINT D$;"CLOSE SUBST": RETURN
1480  GOSUB 1370
1490  REM     ::::::::::::::::::::::::::::::::::
1500  REM     :: LOOP OVER EACH SUBSTRUCTURE ::
1510  REM     ::::::::::::::::::::::::::::::::::
1520  FOR IT = NB TO 1 STEP  - 1:LB = FS(IT):PG = PG + 1:D2 = 0
1530  IF LB < 0 THEN PG = PG - 1: GOTO 2290
1540  IF LB > 0 AND PG > 1 THEN PG = 1: GOSUB 1370
1550  IF IM = KSUB THEN 1700
1560  LET ZR = ZR + 1:LW = ZR:ZD = ZD(LW)
1570  PRINT D$;"OPEN DISPL,L";ZD;",D2": PRINT D$;"READ DISPL,R";LW
1580  INPUT D3,VV
1590  FOR I = 1 TO D3
1600  FOR J = 1 TO VV: INPUT Y(I,J): NEXT J
1610  NEXT I
1620  PRINT D$;"CLOSE DISPL": IF KSUB = 1 THEN 1700
1630  IF VV = NB - IT + 1 THEN VV = VV - C3: GOTO 1700
1640  FOR I = 1 TO D3:K = 0
1650  FOR J = IT TO VV:K = K + 1:Y(I,K) = Y(I,J): NEXT J
1660  NEXT I:ZR = ZR - 1:VV = NB - IT + 1 - C3:C3 = C3 + VV
1670  REM     ::::::::::::::::::::::::::::::
1680  REM     :: LOOP OVER EACH ELEMENT ::
1690  REM     ::::::::::::::::::::::::::::::
1700  FOR LZ = 1 TO LB:XZ = XZ - 1:LW = XZ
1710  PRINT D$;"OPEN IQ1,L";ZU;",D1": PRINT D$;"READ IQ1,R";LW
1720  INPUT QM,D1,QN,CT
1730  FOR I = 1 TO CT: INPUT LF(I): NEXT I
1740  PRINT D$;"CLOSE IQ1"
1750  IF D2 <  = D1 THEN D2 = D1
1760  IF QM = QN THEN 1840
1770  REM     ::::::::::::::::::::::::::::::::::::::::::::
1780  REM     :: INTERCHANGES ROWS OF DISPLACEMENTS ::
1790  REM     :: ACCOMODATE TO CURRENT FRONT        ::
1800  REM     ::::::::::::::::::::::::::::::::::::::::::::
1810  FOR J = 1 TO VV
1820  FOR I = QM + 1 TO QN:L = LF(D1 + I):X(I,J) = Y(L,J): NEXT I
1830  NEXT J
1840  IF QM <  = 0 THEN 1940
1850  LET T$ = "QM": IF IM = 1 THEN X4 = X4 - 1:LW = X4:J = 2:ZP = ZP(LW): GOTO 187
   0
1860  LET X8 = X8 - 1:LW = X8:J = 1:ZP = ZT(LW)
```

```
1870  PRINT D$;"OPEN";T$;",L";ZP;",D";J: PRINT D$;"READ";T$;",R";LW
1880  INPUT MR,QN,EJ,VV
1890  FOR I = 1 TO EJ: INPUT A(I): NEXT I
1900  FOR I = 1 TO MR
1910  FOR J = 1 TO VV: INPUT X(I,J): NEXT J
1920  NEXT I
1930  PRINT D$;"CLOSE";T$
1940  IF QM <  = 0 THEN 2070
1950  REM     ::::::::::::::::::::::::::::::::::::::::::::::
1960  REM     :: BACKSUBSTITUTION TO FIND DISPLACEMENTS ::
1970  REM     ::::::::::::::::::::::::::::::::::::::::::::::
1980  FOR N = 1 TO VV: IF QM = QN THEN 2020
1990  FOR I = 1 TO QM:K =  FN WH(I - 1) - I
2000  FOR J = QM + 1 TO QN:X(I,N) = X(I,N) - A(J + K) * X(J,N): NEXT J
2010  NEXT I
2020  LET M = QM + 1:I =  FN WH(QM - 1):X(QM,N) = X(QM,N) / A(I): IF QM = 1 THEN 20
    60
2030  FOR L = 2 TO QM:I = M - L:J1 = I + 1:K =  FN WH(I - 1) - I
2040  FOR J = J1 TO QM:X(I,N) = X(I,N) - A(J + K) * X(J,N): NEXT J
2050  LET K =  FN WH(I - 1):X(I,N) = X(I,N) / A(K): NEXT L
2060  NEXT N
2070  FOR J = 1 TO VV
2080  FOR I = 1 TO D1:L = LF(I):E(I,J) = X(L,J): NEXT I
2090  FOR I = 1 TO QN:Y(I,J) = X(I,J): NEXT I
2100  NEXT J
2110  IF IM = 1 THEN 2150
2120  FOR J = 1 TO VV:K = LB - LZ + 1
2130  FOR I = 1 TO D1:E(I,K) = E(I,J): NEXT I
2140  NEXT J
2150  IF IM > 1 AND LZ < LB THEN 2280
2160  LET LV = VV: IF IM > 1 THEN LV = LB
2170  LET ZS = ZS + 1:LW = ZS:ZD(LW) = D2 * LV * 20 + 10: IF LW <  = 0 THEN 2220
2180  IF ZD(LW) < ZD(LW - 1) THEN ZD(LW) = ZD(LW - 1)
2190  REM     ::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
2200  REM     :: SAVES DISPLACEMENT ARRAY IN DATA FILE " DISPL " ::
2210  REM     ::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
2220  LET ZD = ZD(LW): PRINT D$;"OPEN DISPL,L";ZD;",D2": PRINT D$;"WRITE DISPL,R";L
    W
2230  PRINT D2: PRINT LV
2240  FOR I = 1 TO D2
2250  FOR J = 1 TO LV: PRINT E(I,J): NEXT J
2260  NEXT I
2270  PRINT D$;"CLOSE DISPL"
2280  NEXT LZ
2290  NEXT IT
2300  NEXT IM
```

```
2310  PRINT D$;"OPEN DAT4,D2": PRINT D$;"DELETE DAT4"
2320  PRINT D$;"OPEN DAT4": PRINT D$;"WRITE DAT4"
2330  PRINT ZS: PRINT NB: PRINT JS: PRINT NS
2340  FOR I = 1 TO NB: PRINT FS(I): NEXT I
2350  FOR I = 1 TO JS: PRINT SM(I): NEXT I
2360  FOR I = 1 TO NS
2370  FOR J = 1 TO 3: PRINT CD(I,J): NEXT J
2380  NEXT I
2390  FOR I = 0 TO ZS: PRINT ZD(I): NEXT I
2400  PRINT D$;"CLOSE DAT4"
2410  PRINT D$;"OPEN QM,D2": PRINT D$;"DELETE QM": PRINT D$;"CLOSE QM"
2420  IF MN = 4 THEN  PRINT D$;"RUN STRESS(Q4),D1"
2430  IF MN = 8 THEN  PRINT D$;"RUN STRESS(Q8),D1"
```

```
1000  HOME :D$ =  CHR$ (4): GOTO 1100: REM    ** STRESS **
1010  PRINT "--------------": PRINT
1020  PRINT "    STRESS    ": PRINT
1030  PRINT "--------------": PRINT : RETURN
1040  REM
1050  REM     ::::::::::::::::::::::::::::::::::::::::
1060  REM     :: THIS SUBPROGRAM CALCULATES STRESSES ::
1070  REM     :: AT THE GAUSS POINTS                 ::
1080  REM     ::::::::::::::::::::::::::::::::::::::::
1090  REM
1100  GOSUB 1010: PRINT D$;"OPEN DAT1,D2": PRINT D$;"READ DAT1"
1110  INPUT XC,PT,EL,MN,NF,MI,PS,NS,KSUB,JZ,ZT,ZQ,ZA,E3,YT,MT,LU,BN,TE
1120  PRINT D$;"CLOSE DAT1"
1130  DIM E(2 * JZ,BN),SM(JZ),FS(BN),CD(NS,3),SN(NS,2),ZD(EL + KSUB * BN)
1140  DIM IX(EL),T(PT),IQ(MN),UX(PT),UY(PT),R(PT),Z(PT),ST(3,9),JAC(2,2),C(3,3),SIG
    (6),YM(15),PR(15),DEN(15),AP(15),B(2 * MN)
1150  PRINT D$;"OPEN DAT4,D2": PRINT D$;"READ DAT4"
1160  INPUT ZS,NB,JS,NS
1170  FOR I = 1 TO NB: INPUT FS(I): NEXT I
1180  FOR I = 1 TO JS: INPUT SM(I): NEXT I
1190  FOR I = 1 TO NS
1200  FOR J = 1 TO 3: INPUT CD(I,J): NEXT J
1210  NEXT I
1220  FOR I = 0 TO ZS: INPUT ZD(I): NEXT I
1230  PRINT D$;"CLOSE DAT4":X1 =  - 1:ZS = ZS + 1:Q1 = 0:HZ = 160
1240  LET ELE$ = "ELE1": IF KSUB > 2 THEN ELE$ = "ELE3"
1250  VTAB 15: PRINT  TAB( 16)"";: FLASH : PRINT "RUNNING": NORMAL
1260  FOR I = 1 TO PT:UX(I) = 0:UY(I) = 0: NEXT I
1270  FOR I = 1 TO 4: READ EP(I): NEXT I
1280  DATA  -1,-1,1,1
1290  FOR I = 1 TO 4: READ XI(I): NEXT I
1300  DATA -1,1,1,-1
1310  PRINT D$;"OPEN MAT,D2": PRINT D$;"READ MAT"
1320  INPUT DM,TT
1330  FOR I = 1 TO DM: INPUT YM(I),PR(I),DEN(I),AP(I): NEXT I
1340  PRINT D$;"CLOSE MAT"
1350  PRINT D$;"OPEN LOCATE,D2": PRINT D$;"READ LOCATE"
1360  FOR I = 1 TO PT: INPUT R(I): NEXT I
1370  FOR I = 1 TO PT: INPUT Z(I): NEXT I
1380  FOR I = 1 TO PT: INPUT T(I): NEXT I
1390  FOR I = 1 TO EL: INPUT IX(I): NEXT I
1400  PRINT D$;"CLOSE LOCATE"
1410  REM     ::::::::::::::::::::::::::::::::
1420  REM     :: LOOP OVER EACH SUBSTRUCTURE ::
1430  REM     ::::::::::::::::::::::::::::::::
1440  FOR IT = 1 TO NB:LB = FS(IT): IF LB < 0 THEN 2300
```

```
1450  REM     ::::::::::::::::::::::::::::
1460  REM     :: LOOP OVER EACH ELEMENT ::
1470  REM     ::::::::::::::::::::::::::::
1480  FOR LZ = 1 TO LB:Q1 = Q1 + 1:Q = IX(Q1)
1490  LET ZS = ZS - 1:LW = ZS:ZD = ZD(LW): PRINT D$;"OPEN DISPL,L";ZD;",D2": PRINT
   D$;"READ DISPL,R";LW
1500  INPUT D1,LV
1510  FOR I = 1 TO D1
1520  FOR J = 1 TO LV: INPUT E(I,J): NEXT J
1530  NEXT I
1540  PRINT D$;"CLOSE DISPL"
1550  LET X1 = X1 + 1:LW = X1: PRINT D$;"OPEN";ELE$;",L";ZQ;",D2": PRINT D$;"READ";
   ELE$;",R";LW
1560  INPUT NE
1570  FOR I = 1 TO NE: INPUT IQ(I): NEXT I
1580  PRINT D$;"CLOSE";ELE$
1590  REM     ::::::::::::::::::::::::::::::::::::::
1600  REM     :: FIND MATRIX OF ELASTIC STIFFNESS ::
1610  REM     :: FOR PLANE STRESS & PLANE  STRAIN ::
1620  REM     ::::::::::::::::::::::::::::::::::::::
1630  IF TE = 2 THEN 1660
1640  LET F = YM(Q) / (1 + PR(Q)):G = F * PR(Q) / (1 - 2 * PR(Q)):H = F + G
1650  LET C(1,1) = H:C(1,2) = G:C(1,3) = 0:C(2,1) = G:C(2,2) = H:C(2,3) = 0:C(3,1) =
   0:C(3,2) = 0:C(3,3) = F / 2: GOTO 1680
1660  LET F = YM(Q) / (1 + PR(Q)):G = YM(Q) / (1 - PR(Q)):H = F * G / YM(Q)
1670  LET C(1,1) = H:C(1,2) = H * PR(Q):C(1,3) = 0:C(2,1) = H * PR(Q):C(2,2) = H:C(
   2,3) = 0:C(3,1) = 0:C(3,2) = 0:C(3,3) = F / 2
1680  FOR LY = 1 TO LV: IF LY = 1 THEN 1700
1690  FOR I = 1 TO NE:IQ(I) = IQ(I) - FS(IT - 1 + LY): NEXT I
1700  FOR I = 1 TO D1:B(I) = E(I,LY): NEXT I
1710  FOR I = 1 TO NE:J = IQ(I):K = 2 * I - 1:L = 2 * I:UX(J) = B(K):UY(J) = B(L): NEXT
   I
1720  FOR I = 1 TO 6:SIG(I) = 0: NEXT I
1730  FOR I = 1 TO NE:J = IQ(I):RR(I) = R(J):ZZ(I) = Z(J): NEXT I
1740  LET N = Q1 + (LY - 1) * LB:CX = (RR(2) + RR(1)) / 2:CY = (ZZ(3) + ZZ(2)) / 2:
   NA = 1
1750  REM     ::::::::::::::::::::::::::::::::::::::::::::::
1760  REM     :: FIND SHAPE FUNCTIONS & SHAPE FUNCTION ::
1770  REM     :: DERIVATIVES FOR 4-NODES ELEMENT     ::
1780  REM     ::::::::::::::::::::::::::::::::::::::::::::::
1790  FOR L = 1 TO 4:ADUM = 0.25:BDUM = 0.25:EN(L) = 4 * ADUM * BDUM:NXI(L) = XI(L)
    * BDUM:NYE(L) = EP(L) * ADUM: NEXT L
1800  REM     ::::::::::::::::::::::::::::::::::
1810  REM     :: INITIALIZE JACOBIAN MATRIX & ::
1820  REM     :: STRAIN-DISPLACEMENT MATRIX   ::
1830  REM     ::::::::::::::::::::::::::::::::::
1840  FOR I = 1 TO 2
1850  FOR J = 1 TO 2:JAC(I,J) = 0: NEXT J
1860  NEXT I
```

