COPY 01

PROGRAM A

FAME

```
*****   MTS*MTS*MTS*MTS*MTS*MTS*MTS ... TS*MTS*MTS*MTS*MTS*MTS*   *****
*****                                                             *****
*****     2ND FEBRUARY                                            *****
*****                                                             *****
*****   THE NEXT COMPUTING SERVICE ... BRUARY AT 2 PM IN          *****
*****   DAYSH LECTURE ROOM A WHEN D ... IMILE - A PACKAGE         *****
*****   FOR SOLVING DIFFERENTIAL EQ ...                           *****
*****                                                             *****
*****   MTS SERVICE IS AVAILABLE AT ...                           *****
*****      MONDAY  TO  FRIDAY  09 ...                             *****
*****                                                             *****
*****   MTS*MTS*MTS*MTS*MTS*MTS*MTS*MTS*MTS*MTS*MTS*MTS*MTS*MTS*MTS*MTS*MTS*MTS*MTS*MTS*   *****
```

```
S8 NO. QMCG8        UNIVERSITY OF MICHIGAN TERMINAL SYSTEM AT NUMAC        12:22:35    MON  5-FEB-79
```

```
 QQQQQQQQQQ    MM          MM   CCCCCCCCCC     GGGGGGGGGG     8888888888
QQQQQQQQQQQQ   MMM        MMM  CCCCCCCCCCCC  GGGGGGGGGGGG   888888888888
QQ        QQ   MMMM      MMMM  CC        CC  GG        GG   88        88
QQ        QQ   MM MM    MM MM  CC            GG             88        88
QQ        QQ   MM  MMMM    MM  CC            GG              88      88
QQ        QQ   MM   MM     MM  CC            GG               88888888
QQ        QQ   MM          MM  CC            GG     GGGGG     88888888
QQ    QQ  QQ   MM          MM  CC            GG     GGGGG    88      88
QQ     QQ QQ   MM          MM  CC            GG        GG   88        88
QQ      QQQQ   MM          MM  CC        CC  GG        GG   88        88
QQQQQQQQQQQ    MM          MM  CCCCCCCCCCCC  GGGGGGGGGGGG   888888888888
 QQQQQQQQ QQ   MM          MM   CCCCCCCCCC     GGGGGGGGGG     8888888888
```

```
C FORCE AND MOMENT ENHANCEMENT(FAME)
C SIMPLX VERSION OF MODIFYING FOUR BAR LINKAGES
C USING SEGMENTAL COUNTERWEIGHTS WITH VARYING MASSES AND INERTIAS
C PROGRAM A
C NPTS=NO. OF ACCURACY POINTS I2
C N=NUMBER OF PARAMETES I2
C EPS1=CONVERGENCE ERROR
C FMIN=FINAL ACCEPTABLE MINIMUM ERROR
C NITER=NO. OF ITERATIONS I5
C NPRINT=FULL PARAMETER PRINT EVERY NPRINT ITER. I4
C NCHECK=NO. OF ITERS. PER CONVERGENCE CHECK I2
C IOUT1=
C IWRIT= 0  EVERY THING WRITTEN CUT
C        1  NOTHING WRITTEN CUT
C INTO= 0 NOTHING TAKEN IN
C      = 1 MIN. SET OF DATA TAKEN IN
C ISTOP= 0 PLOTTED OUTPUT
C      = 1 PLOTTED OUTPUT SUPPRESSED
C IER= 0 NO PRINTING AFTER EACH READ STATEMENT
C    = 1 EACH READ IS FOLLOWED BY PRINTING ** ERROR CHECK **
C NPRT=WRITES MAX. AND MIN. EVERY NPRT ITERS.
      DIMENSION X1(400),Y1(400),H1(400)
     1,X2(400),Y2(400),H2(400),H3(400),ITITLE(6)
      COMMON/RESTR/VLOW(10),VHIGH(10)
      COMMON /DEVICE/ IO1,IO2,IO3,IO4,IO5,IO6,IO7,IO8,IO9
      COMMON/CONV/ FMIN,NITER,NPRINT,NCHECK
      COMMON/OUT/ IOUT1,IOUT2
      COMMON/STR/XE2,YE2,ZE2
      COMMON/APEX/ V(1000),N,KOUT
      COMMON /SMPLX/ EXPAND,SHRINK,REDUCE,NPRT
      COMMON/FUNC/ F(30),MN,NIN
      COMMON/ZYXWVU/XYSCAL,XSCAL,YSCAL,XO,YO
      COMMON/KIN/TH2(500),TH3(500),TH4(500),C2(500),C3(500),C4(500)
     1,S2(500),S3(500),S4(500),W3(500),W4(500),W2,W22
     1,ALP3(500),ALP4(500),AL4,AL3,AL2,AL1,NPTS,JK
      COMMON/DYN/XA(36),YA(36),AMOM(36),AAM2,AAM3,AAM4,AAI2,AAI3,AAI4
     1,R22,R33,R44,G2,G3,G4,A1,A2,A3,A4,A5,A6,A7,A8,A9,ERRM,ERRF
     1,CA2,SA2,CA5,SA5,CA8,SA8
      COMMON/COUNTR/AM2,RR2,AM3,RR3,AI3,AM4,RR4,AI4,GG2,GG3,GG4
      COMMON/CONST/PIE,DEG,PI2,INTL
      COMMON/AAA/T2,T3,T4,RHO,GRAV
      COMMON/TITLE/KTITLE(6)
      DATA ITITLE/' CRA','NK  ','COUP','LER ',' ROC','KER '/
      DO 14 I=1,6
   14 KTITLE(I)=ITITLE(I)
      ICOMP=0
      INPLT=0
      KOUT=0
      NV=361
      IRD=0
      LSYMP=0
      ANGST=0.
      INTL=0
      DEGINC=1.
      XPAGE=6.0
      YPAGE=6.0
      NSUPER=1
      PIE=4.0*ATAN(1.0)
      DEG=180.0/PIE
      PI2=PIE*2.

      IO7=4
      IO9=6
      IO6=6
      READ(5,100)NSTP,N,NITER,NPRINT,NCHECK,IOUT1,IWRIT,INTO,ISTOP,IER,N
     1PRT,IRAND,NRD,EPS1,FMIN,REQERR,ERRM,ERRF,EXPAND,SHRINK,REDUCE
     1,LINTYP,LABUNB,LABB,GRAV,XPAGE,YPAGE
  100 FORMAT(2I2,I5,I4,I2,5I1,I3,I5,I2/8G10.5/I1,2I5,3G10.5)
      IF(IER.EQ.1)WRITE(IO6,106)
      READ(5,101)(VLOW(I),VHIGH(I),I=1,N)
      IF(IER.EQ.1)WRITE(IO6,106)
      READ(5,104)(TH2(I),XA(I),YA(I),AMOM(I),I=1,NSTP)
      IF(IER.EQ.1)WRITE(IO6,106)
      READ(5,103)W2,AL1,AL2,AL3,AL4
      IF(IER.EQ.1)WRITE(IO6,106)
      READ(5,102)AAM2,AAM3,AAM4,R22,R33,R44,G2,G3,G4
     1,AAI3,AAI4
      READ(5,103)T2,T3,T4,RHO
      IF(IER.EQ.1)WRITE(IO6,106)
  101 FORMAT(6G10.5)
  102 FORMAT(3G14.6)
  103 FORMAT(5G14.6)
  104 FORMAT(4G14.6)
  106 FORMAT(1H ,' ****** ERROR CHECK ****** ')
      WRITE(IO6,880)IRAND
      IF(IWRIT.EQ.1)GO TO 31
  880 FORMAT(1H ,//25X,' RANDOM NUMBER = ',I5)
      WRITE(IO6,879)NITER,REQERR,ERRM,ERRF,EXPAND,SHRINK,REDUCE
  879 FORMAT(1H ,'NO. ITERATIONS PER AREA =',I5,
     12X,'FOR REQUIRED % IMPROVEMENT =',G10.5//
     1' MOMENT ERROR FACTOR =',G10.5,5X,' FORCE ERROR FACTOR =',G10.5//
     1' SIMPLX CONTROLS:- EXPAND =',G10.5,2X,' SHRINK =',G10.5,2X,
     1' REDUCE =',G10.5//)
      WRITE(IO6,881)
  881 FORMAT(1H ,//15X,' LIMITS FOR RANDOM GENERATION OF '
     1'COUNTERWEIGHT PARAMETERS ')
      WRITE(IO6,882)
  882 FORMAT(1H ,/12X,'  COUNTERWEIGHT RADIUS ',8X,'   ANGULAR OFFSET '
     1,10X,'  SEGMENTAL ANGLE '
     1/16X,' LOW ',10X,' HIGH ',8X,' LOW ',8X,' HIGH '
     1,8X,' LOW ',8X' HIGH ')
      DO 13 I=1,3
      K=(I-1)*3+1
      KL=K+2
      L=(I-1)*2+1
      M=L+1
   13 WRITE(IO6,883)ITITLE(L),ITITLE(M),(VLOW(J),VHIGH(J),J=K,KL)
  883 FORMAT(1H ,4X,2A4,6(4X,G10.5))
      WRITE(IO6,884)(TH2(I),XA(I),YA(I),AMOM(I),I=1,NSTP)
  884 FORMAT(1H ,//15X,' DYNAMIC SYNTHESIS INPUT DATA '
     1//' INPUT ANGLE   X FORCE ',6X,' Y FORCE    INPUT MOMENT'
     1/4(4X,G10.5))
      WRITE(IO6,885)AL1,AL2,AL3,AL4,W2
  885 FORMAT(1H ,//30X,'LINKAGE KINEMATICS '//' AL1 = ',G10.5,
     1' AL2 = ',G10.5,' AL3 = ',G10.5,' AL4 = ',G10.5
     1//' ANGULAR VELOCITY OF INPUT CRANK = ',G10.5)
      WRITE(IO6,887)
      WRITE(IO6,886)ITITLE(1),ITITLE(2),AAM2,R22,G2
     1,ITITLE(3),ITITLE(4),AAM3,R33,G3,ITITLE(5)
     1,ITITLE(6),AAM4,R44,G4
      WRITE(IO6,888)AAI3,AAI4

  888 FORMAT(1H ,' INERTIA OF COUPLER ABOUT ITS CENTRE OF GRAVITY ='
     1,G14.6/' INERTIA OF ROCKER ABOUT ITS CENTRE OF GRAVITY =',G14.6)
  887 FORMAT(1H ,//'INITIAL DYNAMIC CHARACTERISTICS OF '
     1'LINKAGE PRIOR TO COUNTERWEIGHT ADDITION '//
     120X,' LINK MASS ',5X,' RADIAL OFFSET ',5X,' ANGULAR OFFSET')
  886 FORMAT(1H ,3(2A4,10X,G14.6,8X,G14.6,8X,G14.6/)//)
      WRITE(IO6,889)T2,T3,T4,RHO
  889 FORMAT(1H ,//30X,'COUNTERWEIGHT PARAMETERS'//
     1' THICKNESSES ARE T2 =',G10.5,2X,' T3 =',G10.5,2X,' T4 =',G10.5
     1,2X,' FOR A DENSITY ='G10.5//)
   31 W22=W2*W2
      AAI2=0.
      AAI3=AAI3+AAM3*R33*R33
      AAI4=AAI4+AAM4*R44*R44
      DO 11 I=1,NSTP
   11 TH2(I+NV)=TH2(I)/DEG
      NINT=N*(N+1)
      G2=G2/DEG
      G3=G3/DEG
      G4=G4/DEG
      DO 12 I=1,3
      J=I+1+(I-1)*2
      K=J+1
      VLOW(J)=VLOW(J)/DEG
      VHIGH(J)=VHIGH(J)/DEG
      VLOW(K)=VLOW(K)/DEG
   12 VHIGH(K)=VHIGH(K)/DEG
      NPTS=NSTP
      JK=NV
      CALL KINSOL
      DO 5 I=1,NV
    5 TH2(I)=(I-1)/DEG
      NPTS=NV
      JK=0
      CALL KINSOL
      ILM=NV+NSTP
      IO7=4
      N1=N+1
      KK=1
      INTL=1
      DO 24 I=1,N
   24 V(NINT+I)=1.0
      CALL RESULT(NPTS,NINT,FMAX,GMAX,HMAX,OMAX,
     1TATA1,XE2,YE2,ZE2)
      INTL=0
      IF(INTO.LT.1)GO TO 23
      READ(IO7,115)(V(I),I=1,N),F(1)
  115 FORMAT(1X,5G15.7)
      KK=2
      GO TO 23
    4 WRITE(IO6,105)IRAND
  105 FORMAT(1H ,//' NEW SEARCH AREA STARTED RANDOM NUMBER =',I10//)
   23 DO 290 J=KK,N1
      K=(J-1)*N
      DO 290 I=1,N
      IY=IRAND*65539
      IF(IY)270,280,280
  270 IY=IY+2147483647+1
  280 RAND=IY
      RAND=RAND*0.4656613E-9

      IRAND=IY
      V(I+K)=VLOW(I)+(VHIGH(I)-VLOW(I))*RAND
  290 CONTINUE
      KK=1
      INTO=0
      KGO=0
   17 IRD=IRD+1
      NPTS=NSTP
      JK=NV
      CALL SIMPLX(EPS1)
      NPTS=NV
      JK=0
      CALL RESULT(NPTS,MN,SMAX,RMAX,ZMAX,VMAX,
     1TETA,XE1,YE1,ZE1)
      TETA=TETA/DEG
      ERRX=(XE2-XE1)*100./XE2
      ERRY=(YE2-YE1)*100./YE2
      ERRZ=(ZE2-ZE1)*100./ZE2
      QQ=XE2+YE2+ZE2
      BH=ABS(FMAX)
      BI=ABS(GMAX)
      BJ=ABS(HMAX)
      BK=ABS(OMAX)
      XDIFF=(BH-ABS(SMAX))*100./BH
      YDIFF=(BI-ABS(RMAX))*100./BI
      ZDIFF=(BJ-ABS(ZMAX))*100./BJ
      PDIFF=(BK-ABS(VMAX))*100./BK
      DIFF=(QQ-XE1-YE1-ZE1)*100./QQ
      WRITE(IO6,120)XE2,XE1,YE2,YE1,ZE2,ZE1,ERRX,ERRY,ERRZ,DIFF
  120 FORMAT(1H ,//2X,'R.M.S. AREAS OF FRAME SHAKING FORCES AND MOMENT '
     1'DETERMINED W.R.T. CRANK ANGLE'//13X,'X FORCE',20X,'Y FORCE',22X
     1,'MOMENT'/4X,'UNBALANCED',6X,'BALANCED',4X,'UNBALANCED',6X,'BALANC
     1ED',4X,'UNBALANCED',6X,'BALANCED'/3X,6G14.6/'%DIFFS= ',3(G14.6,14X
     1)/' TOTAL PERCENTAGE R.M.S. IMPROVEMENT =',G14.6)
      WRITE(IO6,121)FMAX,SMAX,GMAX,RMAX,HMAX,ZMAX,XDIFF,YDIFF,ZDIFF,OMAX
     1,TATA1,VMAX,TETA,PDIFF
  121 FORMAT(1H ,//2X,' PEAK VALUES OF FRAME SHAKING FORCES AND MOMENT '
     1'DETERMINED W.R.T. CRANK ANGLE'//13X,'X FORCE',20X,'Y FORCE',22X
     1,'MOMENT'/4X,'UNBALANCED',6X,'BALANCED',4X,'UNBALANCED',6X,'BALANC
     1ED',4X,'UNBALANCED',6X,'BALANCED'/3X,6G14.6/'%DIFFS= ',3(G14.6,14X
     1)/' OLD MAX. PEAK =',G14.6,' AT AN ANGLE =',G14.6/
     1' NEW MAX. PEAK = ',G14.6,' AT AN ANGLE = ',G14.6
     1/' PERCENTAGE IMPROVEMENT IN MAX. PEAK = ',G14.6)
      IF(IRD.GT.NRD)GO TO 26
      IF(DIFF.GT.REQERR)GO TO 26
      IF(KOUT.EQ.1)CALL OUTPUT
      IF(IRD.NE.NRD)GO TO 21
      KK=NRD+1
      REWIND 4
      DO 20 J=1,NRD
      K=(J-1)*N
   20 READ(IO7,115)(V(I+K),I=1,N),F(J)
      WRITE(IO6,118)NRD
  118 FORMAT(//' FINAL SEARCH MADE WITH',I4,' BEST VALUES '//)
      GO TO 23
   21 IF(KOUT.EQ.0)GO TO 4
   15 CONTINUE
   26 CALL OUTPUT
      IF(ISTOP.EQ.1)STOP
      XMAX=0.0
```

```
      YMAX=0.0
      XMIN=0.0
      YMIN=0.0
      TMAX=0.
      TMIN=0.
      JK=0
      NPTS=NV
      CALL QUP(MN)
      DO 6 I=1,NV
      CALL PUP(I,X,Y,Y41)
      X1(I)=X
      Y1(I)=Y
      H1(I)=Y41
      IF(X.GT.XMAX)XMAX=X
      IF(X.LT.XMIN)XMIN=X
      IF(Y.GT.YMAX)YMAX=Y
      IF(Y.LT.YMIN)YMIN=Y
      IF(Y41.GT.TMAX)TMAX=Y41
    6 IF(Y41.LT.TMIN)TMIN=Y41
      INTL=1
      CALL QUP(NINT)
      INTL=0
      DO 8 I=1,NV
      CALL PUP(I,X,Y,Y41)
      X2(I)=X
      Y2(I)=Y
      H2(I)=Y41
      IF(X.GT.XMAX)XMAX=X
      IF(X.LT.XMIN)XMIN=X
      IF(Y.GT.YMAX)YMAX=Y
      IF(Y.LT.YMIN)YMIN=Y
      IF(Y41.GT.TMAX)TMAX=Y41
    8 IF(Y41.LT.TMIN)TMIN=Y41
      LABINC=LABUNB
      AMAX=360.
      AMIN=0.