```
1870  FOR I = 1 TO 3
1880  FOR J = 1 TO 9:ST(I,J) = 0: NEXT J
1890  NEXT I
1900  REM     ::::::::::::::::::::::::::::::::::::::::
1910  REM     :: FIND JACOBIAN AND ITS DETERMINANT ::
1920  REM     ::::::::::::::::::::::::::::::::::::::::
1930  FOR L = 1 TO NE:JAC(1,1) = JAC(1,1) + NXI(L) * RR(L):JAC(1,2) = JAC(1,2) + NX
   I(L) * ZZ(L):JAC(2,1) = JAC(2,1) + NYE(L) * RR(L):JAC(2,2) = JAC(2,2) + NYE(L) *
   ZZ(L): NEXT L
1940  LET DJ = JAC(1,1) * JAC(2,2) - JAC(2,1) * JAC(1,2):ADUM = JAC(1,1) / DJ
1950  LET JAC(1,1) = JAC(2,2) / DJ:JAC(1,2) =  - JAC(1,2) / DJ:JAC(2,1) =  - JAC(2,
   1) / DJ:JAC(2,2) = ADUM
1960  FOR J = 1 TO NE:L = 2 * J:K = L - 1
1970  LET ST(1,K) = JAC(1,1) * NXI(J) + JAC(1,2) * NYE(J):ST(2,L) = JAC(2,1) * NXI(
   J) + JAC(2,2) * NYE(J):ST(3,K) = ST(2,L):ST(3,L) = ST(1,K): NEXT J
1980  REM     ::::::::::::::::::::::::::::::::::
1990  REM     :: FIND INITIAL THERMAL STRESSES ::
2000  REM     ::::::::::::::::::::::::::::::::::
2010  FOR L = 1 TO NE:I = IQ(L): IF TE = 1 THEN 2030
2020  LET EX(L) = AP(IX) * (T(I) - TT):EY(L) = AP(IX) * (T(I) - TT):ES(L) = 0: GOTO
   2040
2030  LET EX(L) = (1 + PR(IX)) * AP(IX) * (T(I) - TT):EY(L) = (1 + PR(IX)) * AP(IX)
    * (T(I) - TT):ES(L) = 0
2040  LET ST(1,9) = ST(1,9) + EN(L) * EX(L):ST(2,9) = ST(2,9) + EN(L) * EY(L):ST(3,
   9) = ST(3,9) + EN(L) * ES(L): NEXT L
2050  REM     ::::::::::::::::::::::::::::::::::::::::::::
2060  REM     :: CALCULATES STRESS & PRINCIPAL STRESS ::
2070  REM     :: AT GAUSS POINTS                       ::
2080  REM     ::::::::::::::::::::::::::::::::::::::::::::
2090  FOR I = 1 TO 3:RP(I) = 0
2100  FOR J = 1 TO NE:II = 2 * J:RP(I) = RP(I) + ST(I,II) * B(II) + ST(I,II - 1) *
   B(II - 1): NEXT J
2110  LET RP(I) = RP(I) - ST(I,9): NEXT I
2120  FOR I = 1 TO 3
2130  FOR J = 1 TO 3:SIG(I) = SIG(I) + C(I,J) * RP(J): NEXT J
2140  NEXT I
2150  LET CC = (SIG(1) + SIG(2)) / 2:BB = (SIG(1) - SIG(2)) / 2:CR =  SQR (BB ^ 2 +
   SIG(3) ^ 2):SIG(4) = CC + CR:SIG(5) = CC - CR:SIG(6) = 0
2160  IF BB = 0 AND SIG(3) = 0 THEN 2270
2170  LET SIG(6) = 28.647889757 *  ATN (SIG(3) / BB)
2180  IF SIG(1) - SIG(2) < 0 AND SIG(3) < 0 THEN SIG(6) = SIG(6) - 90
2190  IF SIG(1) - SIG(2) < 0 AND SIG(3) > 0 THEN SIG(6) = SIG(6) + 90
2200  REM     ::::::::::::::::::::::::::::::::
2210  REM     :: SAVE ELEMENT STRESS       ::
2220  REM     :: INTO DATA FILE " STRESS " ::
2230  REM     ::::::::::::::::::::::::::::::::
2240  PRINT D$;"OPEN STRESS,L";HZ;",D2": PRINT D$;"WRITE STRESS,R";N
2250  PRINT CX: PRINT CY
2260  FOR I = 1 TO 6: PRINT SIG(I): NEXT I
2270  PRINT D$;"CLOSE STRESS"
```

```
2280  NEXT LY
2290  NEXT LZ:Q1 = Q1 + (LV - 1) * LB
2300  NEXT IT
2310  PRINT D$;"OPEN SUPSET,D2": PRINT D$;"READ SUPSET"
2320  FOR I = 1 TO NS: INPUT SN(I,1),SN(I,2): NEXT I
2330  PRINT D$;"CLOSE SUPSET"
2340  REM     ::::::::::::::::::::::::::::::::::::::::
2350  REM     :: EVALUATES DISPLACEMENT AT SUPPORTS ::
2360  REM     ::::::::::::::::::::::::::::::::::::::::
2370  FOR I = 1 TO NS
2380  FOR J = 1 TO PT: IF CD(I,1) <  > J THEN 2410
2390  IF CD(I,2) = 1 AND SN(I,1) = 0 THEN UX(J) = 0
2400  IF CD(I,3) = 1 AND SN(I,2) = 0 THEN UY(J) = 0
2410  NEXT J
2420  NEXT I
2430  PRINT D$;"OPEN IQ1,D1": PRINT D$;"DELETE IQ1": PRINT D$;"CLOSE IQ1"
2440  PRINT D$;"OPEN QM,D1": PRINT D$;"DELETE QM": PRINT D$;"CLOSE QM"
2450  PRINT D$;"OPEN SUBST,D2": PRINT D$;"DELETE SUBST": PRINT D$;"CLOSE SUBST"
2460  PRINT D$;"OPEN MAT,D2": PRINT D$;"DELETE MAT": PRINT D$;"CLOSE MAT"
2470  PRINT D$;"OPEN SUPSET,D2": PRINT D$;"DELETE SUPSET": PRINT D$;"CLOSE SUPSET"
2480  PRINT D$;"OPEN NEWSUP,D2": PRINT D$;"DELETE NEWSUP": PRINT D$;"CLOSE NEWSUP"
2490  PRINT D$;"OPEN LOCATE,D2": PRINT D$;"DELETE LOCATE": PRINT D$;"CLOSE LOCATE"
2500  PRINT D$;"OPEN ELE1,D2": PRINT D$;"DELETE ELE1": PRINT D$;"CLOSE ELE1": IF KS
    UB <  = 2 THEN 2520
2510  PRINT D$;"OPEN ELE3,D2": PRINT D$;"DELETE ELE3": PRINT D$;"CLOSE ELE3"
2520  PRINT D$;"OPEN DISPL,D2": PRINT D$;"DELETE DISPL": PRINT D$;"CLOSE DISPL"
2530  PRINT D$;"OPEN DAT1,D2": PRINT D$;"DELETE DAT1": PRINT D$;"CLOSE DAT1"
2540  PRINT D$;"OPEN DAT2,D2": PRINT D$;"DELETE DAT2": PRINT D$;"CLOSE DAT2"
2550  PRINT D$;"OPEN DAT3,D2": PRINT D$;"DELETE DAT3": PRINT D$;"CLOSE DAT3"
2560  PRINT D$;"OPEN DAT4,D2": PRINT D$;"DELETE DAT4": PRINT D$;"CLOSE DAT4"
2570  REM     ::::::::::::::::::::::::::::::
2580  REM     :: SAVE NODAL DISPLACEMENTS ::
2590  REM     :: INTO DATA FILE " DIST "  ::
2600  REM     ::::::::::::::::::::::::::::::
2610  PRINT D$;"OPEN DIST,D2": PRINT D$;"WRITE DIST"
2620  PRINT PT: PRINT EL
2630  FOR I = 1 TO PT: PRINT UX(I): PRINT UY(I): NEXT I
2640  PRINT D$;"CLOSE DIST"
2650  PRINT D$;"RUN PRINT RESULT(Q4),D1"
```

```
1000  REM    **  BACKPRINT  **
1010  HOME :D$ =  CHR$ (4): POKE 1656 + SLOT,72: VTAB 4: GOTO 1110
1020  PRINT "--------------------------": PRINT
1030  PRINT "   PRINT DISPL. & STRESS   ": PRINT
1040  PRINT "--------------------------": PRINT : RETURN
1050  REM
1060  REM      ::::::::::::::::::::::::::::::::::::::::::::::::
1070  REM      :: THIS SUBPROGRAM PRINTS NODAL DISPLACEMENTS ::
1080  REM      :: AND ELEMENT STRESSES AT GAUSS POINTS       ::
1090  REM      ::::::::::::::::::::::::::::::::::::::::::::::::
1100  REM
1110  GOSUB 1020: PRINT D$;"OPEN DIST,D2": PRINT D$;"READ DIST"
1120  INPUT PT,EL
1130  DIM UX(PT),UY(PT),W(6)
1140  FOR I = 1 TO PT: INPUT UX(I),UY(I): NEXT I
1150  PRINT D$;"CLOSE DIST":HZ = 160: GOTO 1180
1160  VTAB 23: PRINT "                                              ": RETURN
1170  VTAB 24: PRINT "Press RETURN key to continue ..... ";: INPUT " ";B$: RETURN
1180  VTAB 24: PRINT "Do you display RESULT on monitor?(Y/N)";: INPUT "";B$: IF B$ <
    > "Y" AND B$ <  > "N" THEN 1180
1190  IF B$ = "N" THEN 1210
1200  LET CP = 0:Z = 5: GOSUB 1250
1210  HOME : GOSUB 1020: VTAB 24: PRINT "Do you print RESULT on printer ?(Y/N)";: INPUT
   "";B$: IF B$ <  > "Y" AND B$ <  > "N" THEN 1210
1220  IF B$ = "N" THEN 1240
1230  LET CP = 1:Z = 18: PRINT D$;"PR#1": PRINT  CHR$ (15): GOSUB 1250: PRINT D$;"P
   R#0"
1240  PRINT D$;"RUN OPTIONS,D1"
1250  LET J = 0:K = 3
1260  FOR I = 1 TO PT
1270  IF CP = 1 THEN 1310
1280  LET K = K + 1: IF K > 22 THEN  GOSUB 1170:K = 4: GOTO 1350
1290  IF I = 1 THEN 1350
1300  VTAB K: GOTO 1360
1310  LET J = J + 1: IF J < 51 THEN 1330
1320  FOR L = 1 TO 7: PRINT : NEXT L:J = 1
1330  IF J <  > 1 THEN 1360
1340  FOR L = 1 TO 6: PRINT : NEXT L
1350  HOME : PRINT  TAB( Z - 4)"NODE No.     X-Displ.       Y-Displ.": PRINT  TAB(
   Z - 4)"--------   ------------  ------------": PRINT
1360  PRINT  SPC( Z -  LEN ( STR$ (I)));I;:UX(I) =  INT (UX(I) * 100000 + 0.5) / 10
   0000:UY(I) =  INT (UY(I) * 100000 + 0.5) / 100000
1370  PRINT  SPC( 17 -  LEN ( STR$ (UX(I))));UX(I);: PRINT  SPC( 15 -  LEN ( STR$ (
   UY(I))));UY(I)
1380  NEXT I: IF CP = 0 THEN  GOSUB 1170: GOTO 1400
1390  FOR L = J + 10 TO 66: PRINT : NEXT L
1400  HOME :J = 0: IF CP = 0 THEN 1590
```

```
1410  FOR I = 1 TO EL: PRINT D$;"OPEN STRESS,L";HZ;",D2": PRINT D$;"READ STRESS,R";
   I
1420  INPUT XC,YC
1430  FOR M = 1 TO 6: INPUT W(M): NEXT M
1440  PRINT D$;"CLOSE STRESS"
1450  LET XC =  INT (XC * 1000 + 0.5) / 1000:YC =  INT (YC * 1000 + 0.5) / 1000
1460  FOR M = 1 TO 6:W(M) =  INT (W(M) * 100000 + 0.5) / 100000: NEXT M
1470  LET J = J + 1: IF J < 51 THEN 1490
1480  FOR L = 1 TO 7: PRINT : NEXT L:J = 1
1490  IF J <  > 1 THEN 1530
1500  FOR L = 1 TO 6: PRINT : NEXT L
1510  PRINT  TAB( 15)"EL.No.     X        Y       X-STRESS       Y-STRESS";: PRINT
   "       XY-STRESS     MAX.-STRESS     MIN.-STRESS       ANGLE"
1520  PRINT  TAB( 15)"------ ------- -------  ----------     ----------";: PRINT
   "     -----------    -----------     -----------       -----": PRINT
1530  PRINT  SPC( 18 -  LEN ( STR$ (I)));I;: PRINT  SPC( 10 -  LEN ( STR$ (XC)));XC
   ;: PRINT  SPC( 9 -  LEN ( STR$ (YC)));YC;
1540  PRINT  SPC( 13 -  LEN ( STR$ (W(1))));W(1);: PRINT  SPC( 15 -  LEN ( STR$ (W(
   2))));W(2);: PRINT  SPC( 16 -  LEN ( STR$ (W(3))));W(3);
1550  PRINT  SPC( 16 -  LEN ( STR$ (W(4))));W(4);: PRINT  SPC( 16 -  LEN ( STR$ (W(
   5))));W(5);: PRINT  SPC( 12 -  LEN ( STR$ (W(6))));W(6)
1560  NEXT I
1570  FOR L = J + 10 TO 66: PRINT : NEXT L
1580  IF CP = 1 THEN  RETURN
1590  FOR I = 1 TO EL
1600  PRINT D$;"OPEN STRESS,L";HZ;",D2": PRINT D$;"READ STRESS,R";I
1610  INPUT XC,YC
1620  FOR M = 1 TO 6: INPUT W(M): NEXT M
1630  PRINT D$;"CLOSE STRESS": HOME
1640  LET XC =  INT (XC * 1000 + 0.5) / 1000:YC =  INT (YC * 1000 + 0.5) / 1000
1650  FOR M = 1 TO 6:W(M) =  INT (W(M) * 100000 + 0.5) / 100000: NEXT M
1660  PRINT "----------------------------------": PRINT : PRINT "    STRESS OF ELE
   MENT No. ";I: PRINT : PRINT "----------------------------------": PRINT
1670  VTAB 8: PRINT "X-AXIS       = ";XC: VTAB 10: PRINT "Y-AXIS       = ";YC
1680  VTAB 12: PRINT "X-STRESS     = ";W(1): VTAB 14: PRINT "Y-STRESS     = ";W(2):
    VTAB 16: PRINT "XY-STRESS    = ";W(3)
1690  VTAB 18: PRINT "MAX.-STRESS  = ";W(4): VTAB 20: PRINT "MIN.-STRESS  = ";W(5):
    VTAB 22: PRINT "ANGLE        = ";W(6)
1700  GOSUB 1170: NEXT I
1710  RETURN
```

```
1000  HOME :D$ =  CHR$ (4): GOTO 1100: REM    ** STRESS **
1010  PRINT "--------------": PRINT
1020  PRINT "    STRESS    ": PRINT
1030  PRINT "--------------": PRINT : RETURN
1040  REM
1050  REM     ::::::::::::::::::::::::::::::::::::::::::::
1060  REM     :: THIS SUBPROGRAM CALCULATES STRESSES AT ::
1070  REM     :: THE GAUSS POINTS                       ::
1080  REM     ::::::::::::::::::::::::::::::::::::::::::::
1090  REM
1100  GOSUB 1010: PRINT D$;"OPEN DAT1,D2": PRINT D$;"READ DAT1"
1110  INPUT XC,PT,EL,MN,NF,MI,PS,NS,KSUB,JZ,ZT,ZQ,ZA,E3,YT,MT,LU,BN,TE
1120  PRINT D$;"CLOSE DAT1"
1130  DIM E(2 * JZ,BN),SM(JZ),FS(BN),CD(NS,3),SN(NS,2),ZD(EL + KSUB * BN),PL(3,3)
1140  DIM IX(EL),T(PT),IQ(MN),UX(PT),UY(PT),R(PT),Z(PT),ST(3,17),XC(4),YC(4),W(4,6)
   ,JAC(2,2),C(3,3),SIG(6),YM(15),PR(15),DEN(15),AP(15),B(2 * MN)
1150  PRINT D$;"OPEN DAT4,D2": PRINT D$;"READ DAT4"
1160  INPUT ZS,NB,JS,NS
1170  FOR I = 1 TO NB: INPUT FS(I): NEXT I
1180  FOR I = 1 TO JS: INPUT SM(I): NEXT I
1190  FOR I = 1 TO NS
1200  FOR J = 1 TO 3: INPUT CD(I,J): NEXT J
1210  NEXT I
1220  FOR I = 0 TO ZS: INPUT ZD(I): NEXT I
1230  PRINT D$;"CLOSE DAT4":X1 =  - 1:ZS = ZS + 1:Q1 = 0:NG = 2:HZ = 640
1240  LET ELE$ = "ELE1": IF KSUB > 2 THEN ELE$ = "ELE3"
1250  VTAB 15: PRINT  TAB( 16)"";: FLASH : PRINT "RUNNING": NORMAL
1260  FOR I = 1 TO 3
1270  FOR J = 1 TO 3: READ PL(I,J): NEXT J
1280  NEXT I
1290  DATA 0,-.577350269190,-.774596669241,0,.577350269190,0,0,0,.774596669241
1300  FOR I = 1 TO PT:UX(I) = 0:UY(I) = 0: NEXT I
1310  FOR I = 1 TO 4: READ EP(I): NEXT I
1320  DATA -1,1,-1,1
1330  FOR I = 1 TO 4: READ XI(I): NEXT I
1340  DATA -1,-1,1,1
1350  PRINT D$;"OPEN MAT,D2": PRINT D$;"READ MAT"
1360  INPUT DM,TT
1370  FOR I = 1 TO DM: INPUT YM(I),PR(I),DEN(I),AP(I): NEXT I
1380  PRINT D$;"CLOSE MAT"
1390  PRINT D$;"OPEN LOCATE,D2": PRINT D$;"READ LOCATE"
1400  FOR I = 1 TO PT: INPUT R(I): NEXT I
1410  FOR I = 1 TO PT: INPUT Z(I): NEXT I
1420  FOR I = 1 TO PT: INPUT T(I): NEXT I
1430  FOR I = 1 TO EL: INPUT IX(I): NEXT I
1440  PRINT D$;"CLOSE LOCATE"
```

```
1450  REM     ::::::::::::::::::::::::::::::::::
1460  REM     :: LOOP OVER EACH SUBSTRUCTURE ::
1470  REM     ::::::::::::::::::::::::::::::::::
1480  FOR IT = 1 TO NB:LB = FS(IT): IF LB < 0 THEN 2490
1490  REM     ::::::::::::::::::::::::::::
1500  REM     :: LOOP OVER EACH ELEMENT ::
1510  REM     ::::::::::::::::::::::::::::
1520  FOR LZ = 1 TO LB:Q1 = Q1 + 1:Q = IX(Q1)
1530  LET ZS = ZS - 1:LW = ZS:ZD = ZD(LW): PRINT D$;"OPEN DISPL,L";ZD;",D2": PRINT
    D$;"READ DISPL,R";LW
1540  INPUT D1,LV
1550  FOR I = 1 TO D1
1560  FOR J = 1 TO LV: INPUT E(I,J): NEXT J
1570  NEXT I
1580  PRINT D$;"CLOSE DISPL"
1590  LET X1 = X1 + 1:LW = X1: PRINT D$;"OPEN";ELE$;",L";ZQ;",D2": PRINT D$;"READ";
    ELE$;",R";LW
1600  INPUT NE
1610  FOR I = 1 TO NE: INPUT IQ(I): NEXT I
1620  PRINT D$;"CLOSE";ELE$
1630  REM     ::::::::::::::::::::::::::::::::::
1640  REM     :: FIND MATRIX OF ELASTIC  FOR ::
1650  REM     :: PLANE STRESS & PLANE STRAIN ::
1660  REM     ::::::::::::::::::::::::::::::::::
1670  IF TE = 2 THEN 1700
1680  LET F = YM(Q) / (1 + PR(Q)):G = F * PR(Q) / (1 - 2 * PR(Q)):H = F + G
1690  LET C(1,1) = H:C(1,2) = G:C(1,3) = 0:C(2,1) = G:C(2,2) = H:C(2,3) = 0:C(3,1) =
    0:C(3,2) = 0:C(3,3) = F / 2: GOTO 1720
1700  LET F = YM(Q) / (1 + PR(Q)):G = YM(Q) / (1 - PR(Q)):H = F * G / YM(Q)
1710  LET C(1,1) = H:C(1,2) = H * PR(Q):C(1,3) = 0:C(2,1) = H * PR(Q):C(2,2) = H:C(
    2,3) = 0:C(3,1) = 0:C(3,2) = 0:C(3,3) = F / 2
1720  FOR LY = 1 TO LV: IF LY = 1 THEN 1740
1730  FOR I = 1 TO NE:IQ(I) = IQ(I) - FS(IT - 1 + LY): NEXT I
1740  FOR I = 1 TO D1:B(I) = E(I,LY): NEXT I
1750  FOR I = 1 TO 4
1760  FOR J = 1 TO 6:W(I,J) = 0: NEXT J
1770  NEXT I
1780  FOR I = 1 TO NE:J = IQ(I):K = 2 * I - 1:L = 2 * I:UX(J) = B(K):UY(J) = B(L): NEXT
    I
1790  FOR I = 1 TO NE:J = IQ(I):RR(I) = R(J):ZZ(I) = Z(J): NEXT I
1800  REM     ::::::::::::::::::::::::::::::::::::::::::::
1810  REM     :: LOOP FOR AREA NUMERICAL INTEGRATION ::
1820  REM     ::::::::::::::::::::::::::::::::::::::::::::
1830  FOR NA = 1 TO NG:PX = PL(NA,NG)
1840  FOR NC = 1 TO NG:PE = PL(NC,NG)
1850  FOR I = 1 TO 6:SIG(I) = 0: NEXT I
1860  LET N = Q1 + (LY - 1) * LB:I = (NA - 1) * 2 + NC:J = 0.577350269190:XC(I) = R
    R(5) + J * XI(I) * (RR(5) - RR(1)):YC(I) = ZZ(6) + J * EP(I) * (ZZ(6) - ZZ(2))
```

```
1870  REM     ::::::::::::::::::::::::::::::::::::::::::::::::
1880  REM     :: FIND THE SHAPE FUNCTIONS AND SHAPE FUNCTION ::
1890  REM     :: DERIVATIVES FOR 8-NODES ELEMENT             ::
1900  REM     ::::::::::::::::::::::::::::::::::::::::::::::::
1910  LET EN(1) = (1 - PX) * (1 - PE) * ( - PX - PE - 1) / 4:EN(2) = (1 + PX) * (1 -
    PE) * (PX - PE - 1) / 4:EN(3) = (1 + PX) * (1 + PE) * (PX + PE - 1) / 4:EN(4) =
    (1 - PX) * (1 + PE) * ( - PX + PE - 1) / 4
1920  LET EN(5) = (1 - PX ^ 2) * (1 - PE) / 2:EN(6) = (1 + PX) * (1 - PE ^ 2) / 2:E
    N(7) = (1 - PX ^ 2) * (1 + PE) / 2:EN(8) = (1 - PX) * (1 - PE ^ 2) / 2
1930  LET NXI(1) = (PE + 2 * PX - 2 * PX * PE - PE ^ 2) / 4:NXI(2) = ( - PE + 2 * P
    X - 2 * PX * PE + PE ^ 2) / 4:NXI(3) = (PE + 2 * PX + 2 * PX * PE + PE ^ 2) / 4
    :NXI(4) = ( - PE + 2 * PX + 2 * PX * PE - PE ^ 2) / 4
1940  LET NXI(5) = ( - PX + PX * PE):NXI(6) = (1 - PE ^ 2) / 2:NXI(7) = ( - PX - PX
     * PE):NXI(8) = ( - 1 + PE ^ 2) / 2
1950  LET NYE(1) = (PX + 2 * PE - 2 * PX * PE - PX ^ 2) / 4:NYE(2) = ( - PX + 2 * P
    E + 2 * PX * PE - PX ^ 2) / 4:NYE(3) = (PX + 2 * PE + 2 * PX * PE + PX ^ 2) / 4
    :NYE(4) = ( - PX + 2 * PE - 2 * PX * PE + PX ^ 2) / 4
1960  LET NYE(5) = ( - 1 + PX ^ 2) / 2:NYE(6) = ( - PE - PX * PE):NYE(7) = (1 - PX ^
    2) / 2:NYE(8) = ( - PE + PX * PE)
1970  REM     ::::::::::::::::::::::::::::::::::
1980  REM     :: INITIALIZE JACOBIAN MATRIX & ::
1990  REM     :: STRAIN-DISPLACEMENT MATRIX   ::
2000  REM     ::::::::::::::::::::::::::::::::::
2010  FOR I = 1 TO 2
2020  FOR J = 1 TO 2:JAC(I,J) = 0: NEXT J
2030  NEXT I
2040  FOR I = 1 TO 3
2050  FOR J = 1 TO 17:ST(I,J) = 0: NEXT J
2060  NEXT I
2070  REM     ::::::::::::::::::::::::::::::::::::::::
2080  REM     :: FIND JACOBIAN AND ITS DETERMINANT ::
2090  REM     ::::::::::::::::::::::::::::::::::::::::
2100  FOR L = 1 TO NE:JAC(1,1) = JAC(1,1) + NXI(L) * RR(L):JAC(1,2) = JAC(1,2) + NX
    I(L) * ZZ(L):JAC(2,1) = JAC(2,1) + NYE(L) * RR(L):JAC(2,2) = JAC(2,2) + NYE(L) *
    ZZ(L): NEXT L
2110  LET DJ = JAC(1,1) * JAC(2,2) - JAC(2,1) * JAC(1,2):ADUM = JAC(1,1) / DJ
2120  LET JAC(1,1) = JAC(2,2) / DJ:JAC(1,2) =  - JAC(1,2) / DJ:JAC(2,1) =  - JAC(2,
    1) / DJ:JAC(2,2) = ADUM
2130  FOR J = 1 TO NE:L = 2 * J:K = L - 1
2140  LET ST(1,K) = JAC(1,1) * NXI(J) + JAC(1,2) * NYE(J):ST(2,L) = JAC(2,1) * NXI(
    J) + JAC(2,2) * NYE(J):ST(3,K) = ST(2,L):ST(3,L) = ST(1,K): NEXT J
2150  REM     ::::::::::::::::::::::::::::::::::
2160  REM     :: FIND INITIAL THERMAL STRESSES ::
2170  REM     ::::::::::::::::::::::::::::::::::
2180  FOR L = 1 TO NE:I = IQ(L): IF TE = 1 THEN 2200
2190  LET EX(L) = AP(IX) * (T(I) - TT):EY(L) = AP(IX) * (T(I) - TT):ES(L) = 0: GOTO
    2210
2200  LET EX(L) = (1 + PR(IX)) * AP(IX) * (T(I) - TT):EY(L) = (1 + PR(IX)) * AP(IX)
     * (T(I) - TT):ES(L) = 0
```

```
2210  LET ST(1,17) = ST(1,17) + EN(L) * EX(L):ST(2,17) = ST(2,17) + EN(L) * EY(L):S
    T(3,17) = ST(3,17) + EN(L) * ES(L): NEXT L
2220  REM     ::::::::::::::::::::::::::::::::::::::::::::
2230  REM     :: CALCULATES STRESS AND PRINCIPAL STRESS ::
2240  REM     :: AT GAUSS POINTS                        ::
2250  REM     ::::::::::::::::::::::::::::::::::::::::::::
2260  FOR I = 1 TO 3:RP(I) = 0
2270  FOR J = 1 TO NE:II = 2 * J:RP(I) = RP(I) + ST(I,II) * B(II) + ST(I,II - 1) *
    B(II - 1): NEXT J
2280  LET RP(I) = RP(I) - ST(I,17): NEXT I
2290  FOR I = 1 TO 3
2300  FOR J = 1 TO 3:SIG(I) = SIG(I) + C(I,J) * RP(J): NEXT J
2310  NEXT I
2320  LET CC = (SIG(1) + SIG(2)) / 2:BB = (SIG(1) - SIG(2)) / 2:CR =  SQR (BB ^ 2 +
    SIG(3) ^ 2):SIG(4) = CC + CR:SIG(5) = CC - CR:SIG(6) = 0
2330  IF BB = 0 AND SIG(3) = 0 THEN 2370
2340  LET SIG(6) = 28.647889757 *  ATN (SIG(3) / BB)
2350  IF SIG(1) - SIG(2) < 0 AND SIG(3) < 0 THEN SIG(6) = SIG(6) - 90
2360  IF SIG(1) - SIG(2) < 0 AND SIG(3) > 0 THEN SIG(6) = SIG(6) + 90
2370  LET I = (NA - 1) * 2 + NC
2380  FOR J = 1 TO 6:W(I,J) = SIG(J): NEXT J
2390  NEXT NC
2400  NEXT NA
2404  REM     ::::::::::::::::::::::::::::::
2405  REM     :: SAVE ELEMENT STRESS       ::
2407  REM     :: INTO DATA FILE " STRESS " ::
2408  REM     ::::::::::::::::::::::::::::::
2410  PRINT D$;"OPEN STRESS,L";HZ;",D2": PRINT D$;"WRITE STRESS,R";N
2420  FOR I = 1 TO 4: PRINT XC(I): PRINT YC(I): NEXT I
2430  FOR I = 1 TO 4
2440  FOR J = 1 TO 6: PRINT W(I,J): NEXT J
2450  NEXT I
2460  PRINT D$;"CLOSE STRESS"
2470  NEXT LY
2480  NEXT LZ:Q1 = Q1 + (LV - 1) * LB
2490  NEXT IT
2500  PRINT D$;"OPEN SUPSET,D2": PRINT D$;"READ SUPSET"
2510  FOR I = 1 TO NS: INPUT SN(I,1),SN(I,2): NEXT I
2520  PRINT D$;"CLOSE SUPSET"
2530  REM     ::::::::::::::::::::::::::::::::::::::::
2540  REM     :: EVALUATES DISPLACEMENT AT SUPPORTS ::
2550  REM     ::::::::::::::::::::::::::::::::::::::::
2560  FOR I = 1 TO NS
2570  FOR J = 1 TO PT: IF CD(I,1) <  > J THEN 2600
2580  IF CD(I,2) = 1 AND SN(I,1) = 0 THEN UX(J) = 0
2590  IF CD(I,3) = 1 AND SN(I,2) = 0 THEN UY(J) = 0
2600  NEXT J
2610  NEXT I
```

```
2620  PRINT D$;"OPEN IQ1,D1": PRINT D$;"DELETE IQ1": PRINT D$;"CLOSE IQ1"
2630  PRINT D$;"OPEN QM,D1": PRINT D$;"DELETE QM": PRINT D$;"CLOSE QM"
2640  PRINT D$;"OPEN SUBST,D2": PRINT D$;"DELETE SUBST": PRINT D$;"CLOSE SUBST"
2650  PRINT D$;"OPEN MAT,D2": PRINT D$;"DELETE MAT": PRINT D$;"CLOSE MAT"
2660  PRINT D$;"OPEN SUPSET,D2": PRINT D$;"DELETE SUPSET": PRINT D$;"CLOSE SUPSET"
2670  PRINT D$;"OPEN NEWSUP,D2": PRINT D$;"DELETE NEWSUP": PRINT D$;"CLOSE NEWSUP"
2680  PRINT D$;"OPEN LOCATE,D2": PRINT D$;"DELETE LOCATE": PRINT D$;"CLOSE LOCATE"
2690  PRINT D$;"OPEN ELE1,D2": PRINT D$;"DELETE ELE1": PRINT D$;"CLOSE ELE1": IF KS
    UB < = 2 THEN 2710
2700  PRINT D$;"OPEN ELE3,D2": PRINT D$;"DELETE ELE3": PRINT D$;"CLOSE ELE3"
2710  PRINT D$;"OPEN DISPL,D2": PRINT D$;"DELETE DISPL": PRINT D$;"CLOSE DISPL"
2720  PRINT D$;"OPEN DAT1,D2": PRINT D$;"DELETE DAT1": PRINT D$;"CLOSE DAT1"
2730  PRINT D$;"OPEN DAT2,D2": PRINT D$;"DELETE DAT2": PRINT D$;"CLOSE DAT2"
2740  PRINT D$;"OPEN DAT3,D2": PRINT D$;"DELETE DAT3": PRINT D$;"CLOSE DAT3"
2750  PRINT D$;"OPEN DAT4,D2": PRINT D$;"DELETE DAT4": PRINT D$;"CLOSE DAT4"
2760  REM     ::::::::::::::::::::::::::::::
2770  REM     :: SAVE NODAL DISPLACEMENTS ::
2780  REM     :: INTO DATA FILE " DIST "  ::
2790  REM     ::::::::::::::::::::::::::::::
2800  PRINT D$;"OPEN DIST,D2": PRINT D$;"WRITE DIST"
2810  PRINT PT: PRINT EL
2820  FOR I = 1 TO PT: PRINT UX(I): PRINT UY(I): NEXT I
2830  PRINT D$;"CLOSE DIST"
2840  PRINT D$;"RUN PRINT RESULT(Q8),D1"
```