      LABINC=LABUNB
      NSUPER=1
      LSYMP=0
      NSCALE=1
      DO 32 I=1,NPTS
   32 H3(I)=I-1
      CALL XYPLOT('X-Y   SHAKING FORCE        ',ICOMP,NV,NSCALE
     1,LABINC,LINTYP,LSYMP,ANGST,DEGINC,XPAGE,YPAGE,X2,Y2
     1,XMAX,XMIN,YMAX,YMIN,NSUPER)
      LABINC=LABB
      NSUPER=3
      LSYMP=8
      NSCALE=0
      DO 33 I=1,NPTS
   33 H3(I)=I-1
       CALL XYPLOT('      ',ICOMP,NV,NSCALE
     1,LABINC,LINTYP,LSYMP,ANGST,DEGINC,XPAGE,YPAGE,X1,Y1
     1,XMAX,XMIN,YMAX,YMIN,NSUPER)
      LABINC=LABUNB
      NSUPER=1
      LSYMP=0
      NSCALE=2
      DO 10 I=1,NPTS
   10 H3(I)=I-1
      CALL XYPLOT('MOMENT ABOUT CRANK PIVOT    ',ICOMP,NV,NSCALE
     1,LABINC,LINTYP,LSYMP,ANGST,DEGINC,XPAGE,YPAGE,H3,H2
     1,AMAX,AMIN,TMAX,TMIN,NSUPER)
      LABINC=LABB
      NSUPER=3
      NSCALE=0
      LSYMP=8
      DO 22 I=1,NPTS
   22 H3(I)=I-1
      CALL XYPLOT('      ',ICOMP,NV,NSCALE
     1,LABINC,LINTYP,LSYMP,ANGST,DEGINC,XPAGE,YPAGE,H3,H1
     1,AMAX,AMIN,TMAX,TMIN,NSUPER)
      STOP
      END
      SUBROUTINE KINSOL
      COMMON/KIN/TH2(500),TH3(500),TH4(500),C2(500),C3(500),C4(500)
     1,S2(500),S3(500),S4(500),W3(500),W4(500),W2,W22
     1,ALP3(500),ALP4(500),AL4,AL3,AL2,AL1,NPTS,JK
      COMMON/CONST/PIE,DEG,PI2,INTL
      KK=1
      DO 1 IK=1,NPTS
      K=IK+JK
      S2(K)=SIN(TH2(K))
      C2(K)=COS(TH2(K))
      D1=-AL1+AL2*C2(K)
      D2=AL2*S2(K)
      D3=D1*D1+D2*D2+AL4*AL4-AL3*AL3
      D4=2.0*AL4*D1
      D5=2.0*AL4*D2
      D6=D4*D4+D5*D5
      ARG=D6-D3*D3
      S4(K)=(-D3*D5+KK*D4*SQRT(ARG))/D6
      C4(K)=-(D3+D5*S4(K))/D4
      S3(K)=-(D2+AL4*S4(K))/AL3
      C3(K)=-(D1+AL4*C4(K))/AL3
      C4(K)=-C4(K)
      S4(K)=-S4(K)
      TH4(K)=ATAN2(S4(K),C4(K))
      TH3(K)=ATAN2(S3(K),C3(K))
      SA=S3(K)*C4(K)-C3(K)*S4(K)
      CA=C3(K)*C4(K)+S3(K)*S4(K)
      SB=S2(K)*C4(K)-C2(K)*S4(K)
      SC=S2(K)*C3(K)-C2(K)*S3(K)
      CB=C2(K)*C4(K)+S2(K)*S4(K)
      CC=C2(K)*C3(K)+S2(K)*S3(K)
      W3(K)=-W2*AL2*SB/(AL3*SA)
      W4(K)=-W2*AL2*SC/(AL4*SA)
      W33=W3(K)*W3(K)
      W44=W4(K)*W4(K)
      ALP3(K)=-(W22*AL2*CB+W33*AL3*CA-W44*AL4)/(AL3*SA)
      ALP4(K)=-(W22*AL2*CC-W44*AL4*CA+W33*AL3)/(AL4*SA)
    1 CONTINUE
      RETURN
      END
      SUBROUTINE QUP(J)
      COMMON/KIN/TH2(500),TH3(500),TH4(500),C2(500),C3(500),C4(500)
     1,S2(500),S3(500),S4(500),W3(500),W4(500),W2,W22
     1,ALP3(500),ALP4(500),AL4,AL3,AL2,AL1,NPTS,JK
      COMMON/APEX/ V(1000),N,KCUT
      COMMON/DYN/XA(36),YA(36),AMOM(36),AAM2,AAM3,AAM4,AAI2,AAI3,AAI4
     1,R22,R33,R44,G2,G3,G4,A1,A2,A3,A4,A5,A6,A7,A8,A9,ERRM,ERRF
     1,CA2,SA2,CA5,SA5,CA8,SA8
      COMMON/COUNTR/AM2,RR2,AM3,RR3,AI3,AM4,RR4,AI4,GG2,GG3,GG4
      COMMON/AAA/T2,T3,T4,RHC,GRAV
      COMMON/RESTR/VLOW(10),VHIGH(10)
      COMMON/CONST/PIE,DEG,PI2,INTL
      DO 1 I=1,3
      K=(I-1)*3+1
      L=K+2
      IF(V(K+J).LT.VLOW(K))V(K+J)=VLOW(K)
      IF(V(L+J).LT.VLOW(L))V(L+J)=VLOW(L)
      IF(V(K+J).GT.VHIGH(K))V(K+J)=VHIGH(K)
      IF(V(L+J).GT.VHIGH(L))V(L+J)=VHIGH(L)
    1 CONTINUE
      A1=V(1+J)
      A2=V(2+J)
      A3=V(3+J)
      A4=V(4+J)
      A5=V(5+J)
      A6=V(6+J)
      A7=V(7+J)
      A8=V(8+J)
      A9=V(9+J)
      AI2=EQUA(A1,A2,A3,RR2,AM2,GG2,RHO,T2,V(2+J),R22,AAM2,AAI2,G2)
      AI3=EQUA(A4,A5,A6,RR3,AM3,GG3,RHO,T3,V(5+J),R33,AAM3,AAI3,G3)
      AI4=EQUA(A7,A8,A9,RR4,AM4,GG4,RHO,T4,V(8+J),R44,AAM4,AAI4,G4)
      CA2=COS(GG2)
      SA2=SIN(GG2)
      CA5=COS(GG3)
      SA5=SIN(GG3)
      CA8=COS(GG4)
      SA8=SIN(GG4)
      RETURN
      END
      SUBROUTINE PUP(I,S,R,Y41)
      COMMON/KIN/TH2(500),TH3(500),TH4(500),C2(500),C3(500),C4(500)
     1,S2(500),S3(500),S4(500),W3(500),W4(500),W2,W22
     1,ALP3(500),ALP4(500),AL4,AL3,AL2,AL1,NPTS,JK
      COMMON/APEX/ V(1000),N,KOUT
      COMMON/DYN/XA(36),YA(36),AMOM(36),AAM2,AAM3,AAM4,AAI2,AAI3,AAI4
     1,R22,R33,R44,G2,G3,G4,A1,A2,A3,A4,A5,A6,A7,A8,A9,ERRM,ERRF
     1,CA2,SA2,CA5,SA5,CA8,SA8
      COMMON/COUNTR/AM2,RR2,AM3,RR3,AI3,AM4,RR4,AI4,GG2,GG3,GG4
      COMMON/AAA/T2,T3,T4,RHC,GRAV
      IP=I+JK
      W44=W4(IP)*W4(IP)
      W33=W3(IP)*W3(IP)
      E1=S2(IP)
      E2=C2(IP)
      E3=S3(IP)
      E4=C3(IP)
      E5=S4(IP)
      E6=C4(IP)
      SG2=E1*CA2+E2*SA2
      CG2=E2*CA2-E1*SA2
      SG3=E3*CA5+E4*SA5
      CG3=E4*CA5-E3*SA5
      SG4=E5*CA8+E6*SA8
      CG4=E6*CA8-E5*SA8
      B4=ALP4(IP)
      B3=ALP3(IP)
      A41=RR4*(W44*CG4+B4*SG4)
      A42=RR4*(W44*SG4-B4*CG4)
      A31=RR3*(W33*CG3+B3*SG3)
      DA31=AL2*W22*E2*AM3
      A32=RR3*(W33*SG3-B3*CG3)
      DA32=AL2*W22*E1*AM3
      A21=RR2*W22*CG2
      A22=RR2*W22*SG2
      Y34=AI3*B3*AL4*E5+AI4*E4*AL3*E3
      F=E1*E4-E2*E3
      E=E2*E4+E1*E3
      G=F*CA5-E*SA5
      C=RR3*AL2*AL4*E5*W22*G
      D=AL3*AL4*(E3*E6-E4*E5)
      Y34=(Y34-C)/D
      Y41=(A42+Y34)*AL1/GRAV
      S=(A31+DA31+A41+A21)/GRAV
      R=(A32+DA32+A42+A22)/GRAV
      RETURN
      END
      FUNCTION ERROR(J)
      COMMON/STR/XE2,YE2,ZE2
      COMMON/KIN/TH2(500),TH3(500),TH4(500),C2(500),C3(500),C4(500)
     1,S2(500),S3(500),S4(500),W3(500),W4(500),W2,W22
     1,ALP3(500),ALP4(500),AL4,AL3,AL2,AL1,NPTS,JK
      COMMON/APEX/ V(1000),N,KOUT
      COMMON/DYN/XA(36),YA(36),AMOM(36),AAM2,AAM3,AAM4,AAI2,AAI3,AAI4
     1,R22,R33,R44,G2,G3,G4,A1,A2,A3,A4,A5,A6,A7,A8,A9,ERRM,ERRF
     1,CA2,SA2,CA5,SA5,CA8,SA8
      COMMON/COUNTR/AM2,RR2,AM3,RR3,AI3,AM4,RR4,AI4,GG2,GG3,GG4
      COMMON/AAA/T2,T3,T4,RHC,GRAV
      INTL=0
      P1=0.
      P2=0.
      P3=0.
      CALL QUP(J)
      DO 2 I=1,NPTS
      CALL PUP(I,S,R,Y41)
      SS=XA(I)+S
      RR=YA(I)+R
      T=AMOM(I)+Y41
      P1=T*T+P1
      P2=SS*SS+P2
    2 P3=RR*RR+P3
      P=(ERRM*P1/ZE2)+ERRF*((P2/XE2)+(P3/YE2))
      ERROR=P
      RETURN
      END
      SUBROUTINE SIMPLX(EPS1)
C     ACTUAL   NO. OF PARAMETERS = N
C     DIMENSION OF 'V' IS V(N*(N+3))
C     DIMENSION OF 'F' IS F(N+3)
C     DIMENSION OF 'VC' IS VC(N)
      COMMON /CONV/ FMIN,NITER,NPRINT,NCHECK
      COMMON /DEVICE/ IO1,IO2,IO3,IO4,IO5,IO6,IO7,IO8,IO9
      COMMON /OUT/ IOUT1,IOUT2
      COMMON/APEX/ V(1000),N,KCUT
      COMMON /SMPLX/ EXPAND,SHRINK,REDUCE,NPRT
      COMMON/FUNC/ F(30),MN,MIN
      COMMON/CONST/PIE,DEG,PI2,INTL
      DATA ITER,MAX,MM/0,2*1/
  101 FORMAT('ERROR VALUES,F1...F(N+1) OF THE INITIAL'
     1,' GUESSES ARE :-')
 119  FORMAT('THE PARAMETERS ASSOCIATED WITH THE CURRENT MAX. ERROR')
 120  FORMAT('THE PARAMETERS ASSOCIATED WITH THE CURRENT MIN. ERROR')
 104  FORMAT('MAX. ERROR VALUE IN CURRENT SIMPLEX',/,G15.7)
 106  FORMAT(' ITER=',I4,'  MAX. ERR=',G15.7,'MIN. ERR=',G15.7)
 105  FORMAT('MIN. ERROR VALUE IN CURRENT SIMPLEX',/,G15.7)
  117 FORMAT('CONVERGENCE***SUCCESS****',/)
 112  FORMAT('0ITERATIONS =',I10)
  121 FORMAT('FINAL VALUES OF THE OPTIMISATION CONTROLLING'
     1,' PARAMETERS CORRESP. TO MIN. ERROR')
  115 FORMAT(1X,5G15.7)
  124 FORMAT(' CONVERGENCE ***FAILURE*** IN THIS AREA ')
C
C     CALCULATE STARTING VERTICES AND FUNCTION VALUES
C
      ITER=0.
      X=EXPAND
      Y=SHRINK
      Z=REDUCE
      MIN=1
      N1 = N + 1
      N2=N+2
      N3 = N + 3
      NN = N*N1
      NNN=N+NN
      DO 2 I=1,N1
 2    F(I) = ERROR((I-1)*N)
      WRITE(IO9,101)
      WRITE(IO9,115)(F(I),I=1,N1)
C
C     CALCULATE HIGHEST(MAX),NEXT HIGHEST(MM),&LOWEST(MIN) FUNCTION
C
      IOUT2=1
      DO 4 I=1,N1
      IF(F(I).GT.F(MAX))MAX=I
      IF(F(I).LT.F(MIN))MIN=I
 4    CONTINUE
      MM=MIN
      DO 5 I=1,N1
      IF(I.EQ.MAX) GO TO 5
      IF(F(I).GT.F(MM))MM=I
 5    CONTINUE
      MN=(MIN-1)*N
      MX=(MAX-1)*N
C
C     BEGIN NEW ITERATION
C
    3 CONTINUE
      KGO=1
C     SELECT NEW VERTEX V*=V(N2) & EVALUATE F*=F(N2)
C
      DO 9 I=1,N
      V(I+NN) = (N/X-N1)*V(I+MX)
      DO 8 J=1,N1
 8    V(I+NN) = V(I+NN) + V(I+(J-1)*N)
    9 V(I+NN) = V(I+NN)*X/N
      F(N2) = ERROR(NN)
```

```
C
C     CHECK IF F* .LT. F(MIN)
C
      IF(F(N2).GE.F(MIN)) GC TO 999
C
C     DIRECTION IS ADVANTAGEOUS SO TRY DOUBLE MOVE TO V**=2V*-V(MAX)
C
      DO 10 I=1,N
   10 V(I+NNN) = 2.*V(I+NN) - V(I+MX)
      F(N3) = ERROR(NNN)
C
C     CHECK IF DOUBLE MOVE IS IMPROVEMENT ON ORIGINAL MOVE
C
      IF(F(N3)-F(N2)) 11,12,12
C
C     DIRECTION IS GOOD SO TRY QUAD. MOVE TO V*=2V**-V(MAX)
C
 11   DO 30 I=1,N
   30 V(I+NN)=2.*V(I+NNN)-V(I+MX)
      F(N2)=ERROR(NN)
      IF(F(N2)-F(N3))12,32,32
 32   NEW = N3
      GO TO 13
 12   NEW = N2
C
C     RESET V(MAX) TO NEW VALUES
C
 13   CONTINUE
      DO 14 I=1,N
 14   V(I+MX) = V(I+(NEW-1)*N)
      F(MAX) = F(NEW)
      GO TO 24
C
C     ORIGINAL MOVE NO GOOD. CHECK IF REFLECTED POINT
C     WILL HAVE THE LARGEST FUNCTION VALUE
C
 999  X1=X-1.
 149  IF(F(N2) - F(MM)) 15,15,16
 15   NEW = N2
      GO TO 13
C
C     V* HAS LARGEST FUNCTICN VALUE
C     PREPARE FOR THE CONTRACTION OF SIMPLEX ETC.
C
 16   MARKER=-1
      IF(F(N2).GT.F(MAX))MARKER=1
      IF(X1.LT.1.99)GO TO 500
      DO 600 I=1,N
 600  V(I+NN)=((X-1.)*V(I+NN)+V(I+MX))/X
      F(N2)=ERROR(NN)
      X1=X1-1.
      GO TO 149
 500  DO 18 I=1,N
C
C     SET V** TEMPORARILY TC V=(V* + VMAX)/2
C
      V(I+NNN) = (V(I+NN) + V(I+MX))/2.
C
C     SET VMAX TO V* IF NECESSARY AND FINALISE CALC. FOR V** & F**
C
      IF(MARKER) 17,17,18
 17   V(I+MX) = V(I+NN)
 18   V(I+NNN)=V(I+NNN)*(1.-Y)+Y*V(I+MX)
      F(N3) = ERROR(NNN)
      IF(MARKER.EQ.-1)F(MAX)=F(N2)
C
C     IF V** STILL HAS GREATEST FUNCTION VALUE, HALVE SIMPLEX ABOUT V(MI
C
      IF(F(N3)-F(MAX)) 19,19,20
 19   NEW = N3
      GO TO 13
C
C     REDUCE SIZE OF SIMPLEX
C
   20 CONTINUE
      DO 23 I=1,N1
      IF(I-MIN) 21,23,21
 21   DO 22 J=1,N
      INX=J+(I-1)*N
 22   V(INX) = Z*V(INX)+(1.-Z)*V(J+MN)
      F(I) = ERROR((I-1)*N)
 23   CONTINUE
C
   24 DO 6 I=1,N1
      IF(F(I).GT.F(MAX))MAX=I
      IF(F(I).LT.F(MIN))MIN=I
 6    CONTINUE
      MM=MIN
      DO 7 I=1,N1
      IF(I.EQ.MAX) GO TO 7
      IF(F(I).GT.F(MM))MM=I
 7    CONTINUE
      MN=(MIN-1)*N
      MX=(MAX-1)*N
C
C     TEST FOR CONVERGENCE
C
      ITER=ITER+1
      IF(ITER.LT.NITER) GO TC 41
      WRITE(IO6,124)
      WRITE(IO7,115)(V(I+MN),I=1,N),F(MIN)
      KOUT=0
      RETURN
C
C     WRITE PROGRESS OF OPTIMISATION EVERY NPRINT ITERATIONS
C
   41 IF(ITER/NPRINT-(ITER-1)/NPRINT.EQ.0) GO TO 40
      WRITE(IO9,112)ITER
      I=MX+1
      J=MAX*N
      WRITE(IO9,119)
      WRITE(IO9,115)(V(K),K=I,J)
      WRITE(IO9,104)F(MAX)
      I=MN+1
      J=MIN*N
      WRITE(IO9,120)
      WRITE(IO9,115)(V(K),K=I,J)
      WRITE(IO9,105)F(MIN)
      GO TO 43
 40   IF(ITER/NPRT-(ITER-1)/NPRT.EQ.0)GO TO 43
      WRITE(IO9,106)ITER,F(MAX),F(MIN)
C
C     CHECK CONVERGENCE EVERY NCHECK ITERATIONS
C
 43   IF(ITER/NCHECK-(ITER-1)/NCHECK.EQ.0) GO TO 3
      IF(F(MIN).LE.FMIN) GO TO 26
C
C     CONVERGENCE IF ABS.(FMAX-FMIN) LESS THAN ABS.(EPS1 * FMIN)
C
      IF(F(MAX)-F(MIN).GT.ABS(EPS1*F(MIN)))GO TO 3
C
C     CONVERGENCE IF ABS.(VMAX-VMIN) LESS THAN ABS.(EPS1 * VMIN)
C
      DO 99 I=1,N
      IF(ABS(V(I+MX)-V(I+MN)).GT.ABS(EPS1*V(I+MN)))GO TO 3
 99   CONTINUE
   26 WRITE(IO6,117)
   44 WRITE(IO6,112)ITER
      WRITE(IO7,115)(V(I+MN),I=1,N),F(MIN)
      WRITE(IO6,121)
      KOUT=1
      IOUT1=1
      RETURN
      END
      SUBROUTINE RESULT(NPTS,MN,SMAX,RMAX,ZMAX,VMAX,TETA,X,Y,Z)
      COMMON/CONST/PIE,DEG,PI2,INTL
      Y=0.