```
1000  REM    ** BACKPRINT **
1010  HOME :D$ =  CHR$ (4): POKE 1656 + SLOT,72: VTAB 4: GOTO 1110
1020  PRINT "--------------------------": PRINT
1030  PRINT "   PRINT DISPL. & STRESS   ": PRINT
1040  PRINT "--------------------------": PRINT : RETURN
1050  REM
1060  REM      ::::::::::::::::::::::::::::::::::::::::::::::::
1070  REM      :: THIS SUBPROGRAM PRINTS NODAL DISPLACEMENTS ::
1080  REM      :: AND ELEMENT STRESSES AT GAUSS POINTS       ::
1090  REM      ::::::::::::::::::::::::::::::::::::::::::::::::
1100  REM
1110  GOSUB 1020: PRINT D$;"OPEN DIST,D2": PRINT D$;"READ DIST"
1120  INPUT PT,EL
1130  DIM UX(PT),UY(PT),XC(4),YC(4),N(4,6)
1140  FOR I = 1 TO PT: INPUT UX(I),UY(I): NEXT I
1150  PRINT D$;"CLOSE DIST":HZ = 640: GOTO 1180
1160  VTAB 23: PRINT "                                              ": RETURN
1170  VTAB 24: PRINT "Press RETURN key to continue ..... ";: INPUT " ";B$: RETURN
1180  VTAB 24: PRINT "Do you display RESULT on monitor?(Y/N)";: INPUT "";B$: IF B$ <
     > "Y" AND B$ <  > "N" THEN 1180
1190  IF B$ = "N" THEN 1210
1200  LET CP = 0:Z = 5: GOSUB 1250
1210  HOME : GOSUB 1020: VTAB 24: PRINT "Do you print RESULT on printer ?(Y/N)";: INPUT
    "";B$: IF B$ <  > "Y" AND B$ <  > "N" THEN 1210
1220  IF B$ = "N" THEN 1240
1230  LET CP = 1:Z = 18: PRINT D$;"PR#1": PRINT  CHR$ (15): GOSUB 1250: PRINT D$;"P
    R#0"
1240  PRINT D$;"RUN OPTIONS,D1"
1250  LET J = 0:K = 3
1260  FOR I = 1 TO PT
1270  IF CP = 1 THEN 1310
1280  LET K = K + 1: IF K > 22 THEN  GOSUB 1170:K = 4: GOTO 1350
1290  IF I = 1 THEN 1350
1300  VTAB K: GOTO 1360
1310  LET J = J + 1: IF J < 51 THEN 1330
1320  FOR L = 1 TO 7: PRINT : NEXT L:J = 1
1330  IF J <  > 1 THEN 1360
1340  FOR L = 1 TO 6: PRINT : NEXT L
1350  HOME : PRINT  TAB( Z - 4)"NODE No.     X-Displ.       Y-Displ.": PRINT  TAB(
    Z - 4)"--------   ------------   ------------": PRINT
1360  PRINT  SPC( Z -  LEN ( STR$ (I)));I;:UX(I) =  INT (UX(I) * 100000 + 0.5) / 10
    0000:UY(I) =  INT (UY(I) * 100000 + 0.5) / 100000
1370  PRINT  SPC( 17 -  LEN ( STR$ (UX(I))));UX(I);: PRINT  SPC( 15 -  LEN ( STR$ (
    UY(I))));UY(I)
1380  NEXT I: IF CP = 0 THEN  GOSUB 1170: GOTO 1400
1390  FOR L = J + 10 TO 66: PRINT : NEXT L
1400  HOME :J = 0: IF CP = 0 THEN 1660
```

```
1410  FOR I = 1 TO EL: PRINT D$;"OPEN STRESS,L";HZ;",D2": PRINT D$;"READ STRESS,R";
    I
1420  FOR M = 1 TO 4: INPUT XC(M),YC(M): NEXT M
1430  FOR M = 1 TO 4
1440  FOR N = 1 TO 6: INPUT W(M,N): NEXT N
1450  NEXT M
1460  PRINT D$;"CLOSE STRESS"
1470  FOR M = 1 TO 4:XC(M) =  INT (XC(M) * 1000 + 0.5) / 1000:YC(M) =  INT (YC(M) *
    1000 + 0.5) / 1000: NEXT M
1480  FOR M = 1 TO 4
1490  FOR N = 1 TO 6:W(M,N) =  INT (W(M,N) * 100000 + 0.5) / 100000: NEXT N
1500  NEXT M
1510  LET J = J + 1: IF J < 9 THEN 1530
1520  FOR L = 1 TO 9: PRINT : NEXT L:J = 1
1530  IF J <  > 1 THEN 1570
1540  FOR L = 1 TO 6: PRINT : NEXT L
1550  PRINT  TAB( 15)" G. P.     X        Y       I-STRESS       Y-STRESS";: PRINT
    "       XY-STRESS     MAX.-STRESS     MIN.-STRESS       ANGLE"
1560  PRINT  TAB( 15)"------  -------  -------   ----------     ----------";: PRINT
    "     -----------    -----------     -----------       -----": PRINT
1570  PRINT  TAB( 16)"ELEMENT No. ";I
1580  FOR K = 1 TO 4
1590  PRINT  SPC( 18 -  LEN ( STR$ (K)));K;: PRINT  SPC( 10 -  LEN ( STR$ (XC(K))))
    ;XC(K);: PRINT  SPC( 9 -  LEN ( STR$ (YC(K))));YC(K);
1600  PRINT  SPC( 13 -  LEN ( STR$ (W(K,1))));W(K,1);: PRINT  SPC( 15 -  LEN ( STR$
    (W(K,2))));W(K,2);: PRINT  SPC( 16 -  LEN ( STR$ (W(K,3))));W(K,3);
1610  PRINT  SPC( 16 -  LEN ( STR$ (W(K,4))));W(K,4);: PRINT  SPC( 16 -  LEN ( STR$
    (W(K,5))));W(K,5);: PRINT  SPC( 12 -  LEN ( STR$ (W(K,6))));W(K,6)
1620  NEXT K: PRINT
1630  NEXT I
1640  FOR L = J * 6 + 10 TO 66: PRINT : NEXT L
1650  IF CP = 1 THEN  RETURN
1660  FOR I = 1 TO EL
1670  PRINT D$;"OPEN STRESS,L";HZ;",D2": PRINT D$;"READ STRESS,R";I
1680  FOR M = 1 TO 4: INPUT XC(M),YC(M): NEXT M
1690  FOR M = 1 TO 4
1700  FOR N = 1 TO 6: INPUT W(M,N): NEXT N
1710  NEXT M
1720  PRINT D$;"CLOSE STRESS"
1730  FOR M = 1 TO 4:XC(M) =  INT (XC(M) * 1000 + 0.5) / 1000:YC(M) =  INT (YC(M) *
    1000 + 0.5) / 1000: NEXT M
1740  FOR M = 1 TO 4
1750  FOR N = 1 TO 6:W(M,N) =  INT (W(M,N) * 100000 + 0.5) / 100000: NEXT N
1760  NEXT M
```

```
1770  FOR K = 1 TO 4: HOME
1780  PRINT "----------------------------------": PRINT : PRINT "  STRESS OF ELEME
   NT No. ";I;: PRINT "  G.P. =";K: PRINT : PRINT "------------------------------
   ----": PRINT
1790  VTAB 8: PRINT "X-AXIS       = ";XC(K): VTAB 10: PRINT "Y-AXIS       = ";YC(K)