      Z=0.
      X=0.
      SMAX=0.
      RMAX=0.
      ZMAX=0.
      VMAX=0.
      CALL QUP(MN)
      DO 1 I=1,NPTS
      CALL PUP(I,S,R,Y41)
      IF(ABS(SMAX).LT.ABS(S))SMAX=S
      IF(ABS(RMAX).LT.ABS(R))RMAX=R
      IF(ABS(ZMAX).LT.ABS(Y41))ZMAX=Y41
      X2=S*S
      Y2=R*R
      Z2=Y41*Y41
      V=SQRT(X2+Y2)
      IF(V.LT.VMAX)GO TO 2
      VMAX=V
      XS=S
      YS=R
    2 X=X+X2
      Y=Y+Y2
    1 Z=Z2+Z
      X=SQRT(X)/NPTS
      Y=SQRT(Y)/NPTS
      Z=SQRT(Z)/NPTS
      TETA=ATAN2(YS,XS)
      RETURN
      END
      FUNCTION EQUA(A,B,C,D,E,P,F,G,H,RI,AMI,AI,GI)
C SEGMENTAL COUNTERWEIGHT
      COMMON/CONST/PIE,DEG,PI2,INTL
      PIH=0.5*PIE
      S1=SIN(C)
      C1=COS(C)
      S2=2.0*S1*C1
      C2=1.0-2.0*S1*S1
      S3=S2*C1+C2*S1
      C3=C2*C1-S2*S1
      S4=2.0*S2*C2
      AA=A*A
      AJ=F*AA*G
      ANG=PIE-C
      IF(C.GT.PIH)GO TO 8
      AK=AJ*AA*(ANG*(1.0+0.5*C2)+0.75*S2)
      E=AJ*(ANG+0.5*S2)
      D=AJ*A*(ANG*C1+0.75*S1+S3/12.0)
      GO TO 4
    8 X=1.0/6.0
      U=0.2
      Z=0.2
      ZU=Z*U
      ZZ=ZU*Z*Z
      AK=AJ*AA*(0.5*ANG+(X-ZZ-0.25*ZU)*S2+(X-0.5*ZU+2.0*ZZ)*0.25*S4)
      E=AJ*(ANG+(0.5-ZU)*S2)
      D=AJ*A*((2.0+ZU)*S1*0.25+(3.0*ZU-2.0)*S3/12.0)
    4 IF(D.GE.0.0)GO TO 1
      D=-D
      B=B+PIE
    1 IF(B.LE.PI2)GO TO 2
      B=B-PI2
      GO TO 1
    2 IF(B.GE.0.0)GO TO 3
      B=B+PI2
      GO TO 2
    3 H=B
      IF(INTL.EQ.0)GO TO 7
      P=GI
      D=AMI*RI
      E=AMI
      AK=AI
      GO TO 9
    7 EH=AMI*RI
      AH=EH*COS(GI)+D*COS(B)
      BH=EH*SIN(GI)+D*SIN(B)
      E=E+AMI
      IF(AH.NE.0.0.OR.BH.NE.0.0)GO TO 5
      B=0.
      D=0.
      GO TO 6
    5 P=ATAN2(BH,AH)
      D=SQRT(BH*BH+AH*AH)
    6 AK=AK+AI
    9 EQUA=AK
      RETURN
      END
      SUBROUTINE OUTPUT
      COMMON/DEVICE/ IO1,IO2,IO3,IO4,IO5,IO6,IO7,IO8,IO9
      COMMON/FUNC/ F(30),MN,MIN
      COMMON/TITLE/KTITLE(6)
      COMMON/COUNTR/AM2,RR2,AM3,RR3,AI3,AM4,RR4,AI4,GG2,GG3,GG4
      COMMON/CONST/PIE,DEG,PI2,INTL
      COMMON/APEX/ V(1000),N,KOUT
      DIMENSION ITITLE(6),PAR(20)
      DO 181 I=1,N
  181 PAR(I)=V(I+MN)
      WRITE(IO6,288)F(MIN)
  288 FORMAT(1H ,//30X,' FINAL MIN. ERROR VALUE = ',G14.6//20X,
     1' COUNTERWEIGHT PARAMETERS REQUIRED FOR DYNAMIC SYNTHESIS '
     1//10X,' COUNTERWEIGHT RADII ',8X,' ANGULAR OFFSET ',8X,
     1'SEGMENTAL HEIGHT ',8X,' SEGMENTAL ANGLE')
      DO 289 I=1,3
      K=(I-1)*3+1
      J=K+1
      L=K+2
      M=(I-1)*2+1
      LM=M+1
      PAR(J)=ANG(PAR(J))
      HE=PAR(K)*(COS(PAR(L))+1.0)
      PAR(L)=ANG(PAR(L))
  289 WRITE(IO6,290)KTITLE(M),KTITLE(LM),PAR(K),PAR(J),HE,PAR(L)
  290 FORMAT(1H ,2A4,5X,G14.6,10X,G14.6,10X,G14.6,10X,G14.6)
      CALL QUP(MN)
      WRITE(IO6,292)
      GG2=ANG(GG2)
      GG3=ANG(GG3)
      GG4=ANG(GG4)
  292 FORMAT(1H ,///' N.B. MCMENTS OF MASS AND INERTIA ARE ABOUT JOINTS'
     1,' AND NOT THE CENTRES CF GRAVITY'
     1,///17X,'MASS   ',4X,' MCMENT OF MASS'
     1'    ANGULAR OFFSET',1X,' MCMENT OF INERTIA ')
      WRITE(IO6,291)KTITLE(1),KTITLE(2),AM2,RR2,GG2
      WRITE(IO6,291)KTITLE(3),KTITLE(4),AM3,RR3,GG3,AI3
     1,KTITLE(5),KTITLE(6),AM4,RR4,GG4,AI4
  291 FORMAT(1H ,2A4,5X,4G15.6)
      RETURN
      END
      FUNCTION ANG(A)
      COMMON/CONST/PIE,DEG,PI2,INTL
    2 IF(A.GT.0.)GO TO 1
      A=A+PI2
      GO TO 2
    1 IF(A.LT.PI2)GO TO 3
      A=A-PI2
      GO TO 1
    3 A=A*DEG
      ANG=A
      RETURN
      END
END OF FILE
```

```
58 NO. QMCG8          UNIVERSITY OF MICHIGAN TERMINAL SYSTEM AT NUMAC                    12:22:35    MON  5-FEB

                         USER: MCG8
                         PROJECT NO: NMC0

                         **** OBTAINED AT      12:22:35     MON  5-FEB-79
                         **** RELEASED AT      12:22:51     MON  5-FEB-79
                         **** LINES PRINTED      875
                         **** PAGES PRINTED       16
```

```
 QQQQQQQQQQ    MM          MM    CCCCCCCCCC     GGGGGGGGGG     8888888888
QQQQQQQQQQQQ   MMM        MMM   CCCCCCCCCCCC   GGGGGGGGGGGG   888888888888
QQ        QQ   MMMM      MMMM   CC        CC   GG        GG   88        88
QQ        QQ   MM MM    MM MM   CC             GG             88        88
QQ        QQ   MM  MMMM   MM    CC             GG              88      88
QQ        QQ   MM   MM    MM    CC             GG               88888888
QQ        QQ   MM         MM    CC             GG     GGGGG     88888888
QQ    QQ  QQ   MM         MM    CC             GG     GGGGG    88      88
QQ     QQ QQ   MM         MM    CC             GG        GG   88        88
QQ      QQQQ   MM         MM    CC        CC   GG        GG   88        88
QQQQQQQQQQQQ   MM         MM    CCCCCCCCCCCC   GGGGGGGGGGGG   888888888888
 QQQQQQQQ QQ   MM         MM     CCCCCCCCCC     GGGGGGGGGG     8888888888
```

```
MM          MM    CCCCCCCCCC     GGGGGGGGGG     8888888888
MMM        MMM   CCCCCCCCCCCC   GGGGGGGGGGGG   888888888888
MMMM      MMMM   CC        CC   GG        GG   88        88
MM MM    MM MM   CC             GG             88        88
MM  MMMM   MM    CC             GG              88      88
MM   MM    MM    CC             GG               88888888
MM         MM    CC             GG     GGGGG     88888888
MM         MM    CC             GG     GGGGG    88      88
MM         MM    CC             GG        GG   88        88
MM         MM    CC        CC   GG        GG   88        88
MM         MM    CCCCCCCCCCCC   GGGGGGGGGGGG   888888888888
MM         MM     CCCCCCCCCC     GGGGGGGGGG     8888888888
```

COPY 01

PROGRAM B

MEDIC


```
 QQQQQQQQQ   MM        MM   CCCCCCCCCC    GGGGGGGGGG    8888888888
QQQQQQQQQQQ  MMM      MMM  CCCCCCCCCCCC  GGGGGGGGGGGG  888888888888
QQ       QQ  MMMM    MMMM  CC        CC  GG        GG  88        88
QQ       QQ  MM MM  MM MM  CC            GG             88      88
QQ       QQ  MM  MMMM  MM  CC            GG             88      88
QQ       QQ  MM   MM   MM  CC            GG              88888888
QQ       QQ  MM        MM  CC            GG     GGGGG    88888888
QQ   QQ  QQ  MM        MM  CC            GG     GGGGG   88      88
QQ    QQ QQ  MM        MM  CC            GG        GG  88        88
QQ     QQQQ  MM        MM  CC        CC  GG        GG  88        88
QQQQQQQQQQQ  MM        MM  CCCCCCCCCCCC  GGGGGGGGGGGG  888888888888
 QQQQQQQQ QQ MM        MM   CCCCCCCCCC    GGGGGGGGGG    8888888888
```

```
C MULTIPLE ENHANCEMENT OF DYNAMICS VIA INERTIAL CHANGES(MEDIC
C MEDIC:COUNTERWEIGHT SYNTHESIS PROGRAM FOR FOUR AND SIX BAR LINKAGES
C PROGRAM B
C
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION ITITLE(18),ISTOR(20)
      COMMON/KIN/INV1,INV2,TH(542,5),S(542,5),C(542,5),W(542,5),W1,STH
     1,ALP(542,5),AJ(542,5),AL(10),SINE2,COSE2,SINE3,COSE3,SINB3,COSB3
      COMMON/CONTRL/NPTS,ILINK,ILOCI,NPLOTS,IK,JX,IQ,IJJ,IVAR,INPT,ISW
     1,ISET,ISYN,IDEV,IKK,IBAL,IMOM,ITORQ,IDOT,MONITR,IOUT,JXX,IDMSP,IAN
      COMMON/PARAM/AM1,AM2,AM3,AM6,AM5,AI2,AI3,AI6,AI5
     1,R1,R2,R3,R6,R5,RM1,RM2,RM3,RM6,RM5,G1,G2,G3,G6,G5
     1,AK1,AK2,AK3,AK4,AK5,AK6,AS2,AS3,AS4,AS5,AS6
     1,DELT2,DELT3,DELT4,DELT5,DELT6
      COMMON/TRN/AAM1,AAM2,AAM3,AAM6,AAM5,AAI2,AAI3,AAI6,AAI5
     1,RR1,RR2,RR3,RR6,RR5,RRM1,RRM2,RRM3,RRM6,RRM5,GG1,GG2,GG3,GG6,GG5
     1,AAK1,AAK2,AAK3,AAK4,AAK5,AAK6,AAS2,AAS3,AAS4,AAS5,AAS6
     1,ADELT2,ADELT3,ADELT4,ADELT5,ADELT6
      COMMON/FORCE/ZX(552,8),ZY(552,8),DX(552,8),DY(552,8),TOR(552,8)
     1,GRAV,FDEG(7),XDEG,YDEG,ADEG,TDEG,HOLD,AERR,ISRT
      COMMON/STORE/AM(40),R(40),G(40),AI(40),AK(60),AS(60),DELT(60)
      COMMON/CONST/PIE,PIE2,RAD,DEG,PY
      COMMON/SYNTH/FP(7),ITP(7),KV(25,5)
      NAMELIST /MNCTRL/ ILINK,IOUT,ISYN,ILOCI,INPT
     1,IGRAV
     1     /ILINK6/SPEED,AL1,AL2A,AL2B,AL2C,AL3A,AL3B,AL3C,AL4
     1,AL5,AL6,AM1,R1,G1,AM2,R2,G2,AI2,AM3,R3,G3,AI3,AM6,R5
     1,G6,AI6,AM5,R5,G5,AI5,AK1,AK2,AK3,AK4,AK5,AK6,AS2,AS3,AS4,AS5,AS6
     1,DELT2,DELT3,DELT4,DELT5,DELT6
     1     /ILINK4/SPEED,AL1,AL2,AL3,AL4,AM1,R1,G1,AM2,R2,G2,AI2,
     1AM3,R3,G3,AI3,AK1,AK2,AK3,AS2,AS3,DELT2,DELT3
     1     /ADLNK/ADM1,ADR1,ADG1,ADM2,ADR2,ADG2,ADI2,ADM3
     1,ADR3,ADG3,ADI3,ADM6,ADR6,ADG6,ADI6,ADM5,ADR5,ADG5,ADI5
     1,ADK1,ADK2,ADK3,ADK4,ADK5,ADK6,ADS2,ADS3,ADS4,ADS5,ADS6
     1,ADELT2,ADELT3,ADELT4,ADELT5,ADELT6
      SPEED=9.5493D0
      PIE=4.0D0*DATAN(1.0D0)
      PIE2=PIE*2.0D0
      PY=0.999999D0*PIE2
      RAD=PIE/1.8D2
      DEG=180/PIE
      INV1=1
      INV2=1
      NPTS=361
      ISW=1
C IJJ IS ASSOCIATED WITH ADDED INERTIA ARRAYS
C IKK IS ASSOCIATED WITH THE VERTICIES IN SIMPLEX
      AM1=0.0D0
      AM2=0.0D0
      AM3=0.0D0
      AM6=0.0D0
      AM5=0.0D0
      R1=0.0D0
      R2=0.0D0
      R3=0.0D0
      R6=0.0D0
      R5=0.0D0
      G1=0.0D0
      G2=0.0D0
      G3=0.0D0

      G6=0.0D0
      G5=0.0D0
      AI2=0.0D0
      AI3=0.0D0
      AI6=0.0D0
      AI5=0.0D0
      AK1=0.0D0
      AK2=0.0D0
      AK3=0.0D0
      AK4=0.0D0
      AK5=0.0D0
      AK6=0.0D0
      AS2=0.0D0
      AS3=0.0D0
      AS4=0.0D0
      AS5=0.0D0
      AS6=0.0D0
      DELT2=0.0D0
      DELT3=0.0D0
      DELT4=0.0D0
      DELT5=0.0D0
      DELT6=0.0D0
  213 FORMAT(//40X,'** NO SPRINGS **')
  214 FORMAT(30X,33('*')/30X,'* SPRING RATES AND PHASE ANGLES *'/30X,
     133('*')/15X,' AR2=',G14.6,' DELT2=',G14.6,' AR3=',G14.6,
     1' DELT3=',G14.6)
  215 FORMAT(15X,' AR4=',G14.6,' DELT4=',G14.6,' AR5=',G14.5,' DELT5=',
     1G14.6/15X,' AR6=',G14.6,' DELT6=',G14.6)
  216 FORMAT(//36X,23('*')/36X,'* SPRING STIFFNESSESS *'/36X,23('*')
     1/20X,' AS2=',G14.6,' AS3=',G14.6)
  217 FORMAT(15X,' AS4=',G14.6,' AS5=',G14.6,' AS6=',G14.6)
      ILINK=0
      ILOCI=1
      IOUT=1
      ISYN=0
      IDAMP=0
      ISPRNG=0
      IDOT=1
      INPT=0
      IGRAV=0
      READ(5,100)(ITITLE(I),I=1,18)
  100 FORMAT(18A4)
      WRITE(6,103)(ITITLE(I),I=1,18)
  103 FORMAT(20X,74('*')/20X,'*',18A4,'*'/20X,74('*'))
      READ(5,MNCTRL)
  200 FORMAT(///40X,' ************* '/40X,' *INPUT DATA *'/
     140X,' ************* '//39X,16('*')/39X,'* LINK LENGTHS *'/
     139X,16('*')//)
  201 FORMAT(4X,' LINK L1= ',G14.6,' LINK L2A= ',G14.6,
     1' LINK L2B= ',G14.6,' LINK L2C= ',G14.6/
     14X,' LINK L3A= ',G14.6,' LINK L3B= ',G14.6,
     1' LINK L3C= ',G14.6/4X,' LINK L4= ',G14.6,
     1' LINK L5= ',G14.6,' LINK L6= ',G14.5/20X,'SPEED=',
     1G14.6,' R.P.M.'//)
  202 FORMAT(4X,' CRANK L1=',G14.6,
     1' COUPLER L2=',G14.6,' ROCKER L3=',G14.6,
     1' FRAME L4=',G14.6/20X,'SPEED=',G14.6,' R.P.M.'//)
  205 FORMAT(//34X,30('*')/34X,'* LINKAGE INERTIA PARAMETERS *'/34X,
     130('*')/6X,'LINK',13X,' MASS',11X,' RADIAL CO-ORD',6X,
     1' ANGULAR OFFSET',7X,' M OF INERTIA')

  209 FORMAT(5X,I4,5X,4(6X,G14.6))
  210 FORMAT(///35X,24('*')/35X,'* DAMPING COEFFICIENTS *'
     1/35X,24('*')//20X,' AC1= ',G14.6,
     1' AC2= ',G14.6,' AC3= ',G14.6)
  211 FORMAT(20X,' AC4= ',G14.6,' AC5= ',G14.6,' AC6= ',G14.6//)
  206 FORMAT(///39X,'*****************'/39X,
     1'* DAMPING RATIOS *'/39X,
     1'*****************'//20X,' AK1= ',G14.6,
     1' AK2= ',G14.6,' AK3= ',G14.6)
  207 FORMAT(20X,' AK4= ',G14.6,' AK5= ',G14.6,' AK6= ',G14.5//)
  212 FORMAT(//40X,'** NO DAMPING **')
C
C************************************************
C
C NPTS= POINTS OF EVALUATION DURING SYNTHESIS
C ILINK=0 SIX-BAR LINKAGE
C      1=FOUR-BAR LINKAGE
C IOUT= 0 INPUT DATA WRITTEN OUT
C     =1 WRITTING OF INPUT DATA SUPRESSED
C ISYN NO. OF SYNTHESIS RUNS
C IPLOT= NUMBER OF SETS OF GRAPHS TO BE PLOTTED
C        (MUST INCLUDE ORIGINAL LINKAGE IF REQUESTED,IE IUNBAL=0)
C INPT= NUMBER OF SETS OF ADDED INERTIA
C       TO BE READ IN
C IGRAV 0= GRAVITY SET TO ZERO
C       1= GRAVITY SET TO 9.80665 METRES/SEC/SEC
C       2= GRAVITY SET TO 32.174 FEET/SEC/SEC
C
C***********************************************
C
      IAN=0
      IF(ILINK.NE.0)GOTO12
      READ(5,ILINK6)
      AL(2)=AL2C
      AL(3)=AL3C
      AL(4)=AL4
      AL(5)=AL2A
      AL(6)=AL5
      AL(7)=AL5
      AL(8)=AL3B
      AL(9)=AL2B
      AL(10)=AL3A
      GOTO13
   12 READ(5,ILINK4)
      AL(2)=AL2
      AL(4)=AL4
      AL(3)=AL3
   13 AL(1)=AL1
      W1=SPEED*PIE/3.0D1
      STR1=AK1+AK2+AK3+AK4+AK5+AK6
      STR2=AS2+AS3+AS4+AS5+AS6
      IF(IOUT.EQ.1)GOTO8
      WRITE(6,200)
      IF(ILINK.EQ.0)WRITE(6,201)AL(1),AL(5),AL(9),AL(2)
     1,AL(10),AL(8),AL(3),AL(4),AL(7),AL(6),SPEED
      IF(ILINK.EQ.1)WRITE(6,202)AL(1),AL(2),AL(3),AL(4),SPEED
      WRITE(6,205)
      I=1
      WRITE(6,209)I,AM1,R1,G1
      I=2

      J=3
      WRITE(6,209)I,AM2,R2,G2,AI2
      WRITE(6,209)J,AM3,R3,G3,AI3
      IF(ILINK.EQ.1)GOTO23
      I=5
      J=6
      WRITE(6,209)I,AM5,R5,G5,AI5
      WRITE(6,209)J,AM6,R6,G6,AI6
   23 IF(STR1.NE.0.0)GOTO25
      WRITE(6,212)
      IDAMP=1
      GOTO8
   25 WRITE(6,206)AK1,AK2,AK3
      IF(ILINK.EQ.0)WRITE(6,207)AK4,AK5,AK6
    8 IF(ILINK.EQ.1)GOTO3
      RM6=AM6*R6
      RM5=AM5*R5
      G6=G6*RAD
      G5=G5*RAD
      ETA2=AL(2)*AL(2)+AL(5)*AL(5)-AL(9)*AL(9)
      COSE2=ETA2/(2.0*AL(2)*AL(5))
      SINE2=DSQRT(1.0D0-COSE2*COSE2)
      ETA3=AL(8)*AL(8)+AL(3)*AL(3)-AL(10)*AL(10)
      COSE3=ETA3/(2.0*AL(3)*AL(8))
      SINE3=DSQRT(1.0D0-COSE3*COSE3)
      SINB3=AL(8)*SINE3/AL(10)
      COSB3=(AL(10)*AL(10)+AL(3)*AL(3)-AL(8)*AL(8))/(2.0D0*AL(10)*AL(3))
    3 RM1=R1*AM1
      RM2=R2*AM2
      RM3=R3*AM3
      G1=G1*RAD
      G2=G2*RAD
      G3=G3*RAD
      GRAV=0.0D0
      IF(IGRAV.EQ.0)GOTO35
      IZZ=1
      IF(IGRAV.GT.0)GOTO36
      IGRAV=-IGRAV
      IZZ=-1
   36 IF(IGRAV.EQ.1)GRAV=9.80665D0
      IF(IGRAV.EQ.2)GRAV=32.174D0
      GRAV=IZZ*GRAV
   35 IK=5
      JX=6
      IF(ILINK.EQ.0)GOTO17
      JX=3
      IK=3
   17 IJJ=0
      JXX=0
      IKK=0
      IB=0
      DO 1 I=1,NPTS
    1 TH(I,1)=(I-1)*RAD
      CALL KINSOL(0)
      IDMSP=0
      CALL DYNMIC(-1,0)
      VWRK=0.0D0
      WK1=0.0D0
      WK2=0.0D0
      WK3=0.0D0

      WK4=0.0D0
      WK5=0.0D0
      WK6=0.0D0
      WS2=0.0D0
      WS3=0.0D0
      WS4=0.0D0
      WS5=0.0D0
      WS6=0.0D0
      MPTS=NPTS-1
      DO 20 I=1,MPTS
      I2=I+1
      TH1=TH(I,1)
      TH11=TH(I2,1)
      TH2=TH(I,2)
      TH21=TH(I2,2)
      TH3=TH(I,3)
      TH31=TH(I2,3)
      AX=(W(I2,2)+W(I,2)-W1-W1)*0.5D0
      WK1=WK1+AX*(TH21-TH11-TH2+TH1)
      AX=(W(I2,3)+W(I,3)-W(I2,2)-W(I,2))*0.5D0
      AY=(TH31+TH3-TH21-TH2)*0.5D0+DELT2
      WK2=WK2+AX*(TH31-TH21-TH3+TH2)
      WS2=WS2+DABS(AY*(TH31-TH3+TH2-TH21))
      AX=(W(I2,3)+W(I,3))*0.5D0
      AY=(TH31+TH3)*0.5D0+DELT3
      WK3=WK3+AX*(TH31-TH3)
      WS3=WS3+DABS(AY*(TH31-TH3))
      IF(ILINK.EQ.1)GOTO20
      TH6=TH(I,4)
      TH61=TH(I2,4)
      TH5=TH(I,5)
      TH51=TH(I2,5)
      AX=(W(I2,5)+W(I,5)-W(I2,2)-W(I,2))*0.5D0
      AY=(TH51+TH5-TH2-TH21)*0.5D0+DELT4
      WK4=WK4+AX*(TH51-TH21-TH5+TH2)
      WS4=WS4+DABS(AY*(TH51-TH5+TH2-TH21))
      AX=(W(I2,4)+W(I,4)-W(I2,5)-W(I,5))*0.5D0
      AY=(TH61+TH6-TH51-TH5)*0.5D0+DELT5
      WK5=WK5+AX*(TH61-TH51-TH6+TH5)
      WS5=WS5+DABS(AY*(TH61-TH6-TH51+TH5))
      AX=(W(I2,3)+W(I,3)-W(I2,4)-W(I,4))*0.5D0
      AY=(TH31+TH3-TH61-TH6)*0.5D0+DELT6
      WK6=WK6+AX*(TH31-TH61-TH3+TH6)
      WS6=WS6+DABS(AY*(TH31-TH3+TH6-TH61))
   20 VWRK=VWRK+DABS(TOR(I,1))
      VWRK=VWRK*RAD
      AK1=AK1*VWRK/WK1
      AK2=AK2*VWRK/WK2
      AK3=AK3*VWRK/WK3
      IF(ILINK.EQ.1)GOTO21
      AK4=AK4*VWRK/WK4
      AK5=AK5*VWRK/WK5
      AK6=AK6*VWRK/WK6
   21 IF(IOUT.EQ.1)GOTO31
      IF(IDAMP.EQ.1)GOTO32
      WRITE(6,210)AK1,AK2,AK3
      IF(ILINK.EQ.0)WRITE(6,211)AK4,AK5,AK5
   32 IF(STR2.NE.0.0D0)GOTO27
      WRITE(6,213)
      ISPRNG=1
```

```
      GOTO22
   27 WRITE(6,214)AS2,DELT2,AS3,DELT3
      IF(ILINK.EQ.0)WRITE(6,215)AS4,DELT4,AS5,DELT5,AS6,DELT6
   31 AS2=AS2*VWRK/WS2
      AS3=AS3*VWRK/WS3
      IF(ILINK.EQ.1)GOTO30
      AS4=AS4*VWRK/WS4
      AS5=AS5*VWRK/WS5
      AS6=AS6*VWRK/WS6
   30 IF(IOUT.EQ.1)GOTO22
      WRITE(6,216)AS2,AS3
      IF(ILINK.EQ.0)WRITE(6,217)AS4,AS5,AS6
   22 IF(INPT.EQ.0)GOTO4
      DO 5 I=1,INPT
      ISTOR(1+IJJ)=1
      ISTOR(2+IJJ)=2
      ISTOR(3+IJJ)=3
      ADM1=0.0D0
      ADM2=0.0D0
      ADM3=0.0D0
      ADM6=0.0D0
      ADM5=0.0D0
      ADR1=0.0D0
      ADR2=0.0D0
      ADR3=0.0D0
      ADR6=0.0D0
      ADR5=0.0D0
      ADG1=0.0D0
      ADG2=0.0D0
      ADG3=0.0D0
      ADG6=0.0D0
      ADG5=0.0D0
      ADI2=0.0D0
      ADI3=0.0D0
      ADI6=0.0D0
      ADI5=0.0D0
      ADK1=0.0D0
      ADK2=0.0D0
      ADK3=0.0D0
      ADK4=0.0D0
      ADK5=0.0D0
      ADK6=0.0D0
      ADS2=0.0D0
      ADS3=0.0D0
      ADS4=0.0D0
      ADS5=0.0D0
      ADS6=0.0D0
      ADELT2=0.0D0
      ADELT3=0.0D0
      ADELT4=0.0D0
      ADELT5=0.0D0
      ADELT6=0.0D0
      READ(5,ADLNK)
      DO 15 KSS=1,5
   15 KV(KSS+IJJ,1)=0
      IF(ADM1.EQ.0.0D0)KV(1+IJJ,1)=2
      IF(ADM2.EQ.0.0D0)KV(2+IJJ,1)=2
      IF(ADM3.EQ.0.0D0)KV(3+IJJ,1)=2
      AM(1+IJJ)=ADM1
      AM(2+IJJ)=ADM2

      AM(3+IJJ)=ADM3
      R(1+IJJ)=ADR1
      R(2+IJJ)=ADR2
      R(3+IJJ)=ADR3
      G(1+IJJ)=ADG1
      G(2+IJJ)=ADG2
      G(3+IJJ)=ADG3
      AI(2+IJJ)=ADI2
      AI(3+IJJ)=ADI3
      STR1=ADK1+ADK2+ADK3+ADK4+ADK5+ADK6
      STR2=ADS2+ADS3+ADS4+ADS5+ADS6
      AK(1+JXX)=ADK1*VWRK/WK1
      AK(2+JXX)=ADK2*VWRK/WK2
      AK(3+JXX)=ADK3*VWRK/WK3
      AS(2+JXX)=ADS2*VWRK/WS2
      AS(3+JXX)=ADS3*VWRK/WS3
      DELT(2+JXX)=ADELT2*RAD
      DELT(3+JXX)=ADELT3*RAD
      IF(ILINK.EQ.1)GOTO10
      ISTOR(4+IJJ)=5
      ISTOR(5+IJJ)=6
      IF(ADM6.EQ.0.0D0)KV(5+IJJ,1)=2
      IF(ADM5.EQ.0.0D0)KV(4+IJJ,1)=2
      AM(5+IJJ)=ADM6
      AM(4+IJJ)=ADM5
      R(5+IJJ)=ADR6
      R(4+IJJ)=ADR5
      G(5+IJJ)=ADG6
      G(4+IJJ)=ADG5
      AI(5+IJJ)=ADI6
      AI(4+IJJ)=ADI5
      AK(4+JXX)=ADK4*VWRK/WK4
      AK(5+JXX)=ADK5*VWRK/WK5
      AK(6+JXX)=ADK6*VWRK/WK6
      AS(4+JXX)=ADS4*VWRK/WS4
      AS(5+JXX)=ADS5*VWRK/WS5
      AS(6+JXX)=ADS6*VWRK/WS6
      DELT(4+JXX)=ADELT4*RAD
      DELT(5+JXX)=ADELT5*RAD
      DELT(6+JXX)=ADELT6*RAD
   10 IF(IOUT.EQ.1)GOTO9
      WRITE(6,208)I
      IF(KV(1+IJJ,1).NE.2)WRITE(6,209)ISTOR(1+IJJ),AM(1+IJJ),R(1+IJJ),
     1G(1+IJJ)
      DO 7 K=2,IK
    7 IF(KV(K+IJJ,1).NE.2)WRITE(6,209)ISTOR(K+IJJ),AM(K+IJJ),R(K+IJJ),
     1G(K+IJJ),AI(K+IJJ)
      IF(STR1.GT.0.0)GOTO28
      WRITE(6,212)
      GOTO34
   28 WRITE(6,206)ADK1,ADK2,ADK3
      IF(ILINK.EQ.0)WRITE(6,207)ADK4,ADK5,ADK6
      WRITE(6,210)AK(1+JXX),AK(2+JXX),AK(3+JXX)
      IF(ILINK.EQ.0)WRITE(6,211)AK(4+JXX),AK(5+JXX),AK(6+JXX)
   34 IF(STR2.GT.0.0D0)GOTO33
      WRITE(6,213)
      ISPRNG=1
      GOTO9
   33 WRITE(6,214)ADS2,ADELT2,ADS3,ADELT3
      IF(ILINK.EQ.0)WRITE(6,215)ADS4,ADELT4,ADS5,ADELT5,ADS6,ADELT6

      WRITE(6,216)AS(2+JXX),AS(3+JXX)
      IF(ILINK.EQ.0)WRITE(6,217)AS(4+JXX),AS(5+JXX),AS(6+JXX)
    9 DO 6 L=1,IK
      IF(KV(L+IJJ,1).EQ.2)GOTO6
      G(L+IJJ)=G(L+IJJ)*RAD
    6 CONTINUE
      IDMSP=IDAMP+ISPRNG
      JXX=JXX+JX
      IJJ=IJJ+IK
    5 CONTINUE
  208 FORMAT(//25X,43('*')/25X,'* ADDED INERTIA PARAMETERS FOR CURVE ',
     1I4,' *'/25X,43('*')/5X,'LINK',14X,' MASS',11X,' RADIAL CO-ORD',6X,
     1' ANGULAR OFFSET',7X,' M OF INERTIA')
    4 CONTINUE
      IF(ISYN.NE.0)GOTO11
      CALL PLTSRT
      STOP
   11 DO 24 I=1,JX
      AS(I+JXX)=0.0D0
      DELT(I+JXX)=0.0D0
   24 AK(I+JXX)=0.0D0
      CALL JSYNTH
      CALL PLTSRT
      STOP
      END
      FUNCTION DIFF(ILL,IN)
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION PR(5)
      COMMON/KIN/INV1,INV2,TH(542,5),S(542,5),C(542,5),W(542,5),W1,STH
     1,ALP(542,5),AJ(542,5),AL(10),SINE2,COSE2,SINE3,COSE3,SINB3,COSB3
      COMMON/STORE/AM(40),R(40),G(40),AI(40),AK(60),AS(60),DL(60)
      COMMON/APEX/V(1000),FF(30),MARBLE,KLOSEV,IVIOL
      COMMON/VARPAR/VM(250,5),VLIM(2,75,5),SENSIV,SENSIT
      COMMON/CONTRL/NPTS,ILINK,ILOCI,NPLOTS,IK,JX,IQ,IJJ,IVAR,INPT,ISW
     1,ISET,ISYN,IDEV,IKK,IBAL,IMOM,ITORQ,IDOT,MONITR,IOUT,JXX,IDMSP,IAN
      COMMON/FORCE/X(552,8),Y(552,8),DX(552,8),DY(552,8),TOR(552,8),GRAV
     1,FDEG(7),XDEG,YDEG,ADEG,TDEG,HOLD,AERR,ISRT
      COMMON/SYNTH/FP(7),ITP(7),KV(25,5)
      COMMON/CONST/PIE,PIE2,RAD,DEG,PY
      COMMON/RATIO/EXPAND,REDUCE,SHRINK,BOUNCE,SNAKE
     1,SMALL,FMIN,NITER,NPRINT,NCHECK,NPRT,NSCAN,ISCAN
      PLIM=1.0D0
      IF(ISRT.EQ.0)GOTO6
      CALL DYNMIC(-1,0)
      GOTO8
    6 K=IN
      L=IN
      AL4=AL(4)
      IAD=IDEV*(ILL-1)
      IZ=IAD
      IY=IKK+IAD
      IF(ILL.GT.ISET)IY=IKK
      IZSTR=IZ
      DO 1 I=1,IK
      IV=IJJ+I
      IF(KV(IV,1).EQ.2)GOTO1
      IY=IY+1
      IZ=IZ+1
      DO 2 J=1,5
      IF(KV(IV,J).EQ.1)GOTO2

      B=1.0D0
      ALIMI1=VLIM(1,IY,J)
      ALIMI2=VLIM(2,IY,J)
      CV=ALIMI2-ALIMI1
      D=CV*SENSIV
      ALIM1=ALIMI1+D
      ALIM2=ALIMI2-D
      K=K+1
      H=V(K)
      IF(J.GT.3)GOTO18
      IF(ALIMI1.NE.0.0D0)GOTO14
      IF(H.LT.0.0D0)GOTO12
      IF(H-ALIM2)11,11,14
   12 V(K)=0.0D0
      GOTO11
   18 IF(J.NE.4)GOTO16
      IF(H.LT.0.0D0)GOTO22
      IF(ALIMI1.EQ.0.0D0.AND.H.LT.ALIM2)GOTO11
      GOTO14
   22 IF(KV(IV,5).NE.0)GOTO23
      V(K)=-H
      V(K+1)=V(K+1)+PIE
      IF(-H-ALIM2)11,11,14
   23 V(K)=0.0D0
      GOTO11
   16 IF(H.GE.0.0D0)GOTO15
      H=H+PIE2
      GOTO16
   15 IF(H.LE.PIE2)GOTO17
      H=H-PIE2
      GOTO15
   17 V(K)=H
      IF(ALIMI1.EQ.0.0.AND.ALIMI2.GE.PY)GOTO11
   14 CX=H/ALIMI2
      A=1.0D-10
      IF(H.GE.ALIM1)GOTO10
      IFIZ=0
      A=(ALIM1-H)/CV
      IF(A.GT.1.5D2)A=1.5D2
      B=1.0D0+HOLD*(DEXP(A)-1.0D0)
      IF(MONITR.EQ.0)WRITE(6,700)VLIM(1,IY,J),IY,J,ALIM1,H,B
  700 FORMAT('LOW LIMIT',G12.4,' IS BROKEN FOR',2I2,
     1' AT',G12.4,' AS IT IS',G12.4,'. ERROR MULT. BY',G12.4)
      GOTO11
   10 IF(H.LE.ALIM2)GOTO11
      A=(H-ALIM2)/CV
      IF(A.GT.1.5D2)A=1.5D2
      B=1.0D0+HOLD*(DEXP(A)-1.0D0)
      IF(MONITR.EQ.0)WRITE(6,701)VLIM(2,IY,J),IY,J,ALIM2,H,B
  701 FORMAT('HIGH LIMIT',G12.4,' IS BROKEN FOR',2I2,
     1' AT',G12.4,' AS IT IS',G12.4,'. ERROR MULT. BY',G12.4)
   11 IF(B.GT.PLIM)PLIM=B
    2 CONTINUE
      DO 20 J=1,5
      PR(J)=VM(IZ,J)
      IF(KV(IV,J).NE.0)GOTO20
      L=L+1
      PR(J)=V(L)
   20 CONTINUE
      CALL DISC(PR,AM(IV),AI(IV),R(IV),G(IV))

    1 CONTINUE
      IDA=JXX
      CALL DYNMIC(IJJ,IDA)
    8 XBAL=0.0D0
      YBAL=0.0D0
      AMOM=0.0D0
      TORQ=0.0D0
      DO 3 J=1,NPTS
      IF(IBAL.EQ.1)GOTO4
      XS=DX(J,1)-DX(J,4)
      YS=DY(J,1)-DY(J,4)
      XBAL=XS*XS+XBAL
      YBAL=YS*YS+YBAL
    4 IF(IMOM.EQ.1)GOTO5
      ZS=TOR(J,2)
      AMOM=ZS*ZS+AMOM
    5 IF(ITORQ.EQ.1)GOTO3
      VS=TOR(J,1)
      TORQ=VS*VS+TORQ
    3 CONTINUE
      FBAL=0.0D0
      IF(IQ.EQ.0)GOTO7
      DO 19 I=1,IQ
      IM=ITP(I)
      AMX=X(1,IM)*X(1,IM)+Y(1,IM)*Y(1,IM)
      DO 21 J=1,NPTS
      XS=X(J,IM)
      YS=Y(J,IM)
      AA=XS*XS+YS*YS
      IF(AA.GT.AMX)AMX=AA
   21 CONTINUE
      FS=DSQRT(AMX)
      IF(FS.LE.FP(I))GOTO19
      IF(NSCAN.LT.NITER)WRITE(6,702)FS,FP(I),FDEG(I)
      FBAL=FBAL+FS*FDEG(I)
   19 CONTINUE
    7 PLIM=PLIM*AERR
      DIFF=PLIM*((XBAL*XDEG+YBAL*YDEG+AMOM*ADEG+TORQ*TDEG)/NPTS+FBAL)
      IF(NSCAN.LT.NITER)WRITE(6,703)XBAL,XDEG,YBAL,YDEG,AMOM,ADEG
     1,TORQ,TDEG,FBAL,PLIM
  702 FORMAT(/' MAX. BEARING FORCE=',G14.6,' AND EXCEEDS LIMIT',G14.6
     1,' THUS MULT. BY',G14.6,' AND PUT IN ERROR FUNC.')