1800  VTAB 12: PRINT "X-STRESS     = ";W(K,1): VTAB 14: PRINT "Y-STRESS     = ";W(K
   ,2): VTAB 16: PRINT "XY-STRESS    = ";W(K,3)
1810  VTAB 18: PRINT "MAX.-STRESS  = ";W(K,4): VTAB 20: PRINT "MIN.-STRESS  = ";W(K
   ,5): VTAB 22: PRINT "ANGLE        = ";W(K,6)
1820  GOSUB 1170: NEXT K
1830  NEXT I
1840  RETURN
```

ภาคผนวก ข.

วิธีการป้อนข้อมูลและแสดงตัวอย่างข้อมูล

ข.1 ความนำ

โปรแกรมที่ใช้สำหรับวิเคราะห์ผนังต้านแรงเฉือนนี้ เขียนขึ้นเพื่อใช้กับเครื่องไมโครคอมพิวเตอร์ Apple II โดยเขียนโปรแกรมเป็นภาษาแอปเปิลซอฟท์เบสิค ถ้าใช้กับเครื่องไมโครคอมพิวเตอร์ชนิดอื่นคำสั่งบางคำสั่งที่ใช้ในโปรแกรมอาจจะต้องเปลี่ยนแปลงให้เหมาะสมกับเครื่อง อนึ่งโปรแกรมนี้เขียนขึ้นโดยใช้ดิสค์ไดรฟ์สองตัว

ในโปรแกรมจะมีโปรแกรม OPTIONS ให้เลือกว่าต้องการให้โปรแกรมอะไรทำงาน โปรแกรมย่อยต่าง ๆ จะมีดังนี้

1. INPUT DATA
2. PRINT DATA
3. PLOT
4. CHANGE DATA
5. EXECUTE
6. PRINT RESULT (Q4)
7. PRINT RESULT (Q8)
8. EXIT

ถ้าต้องการจะหยุดการคำนวณและการทำงานอื่น ๆ จะเลือกหัวข้อที่ 8

ข.2 การป้อนข้อมูล

โปรแกรมย่อย INPUT DATA เป็นโปรแกรมสำหรับป้อนข้อมูลต่าง ๆ ของโครงสร้างย่อย ชิ้นส่วนย่อยและอื่น ๆ ลำดับการป้อนข้อมูลมีดังนี้

1. ป้อนข้อมูลรายละเอียดของโครงสร้าง

   1.1 จำนวนของขั้ว (PT)

   1.2 จำนวนของชิ้นส่วนย่อย (EL)

1.3 จำนวนของชิ้นส่วนย่อยที่เก็บไว้ในแผ่นดิสก์ (NF)

1.4 จำนวนขั้วของชิ้นส่วนย่อยในโครงสร้างย่อยระดับที่ 1 (MN)

1.5 จำนวนขั้วของโครงสร้างย่อยที่มากที่สุด (IZ)

1.6 จำนวนชิ้นส่วนย่อยในแต่ละระดับของโครงสร้างย่อยที่มากที่สุด (PS)

1.7 จำนวนขั้วที่ต้องกำหนดสภาพเงื่อนไข (NS)

1.8 จำนวนโครงสร้างย่อยในระดับที่ 1 (BN)

1.9 จำนวนระดับของโครงสร้างย่อยที่ใช้ (KSUB)

1.10 จำนวนชนิดของวัสดุ (DM)

1.11 ชนิดของระนาบของชิ้นส่วนย่อย (TE)

1.12 จำนวนโครงสร้างย่อยที่เหมือนกันที่มากที่สุด (LU)

1.13 จำนวนน้ำหนักกระทำเป็นจุด (E3)

1.14 อุณหภูมิอ้างอิง (RT)

1.15 ชิ้นส่วนย่อยในแต่ละโครงสร้างย่อยเหมือนกันหรือไม่ โดยตอบว่า Y(es) หรือ N(o)

1.16 ตอบคำถามว่ายอมรับข้อมูลชุดนี้หรือไม่ โดยตอบ Y(es) หรือ N(o)

หมายเหตุ ขั้วที่กำหนดสภาพเงื่อนไขต้องเป็นขั้วของโครงสร้างย่อยด้วย

2. ป้อนข้อมูลเกี่ยวกับโครงสร้างย่อย โดยเริ่มจากโครงสร้างย่อยระดับที่ 1 ถึงระดับที่ KSUB และป้อนข้อมูลเกี่ยวกับขั้วที่ต้องกำหนดสภาพเงื่อนไข

2.1 จำนวนโครงสร้างย่อยในระดับที่ IM (NB)

2.2 จำนวนชิ้นส่วนย่อยในแต่ละโครงสร้างย่อย FS(I)

2.3 ตอบคำถามว่ายอมรับข้อมูลชุดนี้หรือไม่ โดยตอบ U(es) หรือ N(o)

2.4 จำนวนขั้วของโครงสร้างย่อยที่ IT (JS)

2.5 จำนวนขั้วที่ต้องกำหนดสภาพเงื่อนไขในโครงสร้างย่อยระดับที่ IM (NP)

2.6 ตอบคำถามว่ายอมรับข้อมูลชุดนี้หรือไม่ โดยตอบ Y(es) หรือ N(o)

2.7 ถ้าจำนวนขั้วที่ต้องกำหนดสภาพเงื่อนไข $\neq 0$ ให้ข้ามไปป้อนข้อมูลข้อ 2.9

2.8 หมายเลขขั้วของโครงสร้างย่อยทีละขั้ว SM(I) และข้ามไปข้อ 2.12

2.9 หมายเลขขั้วที่กำหนดสภาพเงื่อนไขทีละขั้ว CD (I,1)

2.10 สภาพเงื่อนไขในทิศทาง X CD (I,2)

2.11 สภาพเงื่อนไขในทิศทาง Y CD (I,3)

2.12 ตอบคำถามว่ายอมรับข้อมูลชุดนี้หรือไม่ โดยตอบ Y(es) หรือ N(o)

2.13 ทำการป้อนข้อมูลซ้ำจาก 2.4 ถึง 2.12 และจาก 2.1 ถึง 2.12

หมายเหตุ ถ้าระดับขั้นความเสรีในทิศทางใดเป็นอิสระ จะป้อนสภาพเงื่อนไขเป็น 0

ถ้าระดับขั้นความเสรีในทิศทางใดถูกบังคับ จะป้อนสภาพเงื่อนไขเป็น 1

3. ป้อนข้อมูลเกี่ยวกับคุณสมบัติของวัสดุแต่ละชนิด

3.1 โมดูลัสความยืดหยุ่น YM(I)

3.2 อัตราส่วนพอยซอง PR(I)

3.3 ความหนาแน่นของวัสดุ DEN(I)

3.4 สัมประสิทธิ์ของอุณหภูมิ AP(I)

3.5 ตอบคำถามว่ายอมรับข้อมูลชุดนี้หรือไม่ โดยตอบ Y(es) หรือ N(o)

จำนวนชุดข้อมูลที่ป้อนขึ้นอยู่กับจำนวนของชนิดวัสดุดังข้อมูลที่ป้อนในข้อ 1.10 แต่ไม่เกิน 15 ชนิด

4. ป้อนข้อมูลเกี่ยวกับโคออร์ดิเนทของขั้วแต่ละขั้ว

4.1 ต้องการป้อนข้อมูลด้วยตัวเองใช่หรือไม่ โดยตอบ Y(es) หรือ N(o)

4.2 ถ้าตอบว่าใช่ ให้ข้ามไปป้อนข้อมูลข้อ 4.7

4.3 ความกว้างและความสูงของผนังด้านแรงเฉือน (W,H)

4.4 จำนวนช่องที่แบ่งทั้งด้านความกว้างและความสูง (SW, SH)

4.5 ตอบคำถามว่ายอมรับข้อมูลชุดนี้หรือไม่ โดยตอบ Y(es) หรือ N(o)

4.6 เรียงหมายเลขขั้วจากน้อยไปมากตามแกน X ใช่หรือไม่ โดยตอบว่า Y(es) หรือ N(o) และข้ามไปป้อนข้อมูลข้อ 5

4.7 หมายเลขขั้ว (I)

4.8 ลำดับขั้นการเพิ่มขึ้น DZ(2)

4.9 x-โคออร์ดิเนท X(I)

4.10 y-โคออร์ดิเนท Y(I)

4.11 ตอบคำถามว่ายอมรับข้อมูลชุดนี้หรือไม่ โดยตอบ Y(es) หรือ N(o)

หมายเลขขั้วจะป้อนจากน้อยไปหามาก โดยโคออร์ดิเนทของหมายเลขขั้วที่ไม่ได้ป้อนจะถูกแบ่งออกเป็นช่วง ๆ ที่เท่ากันระหว่างโคออร์ดิเนทของหมายเลขขั้วที่ป้อนติดกัน การป้อนโคออร์ดิเนทของขั้วจะป้อนจนถึงขั้วสุดท้าย (PT)

5. ป้อนข้อมูลเกี่ยวกับความเครียดแรกเริ่มอันเนื่องมาจากการเปลี่ยนแปลงของอุณหภูมิที่ต่างจากอุณหภูมิอ้างอิง (RT)

5.1 มีค่าความเครียดเริ่มแรกอันเนื่องมาจากการเปลี่ยนแปลงของอุณหภูมิใช่หรือไม่ โดยตอบ Y(es) หรือ N(o)

5.2 ถ้าตอบว่าไม่ใช่ ให้ข้ามไปป้อนข้อมูลข้อ 6

5.3 อุณหภูมิที่ขั้วต่าง ๆ ของโครงสร้าง T(I)

5.4 ตอบคำถามว่ายอมรับข้อมูลชุดนี้หรือไม่ โดยตอบ Y(es) หรือ N(o)

จำนวนชุดข้อมูลที่ป้อนเท่ากับจำนวนขั้วที่มีอุณหภูมิต่างจากอุณหภูมิอ้างอิง

6. ป้อนข้อมูลเกี่ยวกับการเคลื่อนที่ของขั้วที่ฐาน

6.1 มีการเคลื่อนที่ที่ฐานใช่หรือไม่ โดยตอบ Y(es) หรือ N(o)

6.2 ถ้าตอบว่าไม่ใช่ ให้ข้ามไปป้อนข้อมูลข้อ 7

6.3 การเคลื่อนที่ของขั้วตามแกน x SN(I,1)

6.4 การเคลื่อนที่ของขั้วตามแกน y SN(I,2)

6.5 ตอบคำถามว่ายอมรับข้อมูลชุดนี้หรือไม่ โดยตอบ Y(es) หรือ N(o)

จำนวนชุดข้อมูลที่ป้อนเท่ากับจำนวนขั้วที่ต้องกำหนดสภาพเงื่อนไขดังข้อมูลที่ป้อนในข้อ 1.7

7. ป้อนข้อมูลเกี่ยวกับน้ำหนักกระทำภายนอก แบ่งเป็นน้ำหนักกระทำเป็นจุดและน้ำหนักแผ่สม่ำเสมอ

7.1 จำนวนน้ำหนักกระทำเป็นจุด = 0 ให้ข้ามไปป้อนข้อมูลข้อ 7.5

7.2 เลขที่ขั้วที่น้ำหนักเป็นจุดกระทำ LO(I,1)

7.3 ค่าของน้ำหนักที่กระทำเป็นจุดในแนวแกน x R1(I,1)

7.4 ค่าของน้ำหนักที่กระทำเป็นจุดในแนวแกน y R2(I,1)

7.5 ตอบคำถามว่ายอมรับข้อมูลชุดนี้หรือไม่ โดยตอบ Y(es) หรือ N(o)

จำนวนชุดข้อมูลที่ป้อนเท่ากับจำนวนน้ำหนักกระทำเป็นจุด ดังข้อมูลที่ป้อนในข้อ 1.13

7.6 มีน้ำหนักแผ่สม่ำเสมอกระทำใช่หรือไม่ โดยตอบ Y(es) หรือ N(o)

7.7 ถ้าตอบว่าไม่ใช่ ให้ข้ามไปป้อนข้อมูลข้อ 8

7.8 จำนวนน้ำหนักแผ่สม่ำเสมอ (DL)

7.9 ตอบคำถามว่ายอมรับข้อมูลชุดนี้หรือไม่ โดยตอบ Y(es) หรือ N(o)

7.10 หมายเลขชิ้นส่วนย่อยที่น้ำหนักแผ่สม่ำเสมอมากระทำ ELN(I)

7.11 หมายเลขขั้วของชิ้นส่วนย่อยที่น้ำหนักแผ่สม่ำเสมอมากระทำเรียงตามทิศทางทวนเข็มนาฬิกา DISN(I,J)

7.12 ค่าของน้ำหนักแผ่สม่ำเสมอที่มากระทำในทิศทางตั้งฉากที่ขั้วของชิ้นส่วนย่อย R3(I,J)

7.13 ค่าของน้ำหนักแผ่สม่ำเสมอที่มากระทำสัมผัสกับด้านข้างที่ขั้วของชิ้นส่วนย่อย R4(I,J)

7.14 ตอบคำถามว่ายอมรับข้อมูลชุดนี้หรือไม่ โดยตอบ Y(es) หรือ N(o)

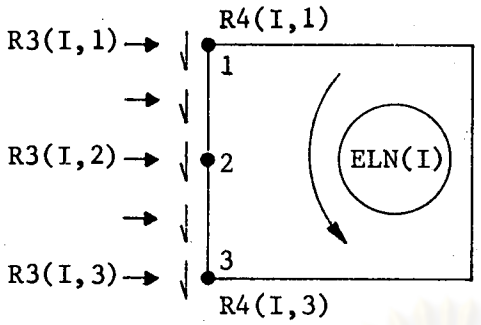
จำนวนชุดข้อมูลที่ป้อนเท่ากับจำนวนน้ำหนักแผ่สม่ำเสมอที่มากระทำ

หมายเหตุ -เครื่องหมายของน้ำหนักที่กระทำเป็นจุด

ในแนวแกน x และ y มีค่าเป็นบวก

ในทิศทางตรงกันข้ามกับแนวแกน x และ y มีค่าเป็นลบ

- เครื่องหมายของน้ำหนักแผ่สม่ำเสมอ



มีค่าเป็นบวก ตามทิศทางดังรูป

มีค่าเป็นลบ ตามทิศทางตรงกันข้ามกับรูป

8. ป้อนข้อมูลเกี่ยวกับขั้วและคุณสมบัติของชิ้นส่วนย่อยในแต่ละโครงสร้างย่อยของโครงสร้างย่อยในระดับที่ 1

8.1 จำนวนขั้วของชิ้นส่วนย่อยมีค่าเท่ากับข้อมูลที่ป้อนในข้อ 1.4

8.2 หมายเลขขั้วของชิ้นส่วนย่อย โดยเรียงตามทิศทางทวนเข็ม IQ(I)

จำนวนชุดข้อมูลที่ป้อนมีจำนวนเท่ากับข้อมูลในข้อ 8.1

8.3 ชนิดของคุณสมบัติของชิ้นส่วนย่อย (IX)

8.4 ความหนาของชิ้นส่วนย่อย (THC)

8.5 ตอบคำถามว่ายอมรับข้อมูลชุดนี้หรือไม่ โดยตอบ Y(es) หรือ N(o)

จำนวนชุดข้อมูลที่ป้อนขึ้นอยู่กับจำนวนชิ้นส่วนย่อยในแต่ละโครงสร้างย่อยของโครงสร้างย่อยในระดับที่ 1

ข.3 ตัวอย่างการป้อนข้อมูล

แสดงตัวอย่างการป้อนข้อมูลในตัวอย่างที่ 1 (หัวข้อ 4.2)

```
------------

 INPUT DATA

------------


  TITLE :    SHEAR WALL

LARGEST NODE NUMBER                  = 100

NUM. OF FINITE ELEMENTS              = 56

NUM. OF FINITE ELEMENTS ON DISK      = 14

NUM. OF NODES/ELEMENT IN LEVEL 1     = 4

MAX. NUM. OF NODES IN SUBST.         = 8

MAX. NUM. OF ELEMENT IN ANY SUBST=   14

MAX. NUM. OF BOUND. CON. NODES       = 4

NUM. OF SUBSTRUCTURES IN LEVEL 1     = 4

NUM. OF SUBSTRUCTURE LEVELS          = 2

NUMBER OF MATERIAL TYPES             = 1

ELEMENT TYPE : (PLANE STRAIN = 1)
             : (PLANE STRESS = 2)    2

MAX. NUM. OF THE SAME SUBST.         = 4

NUM. OF NODAL POINT LOADS            = 4

REFERENCE TEMPERATURE                = 0

DO THE ELEMENT IS THE SAME TYPE

(IN SUBST. FLOOR) ? (Y/N) ....       Y

Do you accept these data ? (Y/N)     Y
```

```
-----------------------------------------
          INPUT CONTROL DATA

           IN SUBST. LEV. 1

-----------------------------------------
Press RETURN key to continue ...

-----------------------------------------
   NUM. OF SUBSTRUCTURE           = 4

-----------------------------------------
Do you accept these data ? (Y/N)    Y

-----------------------------------------
   NUM. OF SUBSTRUCTURE           = 4

-----------------------------------------
NUM. OF ELEMENTS IN SUBST.        = 14
NUM. OF ELEMENTS IN SUBST.        = -24
NUM. OF ELEMENTS IN SUBST.        = -24
NUM. OF ELEMENTS IN SUBST.        = -24
Do you accept these data ? (Y/N)    Y
-----------------------------------------
          INPUT CONTROL DATA

           IN SUBST. NO.  1

-----------------------------------------
Press RETURN key to continue ...

-----------------------------------------
   NUM. OF SUBST. NODES            = 8
   NUM. OF BOUN. CON. NODES        = 0

-----------------------------------------
Do you accept these data ? (Y/N)    Y

-----------------------------------------
   NUM. OF SUBST. NODES     = 8

-----------------------------------------
SUBST. NODE NUMBER           = 1
SUBST. NODE NUMBER           = 2
SUBST. NODE NUMBER           = 3
SUBST. NODE NUMBER           = 4
SUBST. NODE NUMBER           = 25
SUBST. NODE NUMBER           = 26
SUBST. NODE NUMBER           = 27
SUBST. NODE NUMBER           = 28
Do you accept these data ? (Y/N)    Y
```

```
-----------------------------------------
          INPUT CONTROL DATA

           IN SUBST. LEV. 2
-----------------------------------------
Press RETURN key to continue ...

-----------------------------------------
   NUM. OF SUBSTRUCTURE          = 1

-----------------------------------------
Do you accept these data ? (Y/N)    Y

-----------------------------------------
   NUM. OF SUBSTRUCTURE          = 1

-----------------------------------------
NUM. OF ELEMENTS IN SUBST.       = 4
Do you accept these data ? (Y/N)    Y
-----------------------------------------
          INPUT CONTROL DATA

           IN SUBST. NO.  1

-----------------------------------------
Press RETURN key to continue ...

-----------------------------------------
   NUM. OF SUBST. NODES             = 0
   NUM. OF BOUN. CON. NODES         = 4

-----------------------------------------
Do you accept these data ? (Y/N)    Y

-----------------------------------------
  NUM. OF BOUN. CON. NODES     = 4

    ( Free = 0    :   Fixed = 1 )

-----------------------------------------

NODE NUMBER WITH BOUN. CONSTRAIN  = 1
B. C. IN X-Coor.                  = 1
B. C. IN Y-Coor.                  = 1
Do you accept these data ? (Y/N)   Y
NODE NUMBER WITH BOUN. CONSTRAIN  = 2
B. C. IN X-Coor.                  = 1
B. C. IN Y-Coor.                  = 1
Do you accept these data ? (Y/N)   Y
```

```
NODE NUMBER WITH BOUN. CONSTRAIN  = 3
B. C. IN X-Coor.                  = 1
B. C. IN Y-Coor.                  = 1
Do you accept these data ? (Y/N)   Y
NODE NUMBER WITH BOUN. CONSTRAIN  = 4
B. C. IN X-Coor.                  = 1
B. C. IN Y-Coor.                  = 1
Do you accept these data ? (Y/N)   Y

-----------------------------------------

    MATERIAL TYPE   ..... 1

-----------------------------------------

YOUNG'S MODULUS          = 2.812E6
POISSON'S RATIO          = .4
MASS DENSITY             = 2.4
COEF. OF TEMPERATURE     = .0001
Do you accept these data ? (Y/N)   Y

-----------------------------------------

  GLOBAL NODAL COORDINATE

-----------------------------------------

DO YOU WANT TO INPUT NODAL COOR.

BY YOURSELF ?  (Y/N) ......   Y

-----------------------------------------

      INPUT NODAL POINT COORDINATES

-----------------------------------------

  NODE No.              = 1
  GENERATE INCREMENT    = 4
  X - Coordinate        = 0
  Y - Coordinate        = 0
Do you accept these data ? (Y/N)   Y
  NODE No.              = 97
  GENERATE INCREMENT    = 0
  X - Coordinate        = 0
  Y - Coordinate        = 24
Do you accept these data ? (Y/N)   Y
  NODE No.              = 2
  GENERATE INCREMENT    = 4
  X - Coordinate        = 2
  Y - Coordinate        = 0
Do you accept these data ? (Y/N)   Y
  NODE No.              = 98
  GENERATE INCREMENT    = 0
  X - Coordinate        = 2
  Y - Coordinate        = 24
Do you accept these data ? (Y/N)   Y
```

```
  NODE No.                = 3
  GENERATE INCREMENT      = 4
  X - Coordinate          = 4
  Y - Coordinate          = 0
Do you accept these data ? (Y/N)    Y
  NODE No.                = 99
  GENERATE INCREMENT      = 0
  X - Coordinate          = 4
  Y - Coordinate          = 24
Do you accept these data ? (Y/N)    Y
  NODE No.                = 4
  GENERATE INCREMENT      = 4
  X - Coordinate          = 6
  Y - Coordinate          = 0
Do you accept these data ? (Y/N)    Y
  NODE No.                = 100
  GENERATE INCREMENT      = 0
  X - Coordinate          = 6
  Y - Coordinate          = 24
Do you accept these data ? (Y/N)    Y
-----------------------------------------
     DO INITIAL STRAIN EFFECT FROM
     TEMPERATURE ? (Y/N) ......  N
-----------------------------------------
   DO THE SUPPORT HAVE DISPLACEMENT ?
    (Y/N)    ..................    N

-----------------------------------------
INPUT NODAL POINT LOADS IN GLOBAL Coor.

-----------------------------------------
NODE No.                  = 21
LOADING IN X-Coor.        = 20
LOADING IN Y-Coor.        = 0
Do you accept these data ? (Y/N)    Y
NODE No.                  = 45
LOADING IN X-Coor.        = 40
LOADING IN Y-Coor.        = 0
Do you accept these data ? (Y/N)    Y
NODE No.                  = 69
LOADING IN X-Coor.        = 60
LOADING IN Y-Coor.        = 0
Do you accept these data ? (Y/N)    Y
NODE No.                  = 93
LOADING IN X-Coor.        = 60
LOADING IN Y-Coor.        = 0
Do you accept these data ? (Y/N)    Y
-----------------------------------------
     DO YOU INPUT DISTRIBUTED LOADS
      (Y/N)  ................    N
```

```
-----------------------------------------
NUM. OF THE SAME SUBST. No.1   = 4

-----------------------------------------
Press RETURN key to continue ...
-----------------------------------------
NUM. OF NODE &NODE NUM. OF ELE. No.1

-----------------------------------------

NUMBER OF ELEMENT NODE          = 4
NODE NUMBER FOR ELEMENT         = 1
NODE NUMBER FOR ELEMENT         = 2
NODE NUMBER FOR ELEMENT         = 6
NODE NUMBER FOR ELEMENT         = 5
TYPE OF ELEMENT                 = 1
THICKNESS OF ELEMENT            = 0.5
Do you accept these data ? (Y/N)    Y
-----------------------------------------
NUM. OF NODE &NODE NUM. OF ELE. No.2

-----------------------------------------

NUMBER OF ELEMENT NODE          = 4
NODE NUMBER FOR ELEMENT         = 5
NODE NUMBER FOR ELEMENT         = 6
NODE NUMBER FOR ELEMENT         = 10
NODE NUMBER FOR ELEMENT         = 9
TYPE OF ELEMENT                 = 1
THICKNESS OF ELEMENT            = 0.5
Do you accept these data ? (Y/N)    Y
-----------------------------------------
NUM. OF NODE &NODE NUM. OF ELE. No.3

-----------------------------------------

NUMBER OF ELEMENT NODE          = 4
NODE NUMBER FOR ELEMENT         = 9
NODE NUMBER FOR ELEMENT         = 10
NODE NUMBER FOR ELEMENT         = 14
NODE NUMBER FOR ELEMENT         = 13
TYPE OF ELEMENT                 = 1
THICKNESS OF ELEMENT            = 0.5
Do you accept these data ? (Y/N)    Y
```

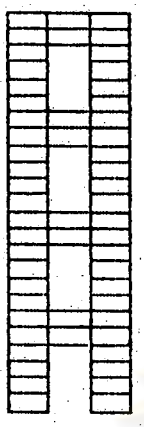
```
-----------------------------------------
NUM. OF NODE &NODE NUM. OF ELE. No.4

-----------------------------------------

NUMBER OF ELEMENT NODE          = 4
NODE NUMBER FOR ELEMENT         = 13
NODE NUMBER FOR ELEMENT         = 14
NODE NUMBER FOR ELEMENT         = 18
NODE NUMBER FOR ELEMENT         = 17
TYPE OF ELEMENT                 = 1
THICKNESS OF ELEMENT            = 0.5
Do you accept these data ? (Y/N)    Y
-----------------------------------------
NUM. OF NODE &NODE NUM. OF ELE. No.5

-----------------------------------------

NUMBER OF ELEMENT NODE          = 4
NODE NUMBER FOR ELEMENT         = 17
NODE NUMBER FOR ELEMENT         = 18
NODE NUMBER FOR ELEMENT         = 22
NODE NUMBER FOR ELEMENT         = 21
TYPE OF ELEMENT                 = 1
THICKNESS OF ELEMENT            = 0.5
Do you accept these data ? (Y/N)    Y
-----------------------------------------
NUM. OF NODE &NODE NUM. OF ELE. No.6

-----------------------------------------

NUMBER OF ELEMENT NODE          = 4
NODE NUMBER FOR ELEMENT         = 21
NODE NUMBER FOR ELEMENT         = 22
NODE NUMBER FOR ELEMENT         = 26
NODE NUMBER FOR ELEMENT         = 25
TYPE OF ELEMENT                 = 1
THICKNESS OF ELEMENT            = 0.5
Do you accept these data ? (Y/N)    Y
-----------------------------------------
NUM. OF NODE &NODE NUM. OF ELE. No.7

-----------------------------------------

NUMBER OF ELEMENT NODE          = 4
NODE NUMBER FOR ELEMENT         = 22
NODE NUMBER FOR ELEMENT         = 23
NODE NUMBER FOR ELEMENT         = 27
NODE NUMBER FOR ELEMENT         = 26
TYPE OF ELEMENT                 = 1
THICKNESS OF ELEMENT            = 0.5
Do you accept these data ? (Y/N)    Y
```

```
-----------------------------------------
NUM. OF NODE &NODE NUM. OF ELE. No.8

-----------------------------------------

NUMBER OF ELEMENT NODE          = 4
NODE NUMBER FOR ELEMENT         = 18
NODE NUMBER FOR ELEMENT         = 19
NODE NUMBER FOR ELEMENT         = 23
NODE NUMBER FOR ELEMENT         = 22
TYPE OF ELEMENT                 = 1
THICKNESS OF ELEMENT            = 0.5
Do you accept these data ? (Y/N)    Y
-----------------------------------------
NUM. OF NODE &NODE NUM. OF ELE. No.9

-----------------------------------------

NUMBER OF ELEMENT NODE          = 4
NODE NUMBER FOR ELEMENT         = 23
NODE NUMBER FOR ELEMENT         = 24
NODE NUMBER FOR ELEMENT         = 28
NODE NUMBER FOR ELEMENT         = 27
TYPE OF ELEMENT                 = 1
THICKNESS OF ELEMENT            = 0.5
Do you accept these data ? (Y/N)    Y
-----------------------------------------
NUM. OF NODE &NODE NUM. OF ELE. No.10

-----------------------------------------

NUMBER OF ELEMENT NODE          = 4
NODE NUMBER FOR ELEMENT         = 19
NODE NUMBER FOR ELEMENT         = 20
NODE NUMBER FOR ELEMENT         = 24
NODE NUMBER FOR ELEMENT         = 23
TYPE OF ELEMENT                 = 1
THICKNESS OF ELEMENT            = 0.5
Do you accept these data ? (Y/N)    Y
23
-----------------------------------------
NUM. OF NODE &NODE NUM. OF ELE. No.11

-----------------------------------------

NUMBER OF ELEMENT NODE          = 4
NODE NUMBER FOR ELEMENT         = 15
NODE NUMBER FOR ELEMENT         = 16
NODE NUMBER FOR ELEMENT         = 20
NODE NUMBER FOR ELEMENT         = 19
TYPE OF ELEMENT                 = 1
THICKNESS OF ELEMENT            = 0.5
Do you accept these data ? (Y/N)    Y
```

```
-----------------------------------------
NUM. OF NODE &NODE NUM. OF ELE. No.12

-----------------------------------------

NUMBER OF ELEMENT NODE          = 4
NODE NUMBER FOR ELEMENT         = 11
NODE NUMBER FOR ELEMENT         = 12
NODE NUMBER FOR ELEMENT         = 16
NODE NUMBER FOR ELEMENT         = 15
TYPE OF ELEMENT                 = 1
THICKNESS OF ELEMENT            = 0.5
Do you accept these data ? (Y/N)    Y
-----------------------------------------
NUM. OF NODE &NODE NUM. OF ELE. No.13

-----------------------------------------

NUMBER OF ELEMENT NODE          = 4
NODE NUMBER FOR ELEMENT         = 7
NODE NUMBER FOR ELEMENT         = 8
NODE NUMBER FOR ELEMENT         = 12
NODE NUMBER FOR ELEMENT         = 11
TYPE OF ELEMENT                 = 1
THICKNESS OF ELEMENT            = 0.5
Do you accept these data ? (Y/N)    Y
-----------------------------------------
NUM. OF NODE &NODE NUM. OF ELE. No.14

-----------------------------------------

NUMBER OF ELEMENT NODE          = 4
NODE NUMBER FOR ELEMENT         = 3
NODE NUMBER FOR ELEMENT         = 4
NODE NUMBER FOR ELEMENT         = 8
NODE NUMBER FOR ELEMENT         = 7
TYPE OF ELEMENT                 = 1
THICKNESS OF ELEMENT            = 0.5
Do you accept these data ? (Y/N)    Y
```

ข.4 ตัวอย่างข้อมูลและผลลัพธ์ (ตัวอย่างที่ 1)

1. ชิ้นส่วนย่อยชนิด Q4



```
-----------------

   PRINT  DATA

-----------------

LARGEST NODE NUMBER                 = 100
NUMBER OF FINITE ELEMENTS           = 56
NUMBER OF FINITE ELEMENTS ON DISK=  14
NUM. OF NODES/ELEMENT IN LEVEL 1  = 4
MAX. NUM. OF NODES IN SUBST.        = 8
MAX. NUM. OF ELEMENT IN ANY LEVEL=  14
MAX. NUM. OF BOUND. CON. NODES      = 4
NUM. OF SUBSTRUCTURES IN LEVEL 1  = 4
NUM. OF SUBSTRUCTURE LEVELS         = 2
NUMBER OF MATERIAL TYPES            = 1
ELEMENT TYPE  (1 OR 2).....         = 2
MAX. NUM. OF THE SAME SUBST.        = 4
MAX. NUM. OF NODAL POINT LOADS      = 4
REFERENCE TEMPERATURE               = 0
THE ELEMENTS ARE THE SAME TYPE      = Y
```

```
-----------------------------------------
NUM. OF SUBST. IN LEV. 1          = 4

-----------------------------------------
NUM. OF ELEMENTS IN SUBST.        = 14
NUM. OF ELEMENTS IN SUBST.        = -24
NUM. OF ELEMENTS IN SUBST.        = -24
NUM. OF ELEMENTS IN SUBST.        = -24

-----------------------------------------

NUM. OF NODES OF SUBSTRUCTURE No.1 = 8

NUM. OF BOUN. CON. NODES IN  LEV.1 = 0

-----------------------------------------
SUBST. NODE NUMBER OF NODE    1  = 1
SUBST. NODE NUMBER OF NODE    2  = 2
SUBST. NODE NUMBER OF NODE    3  = 3
SUBST. NODE NUMBER OF NODE    4  = 4
SUBST. NODE NUMBER OF NODE    5  = 25
SUBST. NODE NUMBER OF NODE    6  = 26
SUBST. NODE NUMBER OF NODE    7  = 27
SUBST. NODE NUMBER OF NODE    8  = 28