  703 FORMAT(/' X-LEVEL=',G14.6,' IT IS MULT. BY',G14.6,' Y-LEVEL='
     1,G14.6,' IT IS MULT. BY',G14.6/' MOMENT LEVEL=',G14.6
     1,' IT IS MULT. BY',G14.6,' TORQUE LEVEL=',G14.6,' IT IS MULT. BY'
     1,G14.6/' BEARING FORCE FACTOR=',G14.6,' WHILST VARIABLE'
     1,' CONSTRAINT FACTOR=',G14.6)
      RETURN
      END
      SUBROUTINE JSYNTH
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION PR(5)
      COMMON/KIN/INV1,INV2,TH(542,5),S(542,5),C(542,5),W(542,5),W1,STH
     1,ALP(542,5),AJ(542,5),AL(10),SINE2,COSE2,SINE3,COSE3,SINB3,COSB3
      COMMON/FORCE/X(552,8),Y(552,8),DX(552,8),DY(552,8),TOR(552,8),GRAV
     1,FDEG(7),XDEG,YDEG,ADEG,TDEG,HOLD,AERR,ISRT
      COMMON/CONST/PIE,PIE2,RAD,DEG,PY
      COMMON/APEX/V(1000),FF(30),MARBLE,KLOSEV,IVIOL
      COMMON/RATIO/EXPAND,REDUCE,SHRINK,BOUNCE,SNAKE
     1,SMALL,FMIN,NITER,NPRINT,NCHECK,NPRT,NSCAN,ISCAN
```

```
      COMMON/VARPAR/VM(250,5),VLIM(2,75,5),SENSIV,SENSIT
      COMMON/SYNTH/FP(7),ITP(7),KV(25,5)
      COMMON/CONTRL/NPTS,ILINK,ILOCI,NPLOTS,IK,JX,IQ,IJJ,IVAR,INPT,ISW
     1,ISET,ISYN,IDEV,IKK,IBAL,INOM,ITORQ,IDOT,MONITR,IOUT,JXX,IDMSP,IAN
      COMMON/STORE/AM(40),R(40),G(40),AI(40),AK(60),AS(60),DL(60)
      NAMELIST /KSYNTH/ XDEG,YDEG,ADEG,TDEG,ISET,IRST,IRAND,NITER,NPRT
     1,NPRINT,NCHECK,NSCAN,HOLD,SENSIV,NPTS,IDEV1,IDEV2,IDEV3
     1,IDEV5,IDEV6,ITP,FDEG,FP,ILIM,IDOT,MONITR,IOUT,EXPAND
     1,SHRINK,REDUCE,B,SNAKE
     1      /START/V1,V2,V3,V4,V5,VLOW1,VHIGH1
     1,VLOW2,VHIGH2,VLOW3,VHIGH3,VLOW4,VHIGH4,VLOW5,VHIGH5
C IFR( ) CONTAINS THE PIN FORCE NUMBERS WHICH FORM THE ERROR FUNC.
C KV(I,J) IS ASSOCIATED WITH THE ADDED INERTIA
C I DEFINES BOTH THE DEVICE SET AND NUMBER
C J DEFINES THE DEVICE PARAMETER
C KV( )=0 PARAMETER SYNTHESIZED
C KV( )=1 PARAMETER FIXED
C KV( )=2 NO DEVICE
      XDEG=0.0D0
      YDEG=0.0D0
      ADEG=0.0D0
      TDEG=0.0D0
      ISET=1
      IRST=0
      NPT=NPTS
      NPTS=36
      IRAND=21
      ISTRD=IRAND
      NITER=300
      NSCAN=NITER+11
      NCHECK=5
      NPRINT=150
      NPRT=30
      HOLD=1.0D1
      SENSIV=0.05
      EXPAND=2.0
      SHRINK=0.5
      REDUCE=0.5
      KLOSEV=1
      IVIOL=1
      FMIN=1.0D-2
      SMALL=0.001
      BOUNCE=2.0
      SNAKE=0.1
      IBAL=0
      INOM=0
      ITORQ=0
      ILIM=1
      MONITR=1
      DO 1 IVV=1,ISYN
      IST=IKK
C TO FIND WHICH LINKS WILL HAVE DEVICES
      IDEV1=2
      IDEV2=2
      IDEV3=2
      IDEV5=2
      IDEV6=2
      IQ=0
      DO 25 IX=1,7
      FDEG(IX)=0.0D0
      FP(IX)=1.0D40
      ITP(IX)=0
   25 CONTINUE
      READ(5,KSYNTH)
      IAN=NPTS
      STEP=3.6D2*RAD/(NPTS-1)
      DO 58 I=1,NPTS
      IU=NPT+I
   58 TH(IU,1)=(I-1)*STEP
      CALL KINSOL(NPT)
      SENSIT=SENSIV*1.0D-2
      IF(NITER.NE.300.AND.NSCAN.EQ.311)NSCAN=NITER+10
      DO 27 IX=1,7
      IF(ITP(IX).NE.0)IQ=IQ+1
   27 CONTINUE
      KV(1+IJJ,1)=IDEV1
      KV(2+IJJ,1)=IDEV2
      KV(3+IJJ,1)=IDEV3
      IF(ILINK.EQ.1)GOTO24
      KV(4+IJJ,1)=IDEV5
      KV(5+IJJ,1)=IDEV6
   24 IVAR=0
C IVAR=NUMBER OF SYNTHESIS VARIABLES
C IQ=NUMBER OF PIN FORCES TO BE MONITORED
      IF(XDEG.EQ.0.0D0.AND.YDEG.EQ.0.0D0)IBAL=1
      IF(ADEG.EQ.0.0D0)INOM=1
      IF(TDEG.EQ.0.0D0)ITORQ=1
      IF(IBAL.EQ.0)IDOT=0
      IS=0
      CALL DYNMIC(-1,0)
      AMOM=0.0D0
      XT=0.0D0
      YT=0.0D0
      ATOR=0.0D0
      DO 31 I=1,NPTS
      IF(IBAL.EQ.1)GOTO9
      XA=DX(I,1)-DX(I,4)
      YA=DY(I,1)-DY(I,4)
      XT=XT+XA*XA
      YT=YT+YA*YA
    9 AA=TOR(I,2)
      BB=TOR(I,1)
      AMOM=AMOM+AA*AA
      ATOR=ATOR+BB*BB
   31 CONTINUE
      FADEG=0.0D0
      IF(IQ.EQ.0)GOTO5
      DO 48 I=1,IQ
   48 FADEG=FADEG+FDEG(I)
    5 ADD=(XDEG+YDEG+ADEG+TDEG+FADEG)/100.0D0
      PXDEG=XDEG/ADD
      PYDEG=YDEG/ADD
      PADEG=ADEG/ADD
      PATOR=TDEG/ADD
      IF(IBAL.EQ.1)GOTO49
      XDEG=XDEG*DABS(1.75D5/XT)
      YDEG=YDEG*DABS(1.75D5/YT)
   49 ADEG=ADEG*DABS(1.75D5/AMOM)
      TDEG=TDEG*DABS(1.75D5/ATOR)
      IF(IOUT.EQ.1)GOTO57
      IF(IQ.EQ.0)GOTO8
      DO 6 I=1,IQ
    6 FDEG(I)=FDEG(I)/ADD
    8 CONTINUE
      WRITE(6,201)PXDEG,PYDEG,PADEG,PATOR,FMIN
  201 FORMAT(//38X,18('*')/38X,'* SYNTHESIS DATA *'/38X,18('*')/
     1//37X,20('*')/37X,'* BALANCE REQUIRED *'/37X,20('*')/5X,
     1' FORCE X=',G14.6,'%  Y= ',G14.6,'%  MOMENT ABOUT CRANK= '
     1,G14.6,'%  AND DRIVING TORQUE=',G14.6,'%'/5X,
     1'REQUESTED REDUCTION IN UNDESIRABLE DYNAMIC CHARACTERISTICS'
     1,' WITH RESPECT TO UNBALANCED LINKAGE',G14.6,'%')
      IF(IQ.LE.0)GOTO55
      WRITE(6,213)
      WRITE(6,214)(ITP(I),FDEG(I),FP(I),I=1,IQ)
  213 FORMAT(/31X,33('*')/31X,'* CONSTRAINTS ON BEARING FORCES *'
     1/31X,33('*')/)
  214 FORMAT(24X,' PIN FORCE ',I2,' = ',G14.6,'% ABOVE A PEAK FORCE='
     1,G14.6)
   55 WRITE(6,202)NITER,NPRINT,NPRT,ISTRD,HOLD,SENSIV,
     1EXPAND,SHRINK,REDUCE,SMALL,BOUNCE,SNAKE
  202 FORMAT(//37X,20('*')/37X,'* SIMPLEX CONTROLS *'/37X,20('*')/25X,
     1' NITER NPRINT NPRT     IRAND',8X,'HOLD',16X,'SENSIV '
     1/24X,3I6,I10,2X,2G18.6//17X,'EXPAND',8X,'SHRINK',8X,'REDUCE',12X,
     1'SMALL',5X,'BOUNCE',9X,'SNAKE'/14X,6G14.6)
      WRITE(6,203)
  203 FORMAT(//24X,60('*')/24X,'* INITIAL VALUES OF SYNTHESIS VARIABLES
     1AND IMPOSED LIMITS *'/24X,60('*')/29X,'VARIABLE',7X,'VALUE',7X,
     1'LOWER LIMIT',2X,'HIGHER LIMIT')
  204 FORMAT(28X,2I4,5X,G14.6,8X,'FIXED')
  205 FORMAT(28X,2I4,5X,3G14.6)
      GOTO36
   57 ADD=1.0D0
   36 IF(IQ.EQ.0)GOTO7
      DO 54 I=1,IQ
   54 FDEG(I)=FDEG(I)*ADD*DABS(1.75D5/FP(I))
C VM(I,J)=ARRAY CONTAINING THE DEVICE PARAMETERS
C I=DEVICE NUMBER
C J=DEVICE PARAMETER
C DEVICE PARAMETERS STORE IN THIS ORDER
C 1=DISC DENSITY
C 2=DISC RADIUS
C 3=DISC THICKNESS
C 4=RADIAL OFFSET
C 5=ANGULAR OFFSET
    7 DO 4 I=1,IK
      IIJJ=I+IJJ
      IF(KV(IIJJ,1).EQ.2)GOTO4
      IS=IS+1
      ISKK=IS+IKK
      V1=7.822D3
      V2=0.0D0
      V3=0.0D0
      V4=0.0D0
      V5=0.0D0
      VLOW1=7.822D3
      VHIGH1=7.822D3
      VLOW2=0.0D0
      VHIGH2=0.0D0
      VLOW3=0.0D0
      VHIGH3=0.0D0
      VLOW4=0.0D0
      VHIGH4=0.0D0
      VLOW5=0.0D0
      VHIGH5=0.0D0
      READ(5,START)
      VM(IS,1)=V1
      VM(IS,2)=V2
      VM(IS,3)=V3
      VM(IS,4)=V4
      VM(IS,5)=V5
      VLIM(1,ISKK,1)=VLOW1
      VLIM(2,ISKK,1)=VHIGH1
      VLIM(1,ISKK,2)=VLOW2
      VLIM(2,ISKK,2)=VHIGH2
      VLIM(1,ISKK,3)=VLOW3
      VLIM(2,ISKK,3)=VHIGH3
      VLIM(1,ISKK,4)=VLOW4
      VLIM(2,ISKK,4)=VHIGH4
      VLIM(1,ISKK,5)=VLOW5
      VLIM(2,ISKK,5)=VHIGH5
      DO 3 J=1,5
      KV(IIJJ,J)=1
      IF(VLIM(1,ISKK,J).NE.VLIM(2,ISKK,J))GOTO2
      GOTO3
    2 IVAR=IVAR+1
      KV(IIJJ,J)=0
      IF(VM(IS,J).LE.VLIM(2,ISKK,J))GOTO3
      VM(IS,J)=RANDOM(IRAND,ISKK,J)
    3 CONTINUE
      VLIM(1,ISKK,5)=VLIM(1,ISKK,5)*RAD
      VLIM(2,ISKK,5)=VLIM(2,ISKK,5)*RAD
      VM(IS,5)=VM(IS,5)*RAD
    4 CONTINUE
      IDEV=IS
      JSET=ISET-1
      IF(JSET.LT.1)GOTO18
      IR=0
      DO 26 K=1,JSET
      IT=0
      DO 16 I=1,IK
      IIJJ=IJJ+I
      IF(KV(IIJJ,1).EQ.2)GOTO16
      IR=IR+1
      IT=IT+1
      IRKK=IR+IKK
      IF(ILIM.EQ.0)IRKK=IT+IKK
      VLOW1=VLIM(1,IRKK,1)
      VHIGH1=VLIM(2,IRKK,1)
      VLOW2=VLIM(1,IRKK,2)
      VHIGH2=VLIM(2,IRKK,2)
      VLOW3=VLIM(1,IRKK,3)
      VHIGH3=VLIM(2,IRKK,3)
      VLOW4=VLIM(1,IRKK,4)
      VHIGH4=VLIM(2,IRKK,4)
      VLOW5=VLIM(1,IRKK,5)*DEG
      VHIGH5=VLIM(2,IRKK,5)*DEG
      IS=IS+1
      ISDEV=IR
      IF(ILIM.EQ.0)ISDEV=IT
      V1=VM(ISDEV,1)
      V2=VM(ISDEV,2)
      V3=VM(ISDEV,3)
      V4=VM(ISDEV,4)
      V5=VM(ISDEV,5)*DEG
      ISKK=IKK+IS
      READ(5,START)
      VLIM(1,ISKK,1)=VLOW1
      VLIM(2,ISKK,1)=VHIGH1
      VLIM(1,ISKK,2)=VLOW2
      VLIM(2,ISKK,2)=VHIGH2
      VLIM(1,ISKK,3)=VLOW3
      VLIM(2,ISKK,3)=VHIGH3
      VLIM(1,ISKK,4)=VLOW4
      VLIM(2,ISKK,4)=VHIGH4
      VLIM(1,ISKK,5)=VLOW5*RAD
      VLIM(2,ISKK,5)=VHIGH5*RAD
      VM(IS,1)=V1
      VM(IS,2)=V2
      VM(IS,3)=V3
      VM(IS,4)=V4
      VM(IS,5)=V5*RAD
      KH=K+1
      DO 28 L=1,5
      IF(VLIM(1,ISKK,L).NE.VLIM(2,ISKK,L).AND.KV(IIJJ,L).NE.0)WRITE(6,20
     10)KH,L,I
   28 CONTINUE
  200 FORMAT(' *******WARNING******* '/' IN SET ',I3,
     1' PARAMETER ',I2,' ON DEVICE ',I2,' WAS ',
     1' SPECIFIED IN FIRST DATA SET AS FIXED'/
     1' YET LIMIT IN THIS SET DEFINES A VARIABLE ',
     1' PRESENT PARAMETER IS ACCEPTED BUT IS FIXED ')
      DO 17 J=1,5
      IF(KV(IIJJ,J).NE.0)GOTO17
      IF(VM(IS,J).GE.VLIM(1,ISKK,J).AND.VM(IS,J).LE.VLIM(2,ISKK,J))GOTO1
     17
      VM(IS,J)=RANDOM(IRAND,ISKK,J)
   17 CONTINUE
   16 CONTINUE
   26 CONTINUE
   18 CONTINUE
      MARBLE=IVAR
      IF(IRST.EQ.0)GOTO40
      IF(IRST.GT.IVAR+1)IRST=IVAR+1
      IO=IRST-IVAR-1+ISET
      IF(IO.GT.0)IS=IS-(IO*IK)
      DO 41 I=1,IRST
      READ(7,400)DMIN
      DO 42 L=1,IK
      IF(KV(L+IJJ,1).EQ.2)GOTO42
      IS=IS+1
      READ(7,400)(VM(IS,JJ),JJ=1,5)
   42 CONTINUE
   41 CONTINUE
   40 KZ=IVAR-JSET-IRST
  400 FORMAT(8G14.6)
      IF(KZ.LE.0)GOTO47
      DO 19 K=1,KZ
      IR=0
      DO 19 I=1,IK
      IF(KV(I+IJJ,1).EQ.2)GOTO19
```

```fortran
      IS=IS+1
      IR=IR+1
      DO 20 J=1,5
      IF(KV(I+IJJ,J).NE.1)GOTO21
      VM(IS,J)=VM(IR,J)
      GOTO20
   21 VM(IS,J)=RANDCM(IRAND,IR+IST,J)
   20 CONTINUE
   19 CONTINUE
   47 IA=IVAR+1
      L=0
      DO 50 I=1,4
      DO 51 K=1,IK
      IF(KV(IJJ+K,1).EQ.2)GOTO51
      L=L+1
      IS=IS+1
      DO 52 J=1,5
      VM(IS,J)=VM(L,J)
   52 CONTINUE
   51 CONTINUE
   50 CONTINUE
      IF(IOUT.NE.0)GOTO29
      K=0
      IM=0
      IL=ISET+IRST
      DO 32 L=1,IL
      IF(L.GT.ISET)K=0
      DO 44 I=1,IK
      IF(KV(I+IJJ,1).EQ.2)GOTO44
      IM=IM+1
      K=K+1
      DO 43 M=1,5
      IF(KV(I+IJJ,M)*L-1)56,33,43
   56 WRITE(6,205)I,M,VM(IM,M),VLIM(1,K+IKK,M),VLIM(2,K+IKK,M)
      GOTO43
   33 WRITE(6,204)I,M,VM(IM,M)
   43 CONTINUE
   44 CONTINUE
   32 CONTINUE
   29 CONTINUE
      IW=0
      IC=0
      DO 11 I=1,IA
      IR=0
      DO 11 J=1,IK
      IR=IR+1
      IF(KV(IR+IJJ,1).EQ.2)GOTO11
      IW=IW+1
      DO 10 L=1,5
      IF(KV(IR+IJJ,L).EQ.1)GOTO10
      IC=IC+1
      V(IC)=VM(IW,L)
   10 CONTINUE
   11 CONTINUE
      ISCAN=IW
C IBAL 0= FRAME FORCE BALANCING
C        1=SUPRESSED
C IMOM 0= MOMENT ABOUT CRANK PIVOT BALANCING
C        1=SUPRESSED
C ITORQ 0= TORQUE BALANCE SOUGHT
C        1TORQUE BALANCE SUPRESSED
C ITP( )=REFERENCE NUMBERS OF PIN FORCES TO BE RE-SHAPED
      IOP=1
      IPP=0
      KSTR=MONITR
      MONITR=1
      ISRT=1
      AERR=1.0D0
      AERR=1.0D2/DIFF(I,J)
      ISRT=0
      PINTL=DIFF(IOP,IPP)
      FF(1)=PINTL
      MIN=1
      K=0
      IVK=IVAR+1
      DO 22 I=2,IVK
      IDA=I
      K=K+IVAR
      FF(I)=DIFF(IDA,K)
      IF(FF(I).GE.PINTL)GOTO22
      MIN=I
      PINTL=FF(I)
   22 CONTINUE
      P=PINTL
      MONITR=KSTR
      IF(NSCAN.GT.NITER)GOTO46
      WRITE(6,210)
  210 FORMAT(20X,' VM( ) MATRIX ')
      DO 45 I=1,ISCAN
   45 WRITE(6,211)(VM(I,J),J=1,5)
      WRITE(6,212)
  212 FORMAT(20X,' V( ) MATRIX ')
      NI=IVAR*(IVAR+1)
      WRITE(6,211)(V(I),I=1,NI)
  211 FORMAT(5G14.6)
   46 CALL SIMPLX(P,MIN)
      XPER=(PINTL-P)*1.0D2/PINTL
      WRITE(6,215)XPER
  215 FORMAT(1H1,' PERCENTAGE CHANGE IN ERROR FUNCTION='
     1,G14.6/)
      WRITE(6,206)
  206 FORMAT(//17X,46('*')/17X,'* PARAMETERS OF SYNTHESIZED COUNTER-INER
     1TIAS *'/17X,46('*')//8X,'LINK',3X,'DENSITY',6X,'RADIUS'
     1,8X,'THICKNESS',4X,'RAD. OFFSET',4X,'ANG. OFFSET',4X,'MASS'
     1,9X,'MOM. OF INERTIA')
      N=IDEV*(MIN-1)
      DO 37 I=1,IK
      ISKK=IJJ+I
      IZERO=0
      L=I
      IF(KV(ISKK,1).EQ.2)GOTO37
      IF(I.EQ.4)L=5
      IF(I.EQ.5)L=6
      N=N+1
      DO 38 J=1,5
      IF(VM(N,J).EQ.0.0D0.AND.J.NE.4)IZERO=1
   38 CONTINUE
      IF(IZERO.EQ.1)GOTO37
      DO 59 J=1,5
   59 PR(J)=VM(N,J)
      CALL DISC(PR,AM(ISKK),AI(ISKK),R(ISKK),G(ISKK))
      V5=VM(N,5)*DEG
      IF(I.EQ.1)WRITE(6,207)L,VM(N,1),VM(N,2),VM(N,3),VM(N,4),V5
     1,AM(ISKK)
      IF(I.NE.1)WRITE(6,207)L,VM(N,1),VM(N,2),VM(N,3),VM(N,4),V5
     1,AM(ISKK),AI(ISKK)
   37 CONTINUE
  207 FORMAT(6X,I5,7G14.6)
      IJJ=IJJ+IK
      IKK=IKK+(IDEV*ISET)
      JXX=JXX+JX
    1 CONTINUE
      NPTS=NPT
      IAN=0
      RETURN
      END
       SUBROUTINE DISC(PR,AM,AI,R,G)
       IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION PR(5)
      COMMON/CONST/PIE,PIE2,RAD,DEG,PY
      R=PR(2)
      RR2=R*R
      AM=PR(1)*PIE*RR2*PR(3)
      AI=AM*RR2*5.0D-1
      R=PR(4)
      G=PR(5)
      RETURN
      END
      FUNCTION RANDCM(IRAND,I,J)
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON/VARPAR/VM(250,5),VLIM(2,75,5),SENSIV,SENSIT
      ALIM1=VLIM(1,I,J)
      ALIM2=VLIM(2,I,J)
      DIFF=ALIM2-ALIM1
      A=DIFF*SENSIV
      ALIM1=ALIM1+A
      ALIM2=ALIM2-A
      IY=IRAND*65539
      IF(IY)270,280,280
  270 IY=IY+2147483647+1
  280 RAND=IY
      RAND=RAND*0.4656613D-9
      IRAND=IY
      RANDOM=ALIM1+(ALIM2-ALIM1)*RAND
      RETURN
      END
      SUBROUTINE PLTSRT
      IMPLICIT REAL*8(A-H,O-Z)
      REAL CC(400),DD(400)
      DIMENSION ITITLE(80),IJ(10),KX(10)
      COMMON/KIN/INV1,INV2,TH(542,5),S(542,5),C(542,5),W(542,5),W1,STH
     1,ALP(542,5),AJ(542,5),AL(10),SINE2,COSE2,SINE3,COSE3,SINB3,COSB3
      COMMON/CONTRL/NPTS,ILINK,ILOCI,NPLOTS,IK,JX,IQ,IJJ,IVAR,INPT,ISW
     1,ISET,ISYN,IDEV,IKK,IBAL,IMOM,ITORQ,IDOT,MONITR,IOUT,JXX,IDMSP,IAN
      COMMON/FORCE/XT(552,8),YT(552,8),DX(552,8),DY(552,8),TOR(552,8)
     1,GRAV,FDEG(7),XDEG,YDEG,ADEG,TDEG,HOLD,AERR,ISRT
      COMMON/STORE2/B(3000),A(3000)
      COMMON/STORE/AM(40),R(40),G(40),AI(40),AK(60),AS(60),DELT(60)
      COMMON/CONST/PIE,PIE2,RAD,DEG,PY
      NAMELIST/PLTCTR/NPLOTS,IPOLAR,IUNBAL,ISELCT,IERR,LABINC,IPLOT
     1,LINTYP,XPAGE,YPAGE,NSCALE,DEGINC,ANGST,IDETC,NCYCL,STH,IRIG
     1,IDOT,NPTS
      ANGST=0.0D0
      DEGINC=1.0D0
      NSCALE=1
      IUNBAL=0
      ISELCT=0
      NPLOTS=INPT+1
      LABINC=90
      XPAGE=10.0D0
      YPAGE=10.0D0
      LINTYP=45
      IDETC=0
      IKIN=0
      IPTS=361
      DO 24 IVV=1,ILOCI
      IDOT=1
      IPOLAR=0
      IERR=1
      IRIG=1
      NCYCL=1
      IPLOT=1
      IAN=0
      NPTS=361
      READ(5,100)(ITITLE(I),I=1,18)
      READ(5,PLTCTR)
  100 FORMAT(18A4)
      IF(IPTS.EQ.NPTS.AND.IVV.GT.1)GOTO20
      DANG=3.6D2*RAD/(NPTS-1)
      DO 21 I=1,NPTS
   21 TH(I,1)=(I-1)*DANG
      CALL KINSCL(0)
      IPTS=NPTS
   20 CONTINUE
C
C*****************************************************
C
C NPLOTS=NUMBER OF CURVES TO BE OVERPLOTTED
C IPOLAR=POLAR OR VERSUS CRANK ANGLE(IPOLAR=0 X-Y PLOT)
C IUNBAL=WHETHER UNBALANCE LINKAGE IS PLOTTED.(0=UNBALANCED PLOTTED)
C SETTING ISELCT TO THE FOLLOWING NUMBERS GIVES:
C 0=TOTAL FORCE ON THE FRAME
C 1 TO 7 =PIN FORES
C 8=MOMENT ABOUT CRANK PIVOT
C 9= INPUT TORQUE
C IERR=NUMBER OF REFERENCE CURVE FOR EVALUATING PERCENTAGE CHANGES
C      0= ERROR CALCULATION SUPRESSED
C LABINC= LABELLING INCREMENT (IE 180 GIVES LABEL EVERY 180 DEGREES)
C LINTYP= WHETHER LINES CONNECTED OR NOT
C IPLOT THIS LOCI PLOTTED (0=YES)
C XPAGE= LENGTH OF PLOTTING PAPER REQUIRED
C       16.8=XPAGE IS USUALLY THE LARGEST SIZE FOR COMPUTEX
C YPAGE=HEIGHT OF PLOTTING PAER REQUIRED
C       12.6=YPAGE IS USUALLY THE MAXIMUM PERMISSIBLE PLOTTING HEIGHT
C DEFAULT FOR BOTH IS 10.0 WHICH IS SUITABLE FOR A4 PLOTS.
C*****************************************************
C
      ICOMP=0
      IF(ISELCT.LT.8)IFORCE=ISELCT
      IF(ISELCT.EQ.0)IFORCE=8
      IF(ISELCT.GT.7)IPOLAR=1
      IF(ISELCT.GT.9)ITORQ=ISELCT-9
      IF(ISELCT.EQ.8)ITORQ=2
      IF(ISELCT.EQ.9)ITORQ=1
      IF(ISELCT.GT.7)IDETC=0
      ICHK=INPT+ISYN
      IF(IUNBAL.EQ.0)GOTO18
      IJ(1)=0
      KX(1)=0
      KK=2
      GOTO16
   18 IJ(1)=-1
      ICHK=ICHK+1
      IJ(2)=0
      KX(1)=0
      KX(2)=0
      KK=3
   16 IF(ICHK.GE.NPLOTS)GOTO29
      NPLOTS=ICHK
      WRITE(6,200)ICHK
  200 FORMAT(1H1,'*****ERROR**** NUMBER OF PLOTS REQUESTED'
     1,' GREATER THAN SPECIFIED INPUT. NUMBER OF PLOTS RESET'/
     1' TO INCLUDE INPUT SET AND UNBALANCED LINKAGE IF REQUESTED'
     1,' I.E.',I4)
   29 IF(NPLOTS.LT.KK)GOTO17
      DO 5 I=KK,NPLOTS
      K=I-1
      KX(I)=KX(K)+JX
    5 IJ(I)=IJ(K)+IK
   17 DO 3 I=1,NPLOTS
      IPL=(I-1)*NPTS
      CALL DYNMIC(IJ(I),KX(I))
      IF(ISELCT.EQ.0)CALL ADD(NPTS,IDOT)
      XAD=0.0D0
      YAD=0.0D0
      IF(IRIG.EQ.1)GOTO6
      ISW=0
      CALL KINSOL(361)
      IAN=361
      CALL DYNMIC(IJ(I),KX(I))
      IAN=0
      IF(ISELCT.NE.0.OR.IDOT.EQ.0)GOTO11
      XAD=XT(362,4)-XT(362,1)
      YAD=YT(362,4)-YT(362,1)
   11 ISW=0
    6 DO 4 J=1,NPTS
      IH=IPL+J
      IF(ISELCT.GT.7)GOTO14
      IF(IDOT.NE.0)GOTO10
      B(IH)=DX(J,IFORCE)
      A(IH)=DY(J,IFORCE)
      GOTO4
   10 B(IH)=XT(J,IFORCE)-XAD
      A(IH)=YT(J,IFORCE)-YAD
      GOTO4
   14 A(IH)=TOR(J,ITORQ)
    4 CONTINUE
    3 CONTINUE
      IP=NPTS*NPLOTS
      IF(IPOLAR.NE.0)GOTO1
```

```
C POLAR PLOT OF FORCES SELECTED
      IF(IERR.GT.0) CALL PERERR(ITITLE,IPOLAR,IERR)
      NCYCL=1
      XMAX=B(1)
      XMIN=XMAX
      DO 8 I=2,IP
      X=B(I)
      IF(X.GT.XMAX) XMAX=X
    8 IF(X.LT.XMIN) XMIN=X
      GOTO7
C
C***************************************************************
C
C SETTING IDETC TO THE FOLLOWING NUMBERS GIVES:
C 1=MAXIMUM FORCE MAGNITUDE PLOTTED
C 2=X-FORCE MAGNITUDE PLOTTED
C 3=Y-FORCE MAGNITUDE PLOTTED
C NCYCL= NUMBER OF CYCLES OVER WHICH VARIABLE IS TO BE PLOTTED
C
C***************************************************************
C
    1 NSCALE=2
      XMIN=0.0D0
      XMAX=NPTS*NCYCL
      IF(IDETC.EQ.1)CALL MAG
      IF(IDETC.NE.2)GOTO19
      DO 27 I=1,IP
   27 A(I)=B(I)
   19 DO 22 J=1,NPLOTS
      K=(J-1)*NPTS
      DO 22 I=1,NPTS
      IM=I+K
   22 B(IM)=(I-1)*NCYCL
      IF(IERR.GT.0)CALL PERERR(ITITLE,IPOLAR,IERR)
    7 YMAX=A(1)
      IF(IPLOT.EQ.1)GOTO24
      YMIN=YMAX
      DO 9 I=2,IP
      Y=A(I)
      IF(Y.GT.YMAX) YMAX=Y
    9 IF(Y.LT.YMIN) YMIN=Y
      NSUPER=1
      IF(NPLOTS.EQ.1)NSUPER=0
      DO 12 I=1,NPLOTS
      IF(I.GT.1.AND.I.EQ.NPLOTS)NSUPER=3
      K=(I-1)*NPTS
      LSYMP=I-1
      IF(LSYMP.GT.1)LSYMP=LYSMP+5
      IC=1
      IZ=NPTS-NCYCL+1
      IF(IPLOT.EQ.2)GOTO2
      DO 13 J=1,NPTS
      CC(J)=A(K+IC)
      DD(J)=B(K+J)
      IF(IC.GE.IZ)IC=IC-NPTS
   13 IC=IC+NCYCL
      CALL XYPLOT(ITITLE,ICOMP,NPTS,NSCALE,LABINC
     1,LINTYP,LSYMP,ANGST,DEGINC,XPAGE,YPAGE,DD,CC,
     1XMAX,XMIN,YMAX,YMIN,NSUPER)
      GOTO12

    2 WRITE(6,102)I
      DGANG=3.6D2/(NPTS-1)
      DO 15 J=1,NPTS
      EGANG=(J-1)*DGANG
      AMAG=DSQRT(A(K+IC)*A(K+IC)+B(K+J)*B(K+J))
      WRITE(6,101)EGANG,A(K+IC),B(K+J),AMAG
      IF(IC.GT.IZ)IC=IC-NPTS
   15 IC=IC+NCYCL
  101 FORMAT(1H ,24X,4(G14.6,6X))
  102 FORMAT(1H ,/63X,'CURVE',I3//
     126X,'CRANK ANGLE',10X,'X COMPONENT',9X,'Y COMPONENT',
     17X,'RADIAL OFFSET')
   12 NSUPER=2
   24 CONTINUE
      RETURN
      END
      SUBROUTINE MAG
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON/CONTRL/NPTS,ILINK,ILOCI,NPLOTS,IK,JX,IQ,IJJ,IVAR,INPT,ISW
     1,ISET,ISYN,IDEV,IKK,IBAL,IMOM,ITORQ,IDOT,MONITR,IOUT,JXX,IDMSP,IAN
      COMMON/STORE2/X(3000),Y(3000)
      DO 1 J=1,NPLOTS
      DO 1 I=1,NPTS
      IJ=I+(J-1)*NPTS
      A=X(IJ)
      B=Y(IJ)
    1 Y(IJ)=DSQRT(A*A+B*B)
      RETURN
      END
      SUBROUTINE ADD(NPTS,IDOT)
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON/FORCE/X(552,8),Y(552,8),DX(552,8),DY(552,8),TOR(552,8)
     1,GRAV,FS(7),XDEG,YDEG,ADEG,TDEG,HOLD,AERR,ISRT
      DO 1 I=1,NPTS
      IF(IDOT.EQ.0)GOTO2
      X(I,8)=X(I,4)-X(I,1)
      Y(I,8)=Y(I,4)-Y(I,1)
      GOTO1
    2 DX(I,8)=DX(I,4)-DX(I,1)
      DY(I,8)=DY(I,4)-DY(I,1)
    1 CONTINUE
      RETURN
      END