-----------------------------------------
NUM. OF SUBST. IN LEV. 2          = 1

-----------------------------------------
NUM. OF ELEMENTS IN SUBST.        = 4

-----------------------------------------

NUM. OF NODES OF SUBSTRUCTURE No.1 = 0

NUM. OF BOUN. CON. NODES IN  LEV.2 = 4

-----------------------------------------
    BOUNDARY CONSTRAIN AT JOINT

-----------------------------------------
  NODE     X     Y     X-Displ.   Y-Displ.
  ----    ---   ---    --------   --------
    1      1     1        0          0
    2      1     1        0          0
    3      1     1        0          0
    4      1     1        0          0
```

MATERIAL TYPE 1

YOUNG'S MODULUS	= 2812000
POISSON'S RATIO	= .4
MASS DENSITY	= 2.4
COEF. OF TEMPERATURE	= 1E-04

GLOBAL NODAL COORDINATE & TEMPERATURE

NODE	X-Coor.	Y-Coor.	TEMP.
1	0	0	0
2	2	0	0
3	4	0	0
4	6	0	0
5	0	1	0
6	2	1	0
7	4	1	0
8	6	1	0
9	0	2	0
10	2	2	0
11	4	2	0
12	6	2	0
13	0	3	0
14	2	3	0
15	4	3	0
16	6	3	0
17	0	4	0
18	2	4	0
19	4	4	0
20	6	4	0
21	0	5	0
22	2	5	0
23	4	5	0
24	6	5	0
25	0	6	0
26	2	6	0
27	4	6	0
28	6	6	0
29	0	7	0
30	2	7	0
31	4	7	0
32	6	7	0
33	0	8	0
34	2	8	0
35	4	8	0

36	6	8	0
37	0	9	0
38	2	9	0
39	4	9	0
40	6	9	0
41	0	10	0
42	2	10	0
43	4	10	0
44	6	10	0
45	0	11	0
46	2	11	0
47	4	11	0
48	6	11	0
49	0	12	0
50	2	12	0
51	4	12	0
52	6	12	0
53	0	13	0
54	2	13	0
55	4	13	0
56	6	13	0
57	0	14	0
58	2	14	0
59	4	14	0
60	6	14	0
61	0	15	0
62	2	15	0
63	4	15	0
64	6	15	0
65	0	16	0
66	2	16	0
67	4	16	0
68	6	16	0
69	0	17	0
70	2	17	0
71	4	17	0
72	6	17	0
73	0	18	0
74	2	18	0
75	4	18	0
76	6	18	0
77	0	19	0
78	2	19	0
79	4	19	0
80	6	19	0
81	0	20	0
82	2	20	0
83	4	20	0
84	6	20	0
85	0	21	0

86	2	21	0
87	4	21	0
88	6	21	0
89	0	22	0
90	2	22	0
91	4	22	0
92	6	22	0
93	0	23	0
94	2	23	0
95	4	23	0
96	6	23	0
97	0	24	0
98	2	24	0
99	4	24	0
100	6	24	0

ELEMENT NODES & TYPE

ELE. No.	I	J	K	L	MAT.	THC.
1	1	2	6	5	1	.5
2	5	6	10	9	1	.5
3	9	10	14	13	1	.5
4	13	14	18	17	1	.5
5	17	18	22	21	1	.5
6	21	22	26	25	1	.5
7	22	23	27	26	1	.5
8	18	19	23	22	1	.5
9	23	24	28	27	1	.5
10	19	20	24	23	1	.5
11	15	16	20	19	1	.5
12	11	12	16	15	1	.5
13	7	8	12	11	1	.5
14	3	4	8	7	1	.5
15	25	26	30	29	1	.5
16	29	30	34	33	1	.5
17	33	34	38	37	1	.5
18	37	38	42	41	1	.5
19	41	42	46	45	1	.5
20	45	46	50	49	1	.5
21	46	47	51	50	1	.5
22	42	43	47	46	1	.5
23	47	48	52	51	1	.5
24	43	44	48	47	1	.5
25	39	40	44	43	1	.5
26	35	36	40	39	1	.5
27	31	32	36	35	1	.5

28	27	28	32	31	1	.5
29	49	50	54	53	1	.5
30	53	54	58	57	1	.5
31	57	58	62	61	1	.5
32	61	62	66	65	1	.5
33	65	66	70	69	1	.5
34	69	70	74	73	1	.5
35	70	71	75	74	1	.5
36	66	67	71	70	1	.5
37	71	72	76	75	1	.5
38	67	68	72	71	1	.5
39	63	64	68	67	1	.5
40	59	60	64	63	1	.5
41	55	56	60	59	1	.5
42	51	52	56	55	1	.5
43	73	74	78	77	1	.5
44	77	78	82	81	1	.5
45	81	82	86	85	1	.5
46	85	86	90	89	1	.5
47	89	90	94	93	1	.5
48	93	94	98	97	1	.5
49	94	95	99	98	1	.5
50	90	91	95	94	1	.5
51	95	96	100	99	1	.5
52	91	92	96	95	1	.5
53	87	88	92	91	1	.5
54	83	84	88	87	1	.5
55	79	80	84	83	1	.5
56	75	76	80	79	1	.5

NODAL POINT LOADS IN GLOBAL Coor.

NODE	FX	FY
21	20	0
45	40	0
69	60	0
93	60	0

NUM. OF THE SAME SUBST. No.1 = 4

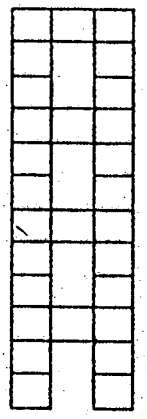
NODE No.	X-Displ.	Y-Displ.
1	0	0
2	0	0
3	0	0
4	0	0
5	2.8E-04	4E-04
6	1.3E-04	-8E-05
7	1.2E-04	3E-05
8	3.1E-04	-4.5E-04
9	6.9E-04	7.5E-04
10	5.5E-04	-6E-05
11	5.4E-04	-3E-05
12	7.2E-04	-8.4E-04
13	1.23E-03	1.03E-03
14	1.09E-03	2E-05
15	1.08E-03	-1.6E-04
16	1.26E-03	-1.15E-03
17	1.79E-03	1.22E-03
18	1.73E-03	1.8E-04
19	1.73E-03	-3.6E-04
20	1.81E-03	-1.38E-03
21	2.36E-03	1.43E-03
22	2.24E-03	2.6E-04
23	2.24E-03	-4.6E-04
24	2.36E-03	-1.63E-03
25	3.01E-03	1.69E-03
26	2.83E-03	2.7E-04
27	2.83E-03	-4.8E-04
28	3.03E-03	-1.93E-03
29	3.83E-03	1.97E-03
30	3.75E-03	2.3E-04
31	3.75E-03	-4.8E-04
32	3.87E-03	-2.24E-03
33	4.84E-03	2.17E-03
34	4.75E-03	2.4E-04
35	4.75E-03	-5.4E-04
36	4.87E-03	-2.47E-03
37	5.9E-03	2.31E-03
38	5.8E-03	3.3E-04
39	5.8E-03	-6.7E-04
40	5.92E-03	-2.63E-03
41	6.9E-03	2.37E-03
42	6.9E-03	4.8E-04
43	6.89E-03	-8.5E-04
44	6.91E-03	-2.71E-03
45	7.89E-03	2.46E-03
46	7.79E-03	5.5E-04
47	7.78E-03	-9.3E-04
48	7.85E-03	-2.82E-03
49	8.84E-03	2.6E-03
50	8.72E-03	5.6E-04

NODE No.	X-Displ.	Y-Displ.
51	8.71E-03	-9.4E-04
52	8.84E-03	-2.99E-03
53	9.93E-03	2.75E-03
54	9.89E-03	5.2E-04
55	9.88E-03	-9.2E-04
56	9.93E-03	-3.17E-03
57	.01113	2.85E-03
58	.0111	5.1E-04
59	.01108	-9.5E-04
60	.01114	-3.28E-03
61	.01236	2.9E-03
62	.01232	5.6E-04
63	.0123	-1.02E-03
64	.01236	-3.34E-03
65	.01354	2.89E-03
66	.01355	6.6E-04
67	.01353	-1.14E-03
68	.01352	-3.35E-03
69	.01469	2.9E-03
70	.01461	7.2E-04
71	.01459	-1.19E-03
72	.01462	-3.37E-03
73	.01575	2.95E-03
74	.01569	7.3E-04
75	.01566	-1.2E-03
76	.01572	-3.43E-03
77	.01688	3E-03
78	.01687	7E-04
79	.01685	-1.19E-03
80	.01686	-3.49E-03
81	.01807	3.03E-03
82	.01806	7E-04
83	.01804	-1.19E-03
84	.01805	-3.53E-03
85	.01926	3.03E-03
86	.01925	7.2E-04
87	.01923	-1.23E-03
88	.01924	-3.54E-03
89	.02043	3.01E-03
90	.02044	7.6E-04
91	.02041	-1.28E-03
92	.02039	-3.51E-03
93	.02159	2.99E-03
94	.02152	7.9E-04
95	.0215	-1.3E-03
96	.02151	-3.5E-03
97	.02264	2.99E-03
98	.02261	8.1E-04
99	.02259	-1.31E-03
100	.0226	-3.49E-03

EL.No.	X	Y	X-STRESS	Y-STRESS	XY-STRESS	MAX.-STRESS	MIN.-STRESS	ANGLE
1	1	.5	96.72682	501.83618	87.83711	520.06138	78.50162	78.2781
2	1	1.5	-3.21538	504.23618	87.83712	519.01018	-17.98938	80.45235
3	1	2.5	-1.87322	506.63617	87.83712	521.38112	-16.61817	80.47078
4	1	3.5	58.47532	509.03615	87.83712	525.55446	41.957	79.34956
5	1	4.5	40.93739	421.56028	-8.47574	421.74892	40.74875	-88.72498
6	1	5.5	-68.26544	362.96132	-33.25968	365.5115	-70.81561	-85.61545
7	3	5.5	-11.07917	-23.58383	221.48308	204.2398	-238.90281	44.1915
8	3	4.5	-13.71068	-25.934	200.18011	180.45104	-220.09572	44.12563
9	5	5.5	51.2267	-443.77812	-28.22357	52.83072	-445.38213	-3.25277
10	5	4.5	-58.60926	-507.2269	-11.70453	-58.30409	-507.53207	-1.4935
11	5	3.5	-64.44485	-626.6368	92.16273	-49.72177	-641.35988	9.07636
12	5	2.5	2.04593	-629.03682	92.16273	15.22986	-642.22075	8.14096001
13	5	1.5	4.08978	-631.43682	92.16273	17.18519	-644.53222	8.08702
14	5	.5	-122.69447	-633.83683	92.16273	-106.58459	-649.9467	9.91502
15	1	6.5	-63.71991	307.80453	80.23204	324.39048	-80.30586	78.32007
16	1	7.5	2.22759	310.20449	80.23206	329.85252	-17.42044	76.23965
17	1	8.5	-1.99425	312.60448	80.23204	331.88448	-21.27424	76.48786
18	1	9.5	56.60241	315.00443	80.23205	337.8892	33.71765	74.08014
19	1	10.5	18.94424	229.94694	-11.591	230.58176	18.30942	-86.86515
20	1	11.5	-80.14539	179.66713	-39.82835	185.63557	-86.11383	-81.47741
21	3	11.5	-21.03017	-14.01363	187.43066	169.9416	-204.98539	45.53616
22	3	10.5	-18.39158	-15.62055	193.00623	176.00514	-210.01727	45.20565
23	5	11.5	45.3493	-236.45403	-27.60249	48.0275	-239.13222	-5.54191
24	5	10.5	-52.02257	-292.32697	-21.4154	-50.129	-294.22054	-5.05301
25	5	9.5	-62.05261	-399.00502	79.76784	-44.12296	-416.93466	12.66799
26	5	8.5	2.17407	-401.40507	79.76783	17.36822	-416.59923	10.78449
27	5	7.5	-2.08283	-403.80508	79.76781	13.1766	-419.06451	10.82974
28	5	6.5	59.26935	-406.20513	79.76781	72.5396	-419.49538	9.45926
29	1	12.5	-55.06055	134.38591	59.94292	151.75926	-72.43391	73.83674
30	1	13.5	1.91157	136.78584	59.94296	159.57578	-20.87837	69.18351
31	1	14.5	-1.33427	139.18582	59.94291	161.28175	-23.4302	69.76529
32	1	15.5	37.44537	141.5858	59.94293	168.91622	10.11495	65.48982
33	1	16.5	-14.17566	82.32863	1.55741	82.33376	-14.20079	89.07567
34	1	17.5	-78.32204	50.73415	-35.74065	59.97101	-87.5589	-75.50946
35	3	17.5	-30.51495	-5.513	113.48754	96.16	-132.18795	48.14297
36	3	16.5	-29.22034	-7.3597	132.85097	115.00984	-151.58987	47.35171
37	5	17.5	25.45956	-82.42151	-17.74709	28.30407	-85.26601	-9.10592
38	5	16.5	-37.54919	-119.36938	-14.40853	-35.086	-121.83257	-9.70113
39	5	15.5	-45.08349	-191.98622	60.05703	-23.65625	-213.41346	19.63546
40	5	14.5	1.58592	-194.38631	60.05696	18.52638	-211.32676	15.75234
41	5	13.5	-1.70021	-196.78632	60.05699	15.30581	-213.79234	15.81025
42	5	12.5	48.57199	-199.18641	60.05694	62.36229	-212.97671	12.93211
43	1	18.5	-34.7246	28.01649	29.81057	39.92159	-46.6297	68.23029
44	1	19.5	1.19494	30.41647	29.81068	49.00437	-17.39296	58.05516
45	1	20.5	-.52532	32.81646	29.81055	50.30092	-18.00977	59.60757
46	1	21.5	14.30365	35.21641	29.81061	56.3513	-6.83124	54.66442
47	1	22.5	-34.99903	7.47616	10.53784	9.94682	-37.46969	76.80497
48	1	23.5	-68.81379	-2.67684	-24.24574	5.25932	-76.74995	-71.8756
49	3	23.5	-29.80406	2.51015	31.69221	21.92619	-49.2201	58.50654
50	3	22.5	-30.57455	-.50042	55.54046	42.00255	-73.07752	52.57456

EL.No.	X	Y	X-STRESS	Y-STRESS	XY-STRESS	MAX.-STRESS	MIN.-STRESS	ANGLE
51	5	23.5	8.86096	-3.43342	-7.44669	12.36992	-6.94238	-25.23028
52	5	22.5	-17.80194	-17.77592	-6.07837	-11.71055	-23.86732	-45.06132
53	5	21.5	-21.75412	-52.01657	30.18941	-3.11621	-70.65448	31.68976
54	5	20.5	.76898	-54.41667	30.18944	14.07562	-67.72331	23.78651
55	5	19.5	-.93115	-56.81668	30.18933	12.26237	-70.0102	23.60657
56	5	18.5	26.70118	-59.21675	30.18933	36.2481	-68.76368	17.54878

2. ชิ้นส่วนย่อยชนิด Q8



PRINT DATA