      SUBROUTINE PERERR(ITITLE,IPOLAR,IERR)
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON/CONTRL/NPTS,ILINK,ILOCI,NPLOTS,IK,JX,IQ,IJJ,IVAR,INPT,ISW
     1,ISET,ISYN,IDEV,IKK,IBAL,IMOM,ITORQ,IDOT,MONITR,IOUT,JXX,IDMSP,IAN
      COMMON/STORE2/XS(3000),YS(3000)
      COMMON/CONST/PIE,PIE2,RAD,DEG,PY
      DIMENSION AMX(10),AMEAN(2,10),RMS(2,10),ITITLE(80),ANG(10)
      WRITE(6,100)(ITITLE(I),I=1,18)
      N=2
      NT=NPTS-1
      IF(IPOLAR.EQ.1)N=1
      DO 1 J=1,NPLOTS
      F=0.0D0
      D=0.0D0
      E=0.0D0
      G=0.0D0
      K=(J-1)*NPTS

      YMAX=YS(K+1)
      YMX=DABS(YMAX)
      ZMAX=YMAX*YMAX+XS(K+1)*XS(K+1)
      DO 2 I=1,NT
      B=XS(K+I)
      C=YS(K+I)
      G=C*C+G
      E=C+E
      IF(IPOLAR.NE.0)GOTO3
      D=D+B
      F=B*B+F
      Z=B*B+C*C
      IF(ZMAX.GE.Z)GOTO2
      ZMAX=Z
      THETA=DATAN2(C,B)
      GOTO2
    3 H=DABS(C)
      IF(YMX.GE.H)GOTO2
      YMX=H
      YMAX=C
    2 CONTINUE
      IF(IPOLAR.NE.0)GOTO4
      AMEAN(2,J)=D/NT
      RMS(2,J)=DSQRT(F)/NT
      AMX(J)=DSQRT(ZMAX)
      ANG(J)=THETA*DEG
      GOTO5
    4 AMX(J)=YMAX
      ANG(J)=THETA
    5 AMEAN(1,J)=E/NT
      RMS(1,J)=DSQRT(G)/NT
    1 CONTINUE
      IF(IERR.LE.NPLOTS)GOTO6
      WRITE(6,104)IERR,NPLOTS
      IERR=1
    6 DO 8 I=1,NPLOTS
      IF(IERR.EQ.I)GOTO8
      WRITE(6,101)I,IERR
      B=DABS(AMX(IERR))
      CMX=(DABS(AMX(I))-B)*1.0D2/B
      DO 9 J=1,N
      B=DABS(AMEAN(J,IERR))
      CMEAN=(DABS(AMEAN(J,I))-B)*1.0D2/B
      CRMS=(RMS(J,I)-RMS(J,IERR))*1.0D2/RMS(J,IERR)
      IF(J.EQ.2)GOTO10
      IF(IPOLAR.NE.0)GOTO11
      WRITE(6,105)AMX(IERR),AMX(I),CMX,ANG(IERR),ANG(I)
      WRITE(6,102)
      GOTO12
   11 WRITE(6,106)AMX(IERR),AMX(I),CMX,ANG(IERR),ANG(I)
   12 WRITE(6,107)AMEAN(J,IERR),AMEAN(J,I),CMEAN,RMS(J,IERR),RMS(J,I)
     1,CRMS
      GOTO9
   10 WRITE(6,103)
      GOTO12
    9 CONTINUE
    8 CONTINUE
  100 FORMAT(//20X,74('*')/20X,'*',18A4,'*'/20X,74('*')/)
  101 FORMAT(25X//' PERCENTAGE CHANGES OF CURVE ',I1,
     1' WITH RESPECT TO CURVE ',I1/)

  102 FORMAT('** FOR Y AXIS **')
  103 FORMAT('** FOR X AXIS **')
  104 FORMAT(' *** ERROR- IERR=',I1,' WHICH IS GREATER THAN',
     1' NUMBER OF PLOTS,IE ',I1/' IERR RESET TO ONE ***')
  105 FORMAT(1X,' MAX. MAGNITUDE WAS ',G14.6,' BUT IS NOW'
     1,G14.6,' THIS CONSTITUTES A RISE OF',G14.6,'%'/
     1'  THE ANGULAR DIRECTION HAS CHANGED FROM',G14.6,
     1' TO ',G14.6/)
  106 FORMAT(1X,' MAX. MAGNITUDE WAS',G14.6,' BUT IS NOW',G14.6
     1,' THIS CONSTITUTES A RISE OF',G14.6,'%'/
     1' THE CYCLIC POSITION OF INPUT CRANK CHANGED FROM ',G14.6,
     1' TO ',G14.6/)
  107 FORMAT(1X,' MEAN MAGNITUDE WAS',G14.6,' BUT IS NOW',G14.6
     1,' THIS CONSTITUTES A RISE OF',G14.6,'%'/
     1'    RMS MAGNITUDE WAS',G14.6,' BUT IS NOW',G14.6
     1,' THIS CONSTITUTES A RISE OF',G14.6,'%'/)
      RETURN
      END
      SUBROUTINE SIMPLX(PMIN,MINI)
C
C     SIMPLEX MINIMIZATION ROUTINE
C     ACTUAL NO. OF OPTIMIZATION VARIABLES = N
C     DIMENSION OF 'V' IS V(N*(N+4))
C     DIMENSION OF 'F' IS F(N+3)
C     CENTROID VALUES ARE V(N*(N+3)+1) TO V(N*(N+4))
C
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON/CONTRL/NPTS,ILINK,ILOCI,NPLOTS,IK,JX,IQ,IJJ,IVAR,INPT,ISW
     1,ISET,ISYN,IDEV,IKK,IBAL,IMOM,ITORQ,IDOT,MONITR,IOUT,JXX,IDMSP,IAN
      COMMON/SYNTH/FP(7),ITP(7),KV(25,5)
      COMMON /APEX/ V(1000),F(30),MARBLE,KLOSEV,IVIOL
      COMMON /RATIO/ EXPAND,REDUCE,SHRINK,BOUNCE,SNAKE
     1,SMALL,FMIN,NITER,NPRINT,NCHECK,NPRT,NSCAN,ISCAN
      COMMON/VARPAR/VM(250,5),VLIM(2,75,5),SENSIV,SENSIT
      DATA ITER,MAX,MM/0,2*1/,LABP,LABF/2H P,2H F/
  101 FORMAT(' FUNCTION VALUES,F1,..F(N+1),OF THE INITIAL SIMPLEX ARE:')
  102 FORMAT(' ITER.=',I4,',MAX.ERR.=',G15.7,',MIN.ERR.=',G15.7)
  104 FORMAT(' MAX. ERROR VALUE IN CURRENT SIMPLEX',/,G15.7)
  105 FORMAT(' MIN. ERROR VALUE IN CURRENT SIMPLEX',/,G15.7)
  106 FORMAT('0NITER LIMIT REACHED,CURRENT VALUES WRITTEN ON I07'/
     1' FINAL MIN. ERROR =',G14.6)
  107 FORMAT(8G14.6)
  112 FORMAT('0ITERATIONS =',I10)
  115 FORMAT(1X,5G15.7)
  116 FORMAT(' ***STUCK WITH SMALL CRITERION***')
  117 FORMAT(' ***CONVERGENCE SUCCESS-FMIN CRITERION***'/)
  119 FORMAT(' THE PARAMETERS ASSOCIATED WITH THE CURRENT MAX. ERROR')
  120 FORMAT(' THE PARAMETERS ASSOCIATED WITH THE CURRENT MIN. ERROR')
  121 FORMAT(' FINAL VALUES OF THE OPTIMIZATION VARIABLES'
     1,' CORRESPONDING TO MIN. ERROR')
  122 FORMAT(' MINIMUM ERROR',/G15.7)
C
C     INITIAL SETS ARE GIVEN
C
      MIN=MINI
      AMIN=PMIN
      N=MARBLE
      N1=N+1
      N2=N+2
      N3=N+3

      N4=N+4
      W=EXPAND
      X=BOUNCE
      Y=REDUCE
      Z=SHRINK
      NSNAKE=0
      NSHRNK=0
      NN = N*N1
      NNN=N+NN
      NNNN=N+NNN
      FN=N
      FN1=N1
      RX=1.0D0/X
      RHN=X/FN
      FACT=FN*RX-FN1
      WMO=W-1.0D0
      XMO=X-1.0D0
      OMY=1.0D0-Y
      OMZ=1.0D0-Z
      WRITE(6,101)
      WRITE(6,115)(F(I),I=1,N1)
      KLOSEV=1
      ASSIGN 2 TO KRANK
      GO TO 24
    2 ASSIGN 28 TO KRANK
C
C     BEGIN NEW ITERATION
C
    3 CONTINUE
C
C     SELECT NEW VERTEX V*=V(N2) & CALCULATE F*=F(N2)
C     V* IS THE REFLECTION OF V(MAX) ABOUT THE MEAN OF THE OTHER V'S
C
      DO 65 I=1,N
      IMN=I-N
      IMON=I+MAXON
      VINN=FACT*V(IMON)
      DO 55 J=1,N1
      JMON=IMN+J*N
   55 VINN=VINN+V(JMON)
      INN=I+NN
      V(INN)=VINN*RHN
      INNNN=I+NNNN
   65 V(INNNN)=(V(INN)+XMO*V(IMON))*RX
      F(N2)=DIFF(N2,NN)
      IF(IVIOL.NE.6)GO TO 9
C
C     CHECK WHETHER CENTROID IS FEASIBLE
C
      NGOOD=NNNN
      F(N4)=DIFF(N4,NNNN)
      IF(IVIOL.EQ.6)NGOOD=MINON
    7 DO 8 I=1,N
      INN=I+NN
      IGOOD=I+NGOOD
    8 V(INN)=0.25D0*(3.D0*V(INN)+V(IGOOD))
      F(N2)=DIFF(N2,NN)
      IF(IVIOL.EQ.6)GO TO 7
C
C     CHECK IF F* .LT. F(MIN)
```

```
C
    9 IF(F(N2).GE.F(MIN))GOTO148
C
C     DIRECTION IS GOOD SO TRY DOUBLE MOVE TO V**=W.V*-(W-1).V(MAX)
C
      MOVE=0
   95 DO 10 I=1,N
      INN=I+NN
      INNN=I+NNN
      IMAXON=I+MAXON
   10 V(INNN)=W*V(INN)-WMO*V(IMAXON)
      F(N3)=DIFF(N3,NNN)
C
C     CHECK IF DOUBLE MOVE IS IMPROVEMENT ON ORIGINAL MOVE
C
      IF(IVIOL.EQ.6)GO TO 12
      IF(F(N3)-F(N2)) 11,12,12
C
C     DIRECTION IS GOOD SO TRY QUADRUPLE MOVE TO V*=W.V**-(W-1).V(MAX)
C
   11 DO 6 I=1,N
      INNN=I+NNN
      INN=I+NN
      IMAXON=I+MAXON
    6 V(INN)=W*V(INNN)-WMO*V(IMAXON)
      F(N2)=DIFF(N2,NN)
      MOVE=1
      IF(IVIOL.EQ.6)GO TO 114
      IF(F(N2)-F(N3))95,114,114
  114 NEW=N3
      NCENT=NN
      NEXP=NNN
      GO TO 123
   12 NEW=N2
      IF(MOVE.EQ.0)GO TO 138
      NCENT=NNN
      NEXP=NN
C
C     DETERMINE VERTEX MOVEMENT
C
  123 IF(F(NEW).GT.SNAKE*F(MIN))GO TO 138
      DO 125 I=1,N
      INN=I+NEXP
      INNN=I+NCENT
      IMAXON=I+MAXON
  125 V(INNN)=V(INN)-V(IMAXON)
C
C     TRANSLATE COMPLETE SIMPLEX
C
      DO 135 J=1,N1
      JMN=(J-1)*N
      DO 13 I=1,N
      INN=I+NCENT
      JNN=I+JMN
   13 V(JNN)=V(JNN)+V(INN)
  132 F(J)=DIFF(J,JMN)
      IF(IVIOL.NE.6)GO TO 135
      DO 133 I=1,N
      INN=I+NEXP
      JNN=I+JMN
  133 V(JNN)=0.25D0*(3.D0*V(JNN)+V(INN))
      GO TO 132
  135 CONTINUE
      NSNAKE=NSNAKE+1
C
C     RESET V(MAX) TO NEW VALUES
C
  138 NEWON=(NEW-1)*N
      DO 14 I=1,N
      IMAXON=I+MAXON
      INEWON=I+NEWON
   14 V(IMAXON)=V(INEWON)
      F(MAX) = F(NEW)
      GO TO 24
C
C     ORIGINAL MOVE NO GOOD. CHECK IF REFLECTED POINT
C     WILL HAVE THE LARGEST FUNCTION VALUE
C
  148 IF(F(N2)-F(MM)) 15,16,16
   15 NEW=N2
      GO TO 138
C
C     REFLECTED POINT WILL HAVE LARGEST FUNCTION VALUE
C     CHECK IF CENTROID IS FEASIBLE
C
   16 NBAD=NN
      IF(F(N2).GT.F(MAX))NBAD=MAXON
      NGOOD=NNNN
      F(N4)=DIFF(N4,NNNN)
      IF(IVIOL.EQ.6)NGOOD=MINON
   17 DO 18 I=1,N
      INNN=I+NNN
      IBAD=I+NBAD
      IGOOD=I+NGOOD
   18 V(INNN)=OMY*V(IGOOD)+Y*V(IBAD)
      F(N3)=DIFF(N3,NNN)
      IF(IVIOL.NE.6)GO TO 186
      NBAD=NNN
      GO TO 17
C
C     IF V** STILL HAS GREATEST FUNCTION VALUE,SHRINK SIMPLEX
C
  186 IF(F(N3)-F(MAX)) 19,20,20
   19 NEW=N3
      GO TO 138
C
C     SHRINK SIMPLEX ABOUT V(MIN)
C
   20 CONTINUE
      DO 23 I=1,N1
      IF(I-MIN) 21,23,21
   21 ION=(I-1)*N
      DO 22 J=1,N
      JION=J+ION
      JMINON=J+MINON
   22 V(JION)=Z*V(JION)+OMZ*V(JMINON)
  222 F(I)=DIFF(I,ION)
      IF(IVIOL.NE.6)GO TO 23
      DO 225 J=1,N
      JION=J+ION
      JMINON=J+MINON
  225 V(JION)=0.25D0*(3.D0*V(JION)+V(JMINON))
      GO TO 222
   23 CONTINUE
      NSHRNK=NSHRNK+1
C
C     RANK FUNCTION VALUES-HIGHEST(MAX),NEXT HIGHEST(MM),& LOWEST(MIN)
C
   24 DO 26 I=1,N1
      FIT=F(I)
      IF(FIT.GT.F(MAX))MAX=I
      IF(FIT.LT.F(MIN))MIN=I
   26 CONTINUE
      MM=MIN
      DO 27 I=1,N1
      IF(I.EQ.MAX)GOTO27
      IF(F(I).GT.F(MM))MM=I
   27 CONTINUE
      MAXON=(MAX-1)*N
      MINON=(MIN-1)*N
      GO TO KRANK,(2,28)
C
C     TEST FOR CONVERGENCE
C
   28 ITER=ITER+1
      IF(ITER.LT.NITER)GOTO30
      M=0
      KK=0
      DO 35 L=1,N1
      WRITE(7,107)F(L)
      DO 36 I=1,IK
      IF(KV(IJJ+I,1).EQ.2)GOTO36
      KK=KK+1
      DO 37 J=1,5
      IF(KV(IJJ+I,J).EQ.1)GOTO37
      M=M+1
      VM(KK,J)=V(M)
   37 CONTINUE
      WRITE(7,107)(VM(KK,LN),LN=1,5)
   36 CONTINUE
   35 CONTINUE
      WRITE(6,106)F(MIN)
      PMIN=F(MIN)
      MINI=MIN
      RETURN
C
C     WRITE PROGRESS OF OPTIMIZATION EVERY NPRINT ITERATIONS
C
   30 IF(MOD(ITER,NPRINT).NE.0)GOTO40
      WRITE(6,112)ITER
      I=MAXON+1
      J=MAXON+N
      WRITE(6,119)
      WRITE(6,107)LABP,MAX,(V(K),K=I,J)
      WRITE(6,104)F(MAX)
      I=MINON+1
      J=MINON+N
      WRITE(6,120)
      WRITE(6,107)LABP,MIN,(V(K),K=I,J)
      WRITE(6,105)F(MIN)
   40 IF(MOD(ITER,NPRT).EQ.0)WRITE(6,102)ITER,F(MAX),F(MIN)
      IF(MOD(ITER,NSCAN).NE.0)GOTO61
      WRITE(6,108)
      DO 62 I=1,ISCAN
   62 WRITE(6,109)(VM(I,J),J=1,5)
      WRITE(6,110)