```
LARGEST NODE NUMBER                 = 135
NUMBER OF FINITE ELEMENTS           = 28
NUMBER OF FINITE ELEMENTS ON DISK=  7
NUM. OF NODES/ELEMENT IN LEVEL 1  = 8
MAX. NUM. OF NODES IN SUBST.        = 12
MAX. NUM. OF ELEMENT IN ANY LEVEL=  7
MAX. NUM. OF BOUND. CON. NODES      = 6
NUM. OF SUBSTRUCTURES IN LEVEL 1  = 4
NUM. OF SUBSTRUCTURE LEVELS         = 2
NUMBER OF MATERIAL TYPES            = 1
ELEMENT TYPE  (1 OR 2).....         = 2
MAX. NUM. OF THE SAME SUBST.        = 4
MAX. NUM. OF NODAL POINT LOADS      = 4
REFERENCE TEMPERATURE               = 0
THE ELEMENTS ARE THE SAME TYPE      = Y
```

NUM. OF SUBST. IN LEV. 1 = 4

NUM. OF ELEMENTS IN SUBST. = 7
NUM. OF ELEMENTS IN SUBST. = -32
NUM. OF ELEMENTS IN SUBST. = -32
NUM. OF ELEMENTS IN SUBST. = -32

NUM. OF NODES OF SUBSTRUCTURE No.1 = 12

NUM. OF BOUN. CON. NODES IN LEV.1 = 0

SUBST. NODE NUMBER OF NODE 1 = 1
SUBST. NODE NUMBER OF NODE 2 = 2
SUBST. NODE NUMBER OF NODE 3 = 3
SUBST. NODE NUMBER OF NODE 4 = 5
SUBST. NODE NUMBER OF NODE 5 = 6
SUBST. NODE NUMBER OF NODE 6 = 7
SUBST. NODE NUMBER OF NODE 7 = 33
SUBST. NODE NUMBER OF NODE 8 = 34
SUBST. NODE NUMBER OF NODE 9 = 35
SUBST. NODE NUMBER OF NODE 10 = 37
SUBST. NODE NUMBER OF NODE 11 = 38
SUBST. NODE NUMBER OF NODE 12 = 39

NUM. OF SUBST. IN LEV. 2 = 1

NUM. OF ELEMENTS IN SUBST. = 4

NUM. OF NODES OF SUBSTRUCTURE No.1 = 0

NUM. OF BOUN. CON. NODES IN LEV.2 = 6

BOUNDARY CONSTRAIN AT JOINT

NODE	X	Y	X-Displ.	Y-Displ.
1	1	1	0	0
2	1	1	0	0
3	1	1	0	0
5	1	1	0	0
6	1	1	0	0
7	1	1	0	0

MATERIAL TYPE 1

YOUNG'S MODULUS = 2812000
POISSON'S RATIO = .4
MASS DENSITY = 2.4
COEF. OF TEMPERATURE = 1E-04

GLOBAL NODAL COORDINATE & TEMPERATURE

NODE	X-Coor.	Y-Coor.	TEMP.
1	0	0	0
2	1	0	0
3	2	0	0
4	3	0	0
5	4	0	0
6	5	0	0
7	6	0	0
8	0	1	0
9	2	1	0
10	4	1	0
11	6	1	0
12	0	2	0
13	1	2	0
14	2	2	0
15	4	2	0
16	5	2	0
17	6	2	0
18	0	3	0
19	2	3	0
20	4	3	0
21	6	3	0
22	0	4	0
23	1	4	0
24	2	4	0
25	3	4	0
26	4	4	0
27	5	4	0
28	6	4	0
29	0	5	0
30	2	5	0

31	4	5	0
32	6	5	0
33	0	6	0
34	1	6	0
35	2	6	0
36	3	6	0
37	4	6	0
38	5	6	0
39	6	6	0
40	0	7	0
41	2	7	0
42	4	7	0
43	6	7	0
44	0	8	0
45	1	8	0
46	2	8	0
47	4	8	0
48	5	8	0
49	6	8	0
50	0	9	0
51	2	9	0
52	4	9	0
53	6	9	0
54	0	10	0
55	1	10	0
56	2	10	0
57	3	10	0
58	4	10	0
59	5	10	0
60	6	10	0
61	0	11	0
62	2	11	0
63	4	11	0
64	6	11	0
65	0	12	0
66	1	12	0
67	2	12	0
68	3	12	0
69	4	12	0
70	5	12	0
71	6	12	0
72	0	13	0
73	2	13	0
74	4	13	0
75	6	13	0
76	0	14	0
77	1	14	0
78	2	14	0
79	4	14	0
80	5	14	0

81	6	14	0
82	0	15	0
83	2	15	0
84	4	15	0
85	6	15	0
86	0	16	0
87	1	16	0
88	2	16	0
89	3	16	0
90	4	16	0
91	5	16	0
92	6	16	0
93	0	17	0
94	2	17	0
95	4	17	0
96	6	17	0
97	0	18	0
98	1	18	0
99	2	18	0
100	3	18	0
101	4	18	0
102	5	18	0
103	6	18	0
104	0	19	0
105	2	19	0
106	4	19	0
107	6	19	0
108	0	20	0
109	1	20	0
110	2	20	0
111	4	20	0
112	5	20	0
113	6	20	0
114	0	21	0
115	2	21	0
116	4	21	0
117	6	21	0
118	0	22	0
119	1	22	0
120	2	22	0
121	3	22	0
122	4	22	0
123	5	22	0
124	6	22	0
125	0	23	0
126	2	23	0
127	4	23	0
128	6	23	0
129	0	24	0
130	1	24	0

131	2	24	0
132	3	24	0
133	4	24	0
134	5	24	0
135	6	24	0

ELEMENT NODES & TYPE

ELE. No.	I	J	K	L	MAT.	THC.
1	1	3	14	12	1	.5
	2	9	13	8		
2	12	14	24	22	1	.5
	13	19	23	18		
3	22	24	35	33	1	.5
	23	30	34	29		
4	24	26	37	35	1	.5
	25	31	36	30		
5	26	28	39	37	1	.5
	27	32	38	31		
6	15	17	28	26	1	.5
	16	21	27	20		
7	5	7	17	15	1	.5
	6	11	16	10		
8	33	35	46	44	1	.5
	34	41	45	40		
9	44	46	56	54	1	.5
	45	51	55	50		
10	54	56	67	65	1	.5
	55	62	66	61		
11	56	58	69	67	1	.5
	57	63	68	62		
12	58	60	71	69	1	.5
	59	64	70	63		
13	47	49	60	58	1	.5
	48	53	59	52		
14	37	39	49	47	1	.5
	38	43	48	42		
15	65	67	78	76	1	.5
	66	73	77	72		
16	76	78	88	86	1	.5
	77	83	87	82		
17	86	88	99	97	1	.5
	87	94	98	93		
18	88	90	101	99	1	.5
	89	95	100	94		

19	90	92	103	101	1	.5
	91	96	102	95		
20	79	81	92	90	1	.5
	80	85	91	84		
21	69	71	81	79	1	.5
	70	75	80	74		
22	97	99	110	108	1	.5
	98	105	109	104		
23	108	110	120	118	1	.5
	109	115	119	114		
24	118	120	131	129	1	.5
	119	126	130	125		
25	120	122	133	131	1	.5
	121	127	132	126		
26	122	124	135	133	1	.5
	123	128	134	127		
27	111	113	124	122	1	.5
	112	117	123	116		
28	101	103	113	111	1	.5
	102	107	112	106		

NODAL POINT LOADS IN GLOBAL Coor.

NODE	FX	FY
29	20	0
61	40	0
93	60	0
125	60	0

NUM. OF THE SAME SUBST. No.1 = 4

NODE No.	X-Displ.	Y-Displ.
1	0	0
2	0	0
3	0	0
4	0	0
5	0	0
6	0	0
7	0	0
8	3.1E-04	4.2E-04
9	1.5E-04	-8E-05
10	1.4E-04	4E-05
11	3.4E-04	-4.7E-04
12	7.4E-04	7.9E-04
13	6.4E-04	3.6E-04
14	6.3E-04	-9E-05
15	6.2E-04	0
16	6.5E-04	-4.5E-04
17	7.6E-04	-8.8E-04
18	1.34E-03	1.1E-03
19	1.2E-03	-1E-05
20	1.19E-03	-1.3E-04
21	1.37E-03	-1.23E-03
22	1.94E-03	1.31E-03
23	1.9E-03	7E-04
24	1.89E-03	1.7E-04
25	1.97E-03	-1.1E-04
26	1.89E-03	-3.5E-04
27	1.91E-03	-8.7E-04
28	1.96E-03	-1.48E-03
29	2.57E-03	1.52E-03
30	2.45E-03	2.3E-04
31	2.45E-03	-4.4E-04
32	2.58E-03	-1.73E-03
33	3.29E-03	1.79E-03
34	3.16E-03	9.9E-04
35	3.08E-03	2.4E-04
36	2.99E-03	-1E-04
37	3.08E-03	-4.8E-04
38	3.17E-03	-1.23E-03
39	3.31E-03	-2.04E-03
40	4.18E-03	2.07E-03
41	4.08E-03	1.7E-04
42	4.08E-03	-4.4E-04
43	4.21E-03	-2.35E-03
44	5.26E-03	2.26E-03
45	5.2E-03	1.24E-03
46	5.17E-03	1.7E-04
47	5.17E-03	-4.8E-04
48	5.22E-03	-1.55E-03
49	5.28E-03	-2.57E-03
50	6.39E-03	2.42E-03

NODE No.	X-Displ.	Y-Displ.
51	6.32E-03	2.9E-04
52	6.31E-03	-6.3E-04
53	6.41E-03	-2.75E-03
54	7.47E-03	2.5E-03
55	7.48E-03	1.43E-03
56	7.48E-03	4.7E-04
57	7.55E-03	-2E-04
58	7.47E-03	-8.5E-04
59	7.47E-03	-1.8E-03
60	7.48E-03	-2.85E-03
61	8.53E-03	2.57E-03
62	8.43E-03	5.3E-04
63	8.42E-03	-9.3E-04
64	8.5E-03	-2.95E-03
65	9.56E-03	2.7E-03
66	9.47E-03	1.61E-03
67	9.41E-03	5.4E-04
68	9.32E-03	-1.9E-04
69	9.4E-03	-9.5E-04
70	9.46E-03	-2.01E-03
71	9.55E-03	-3.11E-03
72	.0107	2.86E-03
73	.01066	4.8E-04
74	.01064	-9.1E-04
75	.0107	-3.29E-03
76	.01198	2.94E-03
77	.01195	1.73E-03
78	.01194	4.6E-04
79	.01192	-9.2E-04
80	.01194	-2.18E-03
81	.01198	-3.39E-03
82	.01327	3.01E-03
83	.01324	5.4E-04
84	.01322	-1.01E-03
85	.01327	-3.47E-03
86	.01451	3.02E-03
87	.01453	1.8E-03
88	.01454	6.6E-04
89	.01457	-2.5E-04
90	.01451	-1.16E-03
91	.0145	-2.29E-03
92	.01449	-3.49E-03
93	.01572	3.01E-03
94	.01564	7.1E-04
95	.01562	-1.21E-03
96	.01566	-3.49E-03
97	.01683	3.04E-03
98	.0168	1.88E-03
99	.01676	7.3E-04
100	.01669	-2.5E-04

NODE No.	X-Displ.	Y-Displ.
101	.01674	-1.22E-03
102	.01677	-2.37E-03
103	.0168	-3.54E-03
104	.01801	3.09E-03
105	.018	6.9E-04
106	.01797	-1.2E-03
107	.01799	-3.6E-03
108	.01925	3.12E-03
109	.01925	1.91E-03
110	.01924	6.8E-04
111	.01922	-1.2E-03
112	.01922	-2.43E-03
113	.01923	-3.63E-03
114	.02049	3.13E-03
115	.02049	7.1E-04
116	.02046	-1.24E-03
117	.02047	-3.65E-03
118	.0217	3.11E-03
119	.02171	1.92E-03
120	.02172	7.8E-04
121	.02172	-2.7E-04
122	.02169	-1.31E-03
123	.02168	-2.45E-03
124	.02167	-3.63E-03
125	.0229	3.08E-03
126	.02283	8E-04
127	.02282	-1.33E-03
128	.02282	-3.6E-03
129	.02398	3.05E-03
130	.02398	1.95E-03
131	.02396	8.2E-04
132	.02392	-2.6E-04
133	.02393	-1.34E-03
134	.02394	-2.46E-03
135	.02395	-3.59E-03

G. P.	X	Y	X-STRESS	Y-STRESS	XY-STRESS	MAX.-STRESS	MIN.-STRESS	ANGLE
ELEMENT No. 1								
1	.423	.423	239.51587	991.81931	141.65983	1017.6099	213.72528	79.68174
2	.423	1.577	3.75254	817.48329	80.16353	825.3053	-4.06946	84.42697
3	1.577	.423	-89.76348	28.31548	35.44746	38.13951	-99.58751	74.50964
4	1.577	1.577	-30.51233	208.19406	96.94376	242.60457	-64.92284	70.45751
ELEMENT No. 2								
1	.423	2.423	-32.19517	689.8605	100.10661	703.48241	-45.81708	82.25113
2	.423	3.577	.60404	515.52449	23.64359	516.60785	-.47933	87.37651
3	1.577	2.423	56.03495	339.87431	77.00069	359.41763	36.49162	75.75864
4	1.577	3.577	128.48224	519.75289	153.4637	572.76248	75.47266	70.94401
ELEMENT No. 3								
1	.423	4.423	53.37811	526.51827	-31.558	528.61388	51.2825	-86.20085
2	.423	5.577	-69.69301	605.53253	-45.99502	608.65122	-72.8117	-86.12101
3	1.577	4.423	72.90637	315.09049	17.05077	316.28505	71.71182	85.99249
4	1.577	5.577	-105.18521	152.31956	-26.61219	155.04107	-107.90672	-84.16091
ELEMENT No. 4								
1	2.423	4.423	201.2702	208.80979	189.73243	394.80987	15.27012	45.56913
2	2.423	5.577	-223.79822	-59.58227	233.48428	105.81052	-389.19101	54.68746
3	3.577	4.423	-224.71775	-228.61058	209.74336	-16.91177	-436.41655	44.73416
4	3.577	5.577	202.18971	44.90375	219.01592	356.25403	-109.16058	35.12405
ELEMENT No. 5								
1	4.423	4.423	-85.85371	-405.08107	13.2399	-85.30553	-405.62925	2.3709
2	4.423	5.577	92.93377	-237.32099	-20.83159	94.24258	-238.62981	-3.59507
3	5.577	4.423	-72.5411	-641.0414	-40.88685	-69.61555	-643.96695	-4.09267
4	5.577	5.577	52.82979	-713.53944	-36.38077	54.55297	-715.26261	-2.71179
ELEMENT No. 6								
1	4.423	2.423	-58.57759	-468.10097	77.77835	-44.30316	-482.3754	10.39959
2	4.423	3.577	-140.33287	-648.22347	162.74552	-92.65868	-695.89766	16.32726
3	5.577	2.423	30.90414	-807.17706	105.11542	43.887	-820.15993	7.04098
4	5.577	3.577	-1.92804	-621.51199	20.14826	-1.27353	-622.1665	1.86058
ELEMENT No. 7								
1	4.423	.423	71.15502	-156.11966	25.6781	74.02008	-158.98472	6.36651
2	4.423	1.577	36.40732	-336.24215	103.6337	63.2887	-363.12354	14.54139
3	5.577	.423	-262.79537	-1128.75835	157.21567	-235.13629	-1156.41744	9.97798
4	5.577	1.577	-.67816	-943.0933	79.26007	5.94137	-949.71282	4.77406
ELEMENT No. 8								
1	.423	6.423	-31.85078	628.70569	25.16002	629.66262	-32.80772	87.82186
2	.423	7.577	28.39123	471.84168	100.0139	493.35472	6.87819	77.86061
3	1.577	6.423	-141.44377	-.95885	134.47529	80.51424	-222.91685	58.79004
4	1.577	7.577	-4.80444	161.44779	59.62144	180.61863	-23.97528	72.17518

G. P.	X	Y	X-STRESS	Y-STRESS	XY-STRESS	MAX.-STRESS	MIN.-STRESS	ANGLE
ELEMENT No. 9								
1	.423	8.423	-17.78325	357.00926	91.4301	378.12396	-38.89796	76.99619
2	.423	9.577	16.82326	200.14522	15.76366	201.49084	15.47763	85.12092
3	1.577	8.423	11.58201	280.33767	68.20515	296.65604	-4.73636	76.54468
4	1.577	9.577	140.71084	442.74428	143.87163	500.30631	83.14881	68.194
ELEMENT No. 10								
1	.423	10.423	12.29834	220.19582	-32.37768	225.12157	7.37259	-81.34969
2	.423	11.577	-90.74784	305.59847	-56.54078	313.50651	-98.65588	-82.03801
3	1.577	10.423	45.80081	237.92855	3.03732	237.97656	45.7528	89.09452
4	1.577	11.577	-121.86095	81.82342	-23.56637	84.51451	-124.55204	-83.48551
ELEMENT No. 11								
1	2.423	10.423	176.00149	190.85624	187.36	370.93603	-4.0783	46.13507
2	2.423	11.577	-215.60065	-61.29011	197.48498	73.57644	-350.4672	55.67003
3	3.577	10.423	-211.61467	-206.42517	203.80083	-5.20258	-412.83727	45.36472
4	3.577	11.577	172.01537	49.52	186.5867	307.14965	-85.61428	35.91371
ELEMENT No. 12								
1	4.423	10.423	-78.42786	-293.82551	1.12706	-78.42196	-293.83141	.29979
2	4.423	11.577	87.24392	-131.62816	-18.56587	88.8076	-133.19184	-4.81429
3	5.577	10.423	-51.99633	-305.84439	-48.30549	-43.11488	-314.72585	-10.41806
4	5.577	11.577	51.35003	-384.51044	-40.03843	54.99746	-388.15787	-5.20518
ELEMENT No. 13								
1	4.423	8.423	-8.70645	-378.04033	67.3395	3.18827	-389.93505	10.01729
2	4.423	9.577	-149.51135	-535.63473	145.85918	-100.60669	-584.5394	18.53563
3	5.577	8.423	18.74391	-437.64983	93.02615	36.9769	-455.88282	11.08932
4	5.577	9.577	-17.56547	-274.51288	14.50652	-16.74907	-275.32929	3.22111
ELEMENT No. 14								
1	4.423	6.423	132.17209	-105.0788	130.97386	190.25586	-163.16257	23.91614
2	4.423	7.577	5.88357	-262.6732	61.81883	19.43026	-276.21988	12.36015
3	5.577	6.423	29.72576	-720.2113	29.39177	30.87593	-721.36147	2.24097
4	5.577	7.577	-29.47127	-557.07436	98.54683	-11.6654	-574.88022	10.24195
ELEMENT No. 15								
1	.423	12.423	-20.1402	339.80533	19.11559	340.81763	-21.15252	86.96856
2	.423	13.577	23.05449	223.34213	77.08627	249.57504	-3.17842	71.20626
3	1.577	12.423	-119.28972	-66.07456	100.1188	10.91196	-198.27625	52.44145
4	1.577	13.577	.43388	55.9312	42.1483	78.64507	-22.27999	61.67961
ELEMENT No. 16								
1	.423	14.423	-14.38875	138.08524	66.09757	162.74903	-39.05254	69.53734
2	.423	15.577	14.25412	21.62206	10.7385	29.29093	6.58524	54.46757
3	1.577	14.423	12.17807	145.24565	53.13668	163.86028	-6.43655	70.69383
4	1.577	15.577	98.67472	267.25145	108.49594	320.35267	45.5735	63.92147

G. P.	X	Y	X-STRESS	Y-STRESS	XY-STRESS	MAX.-STRESS	MIN.-STRESS	ANGLE
ELEMENT No. 25								
1	2.423	22.423	14.80202	56.03082	49.02884	88.6027	-17.76986	56.40227
2	2.423	23.577	-75.40179	1.10445	36.41696	15.66701	-89.96435	68.20432
3	3.577	22.423	-73.41482	-68.02558	61.63156	-9.02977	-132.41064	46.25173
4	3.577	23.577	12.81431	-8.77269	29.3568	33.29893	-29.25732	34.90662
ELEMENT No. 26								
1	4.423	22.423	-26.0862	-58.92328	1.48501	-26.01918	-58.9903	2.58409
2	4.423	23.577	24.9569	-16.73745	-4.77557	25.49689	-17.27743	-6.45121
3	5.577	22.423	-17.4933	26.2799	-12.08585	29.39512	-20.60852	-75.54619
4	5.577	23.577	12.38104	8.21606	-12.69288	23.16113	-2.56403	-40.34132
ELEMENT No. 27								
1	4.423	20.423	-3.97036	-77.17659	25.91673	4.27588	-85.42283	17.65015
2	4.423	21.577	-51.80293	-135.00545	51.89828	-26.89031	-159.91808	25.6423
3	5.577	20.423	9.92222	-33.01177	34.68336	29.24453	-52.33407	29.12244
4	5.577	21.577	-6.11209	30.35969	8.70186	32.3295	-8.0819	77.24515
ELEMENT No. 28								
1	4.423	18.423	60.33041	22.9858	52.05169	96.95759	-13.64138	35.13288
2	4.423	19.577	.69963	-34.84289	23.06776	12.04777	-46.19103	26.19478
3	5.577	18.423	8.48125	-142.77425	8.54824	8.96282	-143.25582	3.2244
4	5.577	19.577	-11.54362	-79.40289	37.53231	5.12215	-96.06866	23.94303

ประวัติ

นายนิวัฒน์ นิลสิทธานุเคราะห์ เกิดเมื่อวันที่ 28 มีนาคม พ.ศ. 2501 ที่จังหวัดจันทบุรี สำเร็จการศึกษาชั้นมัธยมศึกษาตอนปลาย (มศ. 5) จากโรงเรียนเบญจมราชูทิศ จันทบุรี ในเดือนมีนาคม พ.ศ. 2519 สำเร็จการศึกษาวิศวกรรมศาสตร์บัณฑิต สาขาวิศวกรรมโยธา จากจุฬาลงกรณ์มหาวิทยาลัย ในเดือนมีนาคม พ.ศ. 2524