      WRITE(6,109)(V(I),I=1,NN)
  108 FORMAT(20X,' VM( ) MATRIX ')
  109 FORMAT(5G14.6)
  110 FORMAT(20X,' V( ) MATRIX ')
C
C     CHECK CONVERGENCE EVERY NCHECK ITERATIONS
C
   61 IF(MOD(ITER,NCHECK).NE. 0)GOTO3
      IF(F(MIN).GT.FMIN)GOTO41
C
C     OPTIMIZATION SUCCESS *****
C
      WRITE(6,117)
      WRITE(6,112)ITER
      GO TO 50
C
C     CONVERGENCE IF FMAX-FMIN LESS THAN SMALL*FMIN
C
   41 IF(F(MAX)-F(MIN).GT.SMALL*F(MIN))GOTO3
C
C     STUCK ON SMALL IF ABS.(VMAX-VMIN) LESS THAN ABS.(SMALL * VMIN)
C
      MAXN=MAXON+N
      MINN=MINON+N
      DO 45 I=1,N
      IMN=I-N
      IMINON=IMN+MINN
      IMAXON=IMN+MAXN
      VIMINN=V(IMINON)
      ABSMIN=DABS(VIMINN)
      IF(DABS(V(IMAXON)-VIMINN).GT.SMALL*ABSMIN.AND.ABSMIN.GT.SMALL)
     1GOTO3
   45 CONTINUE
      WRITE(6,116)
      WRITE(6,112)ITER
   50 DO 60 I=1,N
      IMINON=I+MINON
   60 V(I)=V(IMINON)
      WRITE(6,121)
      WRITE(6,115)(V(I),I=1,N)
      WRITE(6,122)F(MIN)
      PMIN=F(MIN)
      MINI=1
      WRITE(7,107)F(MIN),(V(K),K=1,N)
      K=0
      KK=0
      DO 63 I=1,IK
      IF(KV(IJJ+I,1).EQ.2)GOTO63
      KK=KK+1
      DO 64 J=1,5
      IF(KV(IJJ+I,J).EQ.1)GOTO64
      K=K+1
      VM(KK,J)=V(K)
   64 CONTINUE
   63 CONTINUE
      RETURN
      END
      SUBROUTINE KINSOL(II)
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON/KIN/INV1,INV2,TH(542,5),S(542,5),C(542,5),W(542,5),W1,STH
     1,ALP(542,5),AJ(542,5),AL(10),SINE2,COSE2,SINE3,COSE3,SINB3,COSB3
      COMMON/CONTRL/NPTS,ILINK,ILOCI,NPLOTS,IK,JX,IQ,IJJ,IVAR,INPT,ISW
     1,ISET,ISYN,IDEV,IKK,IEAL,INCM,ITORQ,IDOT,MONITR,IOUT,JXX,IDMSP,IAN
      W11=W1*W1
      AL1=AL(1)
      AL4=AL(4)
      AL2C=AL(2)
      AL3C=AL(3)
      AL23C=2.0D0*AL3C
      SALP3=0.0D0
      SALP2=0.0D0
      SAJ3=0.0D0
      SAJ2=0.0D0
      IF(ILINK.EQ.1)GOTO2
      AL2A=AL(5)
      AL3B=AL(8)
      AL5=AL(7)
      AL6=AL(6)
      AL26=2.0D0*AL6
      SALP6=0.0D0
      SALP5=0.0D0
      SAJ6=0.0D0
      SAJ5=0.0D0
    2 NPT=NPTS
      III=II
      IF(ISW.EQ.1)GOTO3
      TH(1,1)=STH
      NPTS=1
      W1=0.0D0
      W11=0.0D0
    3 DO 1 K=1,NPTS
      I=III+K
      SIN1=DSIN(TH(I,1))
      COS1=DCOS(TH(I,1))
      S(I,1)=SIN1
      C(I,1)=COS1
      C1=AL1*COS1-AL4
      C2=AL1*SIN1
      DC1=-AL1*W1*SIN1
      DC2=AL1*W1*COS1
      D2C1=-DC2*W1
      D2C2=DC1*W1
      D3C1=-D2C2*W1
      D3C2=D2C1*W1
      THQ3=DYAD(INV1,AL3C,AL23C,AL2C,C1,C2,DC1,DC2,D2C1,D2C2,D3C1,D3C2,
     1THP2,SIN3,COS3,SIN2,COS2,W3,W2,ALP3,ALP2,AJ3,AJ2)
      TH(I,3)=THQ3
      TH(I,2)=THP2
      S(I,3)=SIN3
      C(I,3)=COS3
      S(I,2)=SIN2
      C(I,2)=COS2
      W(I,3)=W3
      W(I,2)=W2
```

```
      ALP(I,3)=ALP3
      ALP(I,2)=ALP2
      AJ(I,3)=AJ3
      AJ(I,2)=AJ2
      IF(ILINK.EQ.1)GOTO1
      SANG2=SIN2*COSE2+COS2*SINE2
      CANG2=COS2*COSE2-SIN2*SINE2
      SANG3=SIN3*COSE3-COS3*SINE3
      CANG3=COS3*COSE3+SIN3*SINE3
      W22=W2*W2
      W33=W3*W3
      C1=AL1*COS1+AL2A*CANG2+AL3B*CANG3-AL4
      C2=AL1*SIN1+AL2A*SANG2+AL3B*SANG3
      DC1=-AL1*W1*SIN1-AL2A*W2*SANG2-AL3B*W3*SANG3
      DC2=AL1*W1*COS1+AL2A*W2*CANG2+AL3B*W3*CANG3
      D2C1=-AL1*W11*COS1-AL2A*ALP2*SANG2-AL2A*W22*CANG2
      D2C1=D2C1-AL3B*ALP3*SANG3-AL3B*W33*CANG3
      D2C2=-AL1*W11*SIN1+AL2A*ALP2*CANG2-AL2A*W22*SANG2
      D2C2=D2C2+AL3B*ALP3*CANG3-AL3B*W33*SANG3
      W222=W22*W2
      W333=W33*W3
      ALPW2=ALP2*W2
      ALPW3=ALP3*W3
      D3C1=AL1*W11*W1*SIN1-AL2A*AJ2*SANG2-3.0D0*AL2A*ALPW2*CANG2
      D3C1=D3C1+AL2A*W222*SANG2-AL3B*AJ3*SANG3
      D3C1=D3C1-3.0D0*AL3B*ALPW3*CANG3+AL3B*W333*SANG3
      D3C2=-AL1*W11*W1*COS1+AL2A*AJ2*CANG2-3.0*AL2A*ALPW2*SANG2
      D3C2=D3C2-AL2A*W222*CANG2+AL3B*AJ3*CANG3
      D3C2=D3C2-3.0D0*AL3B*ALPW3*SANG3-AL3B*W333*CANG3
      THQ6=DYAD(INV2,AL6,AL26,AL5,C1,C2,DC1,DC2,D2C1,D2C2,D3C1,D3C2,
     1THP5,SIN6,COS6,SIN5,COS5,W6,W5,ALP6,ALP5,AJ6,AJ5)
      TH(I,4)=THQ6
      TH(I,5)=THP5
      S(I,4)=SIN6
      C(I,4)=COS6
      S(I,5)=SIN5
      C(I,5)=COS5
      W(I,4)=W6
      W(I,5)=W5
      ALP(I,4)=ALP6
      ALP(I,5)=ALP5
      AJ(I,4)=AJ6
      AJ(I,5)=AJ5
    1 CONTINUE
      NPTS=NPT
      RETURN
      END
      FUNCTION DYAD(INV,ALQ,ALQ2,ALP,C1,C2,DC1,DC2,D2C1,D2C2,D3C1,D3C2,
     1THP,SINQ,COSQ,SINP,COSP,DTHQ,DTHP,D2THQ,D2THP,D3THQ,D3THP)
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON/CONTRL/NPTS,ILINK,ILOCI,NPLOTS,IK,JX,IQ,IJJ,IVAR,INPT,ISW
     1,ISET,ISYN,IDEV,IKK,IBAL,IMOM,ITORQ,IDOT,MONITR,IOUT,JXX,IDMSP,IAN
      C3=C1*C1+C2*C2+ALQ*ALQ-ALP*ALP
      C4=ALQ2*C1
      C5=ALQ2*C2
      ARG=C4*C4+C5*C5
      SINQ=(-C3*C5+INV*C4*DSQRT(ARG-C3*C3))/ARG
      COSQ=-(C3+C5*SINQ)/C4
      THQ=DATAN2(SINQ,COSQ)
      SINP=-(C2+ALQ*SINQ)/ALP
      COSP=-(C1+ALQ*COSQ)/ALP
      THP=DATAN2(SINP,COSP)
      DC3=2.0*(C1*DC1+C2*DC2)
      DC4=ALQ2*DC1
      DC5=ALQ2*DC2
      DIV1=C4*SINQ-C5*COSQ
      DIV2=ALP*(COSP+SINP)
      DIV3=ALQ*(COSQ+SINQ)
      DTHQ=(DC3+DC4*COSQ+DC5*SINQ)/DIV1
      DTHP=(DC1-DC2-DTHQ*DIV3)/DIV2
      D2C3=2.0D0*(DC1*DC1+DC2*DC2+C1*D2C1+C2*D2C2)
      D2C4=ALQ2*D2C1
      D2C5=ALQ2*D2C2
      D2THQ=D2C3+D2C4*COSQ+D2C5*SINQ+2.0*DTHQ*(DC5*COSQ-DC4*SINQ)
      D2THQ=(D2THQ-DTHQ*DTHQ*(C4*COSQ+C5*SINQ))/DIV1
      DIV4=ALQ*(COSQ-SINQ)
      DIV5=ALP*(COSP-SINP)
      D2THP=(D2C1-D2C2-D2THQ*DIV3-DTHQ*DTHQ*DIV4-DTHP*DTHP*DIV5)/DIV2
      DYAD=THQ
      D3C3=6.0D0*(D2C1*DC1+D2C2*DC2)+2.0*(D3C1*C1+D3C2*C2)
      D3C4=ALQ2*D3C1
      D3C5=ALQ2*D3C2
      D3THQ=D3C3+D3C4*COSQ+D3C5*SINQ+3.0D0*DTHQ*(D2C5*COSQ-D2C4*SINQ)
      D3THQ=D3THQ+3.0D0*D2THQ*(DC5*COSQ-DC4*SINQ-DTHQ*(C4*COSQ+C5*SINQ))
      D3THQ=D3THQ+DTHQ*DTHQ*(DTHQ*(C4*SINQ-C5*COSQ)-3.0D0*(DC4*COSQ+DC5*
     1SINQ))
      D3THQ=D3THQ/DIV1
      D3THP=D3C1-D3C2-D3THQ*DIV3+DTHQ*(DTHQ*DTHQ*DIV3-3.0D0*D2THQ*DIV4)
      D3THP=(D3THP+DTHP*(DTHP*DTHP*DIV2-3.0D0*D2THP*DIV5))/DIV2
      RETURN
      END
      SUBROUTINE DYNMIC(IJK,JXX)
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON/PARAM/AM1,AM2,AM3,AM6,AM5,AI2,AI3,AI6,AI5
     1,R1,R2,R3,R6,R5,RM1,RM2,RM3,RM6,RM5,G1,G2,G3,G6,G5
     1,AK1,AK2,AK3,AK4,AK5,AK6,AS2,AS3,AS4,AS5,AS6
     1,DELT2,DELT3,DELT4,DELT5,DELT6
      COMMON/FORCE/X(552,8),Y(552,8),DX(552,8),DY(552,8),TOR(552,8)
     1,GRAV,FDEG(7),XDEG,YDEG,ADEG,TDEG,HOLD,AERR,ISRT
      COMMON/KIN/INV1,INV2,TH(542,5),S(542,5),C(542,5),W(542,5),W1,STH
     1,ALP(542,5),AJ(542,5),AL(10),SINE2,COSE2,SINE3,COSE3,SINB3,COSB3
      COMMON/CONTRL/NPTS,ILINK,ILOCI,NPLOTS,IK,JX,IQ,IJJ,IVAR,INPT,ISW
     1,ISET,ISYN,IDEV,IKK,IBAL,IMOM,ITORQ,IDOT,MONITR,IOUT,KXX,IDMSP,IAN
      COMMON/CONST/PIE,PIE2,RAD,DEG,PY
      COMMON/STORE/AM(40),R(40),G(40),AI(40),AK(60),AS(60),DELT(60)
      COMMON/SYNTH/FP(7),ITP(7),KV(25,5)
      W11=W1*W1
      STOR=0.0D0
      ST=0.0D0
      IF(ILINK.EQ.0)GOTO16
      CANGB3=0.0D0
      SANGB3=0.0D0
      X7=0.0D0
      Y7=0.0D0
      X5=0.0D0
      Y5=0.0D0
      DY7=0.0D0
      DX7=0.0D0
      DX5=0.0D0
      DY5=0.0D0
   16 IF(IJK.NE.-1)GOTO4
      AAM1=AM1
      AAM2=AM2
      AAM3=AM3
      RR1=R1
      RR2=R2
      RR3=R3
      GG1=G1
      GG2=G2
      GG3=G3
      AAI2=AI2
      AAI3=AI3
      AAK1=AK1
      AAK2=AK2
      AAK3=AK3
      ADELT2=0.0D0
      ADELT3=0.0D0
      AAS2=0.0D0
      AAS3=0.0D0
      IF(ILINK.EQ.1)GOTO5
      AAM6=AM6
      AAM5=AM5
      RR6=R6
      RR5=R5
      GG6=G6
      GG5=G5
      AAI5=AI5
      AAI6=AI6
      AAK4=AK4
      AAK5=AK5
      AAK6=AK6
      AAS4=0.0D0
      AAS5=0.0D0
      AAS6=0.0D0
      ADELT4=0.0D0
      ADELT5=0.0D0
      ADELT6=0.0D0
      GOTO5
    4 AAK1=AK1+AK(1+JXX)
      AAK2=AK2+AK(2+JXX)
      AAK3=AK3+AK(3+JXX)
      AAS2=AS(2+JXX)
      AAS3=AS(3+JXX)
      ADELT2=DELT(2+JXX)
      ADELT3=DELT(3+JXX)
      IF(KV(1+IJK,1).NE.2)GOTO6
      AAM1=AM1
      RR1=R1
      GG1=G1
      GOTO7
    6 GG1=SOL(1,AM(1+IJK),R(1+IJK),G(1+IJK),AM1,R1,RM1,G1,AI1,
     1AI(1+IJK),AAM1,RR1,RRM1,AAI1)
    7 IF(KV(2+IJK,1).NE.2)GOTO8
      AAM2=AM2
      RR2=R2
      GG2=G2
      AAI2=AI2
      GOTO9
    8 GG2=SOL(2,AM(2+IJK),R(2+IJK),G(2+IJK),AM2,R2,RM2,G2,AI2,
     1AI(2+IJK),AAM2,RR2,RRM2,AAI2)
    9 IF(KV(3+IJK,1).NE.2)GOTO10
      AAM3=AM3
      RR3=R3
      GG3=G3
      AAI3=AI3
      GOTO11
   10 GG3=SOL(2,AM(3+IJK),R(3+IJK),G(3+IJK),AM3,R3,RM3,G3,AI3,
     1AI(3+IJK),AAM3,RR3,RRM3,AAI3)
   11 IF(ILINK.EQ.1)GOTO5
      AAK4=AK4+AK(4+JXX)
      AAK5=AK5+AK(5+JXX)
      AAK6=AK6+AK(6+JXX)
      AAS4=AS(4+JXX)
      AAS5=AS(5+JXX)
      AAS6=AS(6+JXX)
      ADELT4=DELT(4+JXX)
      ADELT5=DELT(5+JXX)
      ADELT6=DELT(6+JXX)
      IF(KV(5+IJK,1).NE.2)GOTO12
      AAM6=AM6
      RR6=R6
      GG6=G6
      AAI6=AI6
      GOTO13
   12 GG6=SOL(2,AM(5+IJK),R(5+IJK),G(5+IJK),AM6,R6,RM6,G6,AI6,
     1AI(5+IJK),AAM6,RR6,RRM6,AAI6)
   13 IF(KV(4+IJK,1).NE.2)GOTO14
      AAM5=AM5
      RR5=R5
      GG5=G5
      AAI5=AI5
      GOTO5
   14 GG5=SOL(2,AM(4+IJK),R(4+IJK),G(4+IJK),AM5,R5,RM5,G5,AI5,
     1AI(4+IJK),AAM5,RR5,RRM5,AAI5)
    5 SING1=DSIN(GG1)
      COSG1=DCOS(GG1)
      SING2=DSIN(GG2)
      COSG2=DCOS(GG2)
      SING3=DSIN(GG3)
      COSG3=DCOS(GG3)
      AL1=AL(1)
      SW1=W1
      AL2C=AL(2)
      AL3C=AL(3)
      AL4=AL(4)
      A1=-AL1*W11
      A2=-RR1*W11
      AL23C=AL3C*AL3C
      RR23=RR3*RR3
      D23=RR23+AL23C-2.0D0*RR3*AL3C*COSG3
      D3=DSQRT(D23)
      SIND3=RR3*SING3/D3
      COSD3=(D23+AL23C-RR23)/(2.0D0*D3*AL3C)
      IF(ILINK.EQ.1)GOTO2
      SING6=DSIN(GG6)
      COSG6=DCOS(GG6)
      SING5=DSIN(GG5)
      COSG5=DCOS(GG5)
      AL2A=AL(5)
      AL6=AL(6)
      AL5=AL(7)
      AL3B=AL(8)
      AL3A=AL(10)
      RR26=RR6*RR6
      AL26=AL6*AL6
      D26=RR26+AL26-2.0D0*RR6*AL6*COSG6
      D6=DSQRT(D26)
      SIND6=RR6*SING6/D6
      COSD6=(D26+AL26-RR26)/(2.0D0*D6*AL6)
    2 NPT=NPTS
      II=IAN
      IKJ=1
      IF(ISW.NE.0)GOTO15
      II=361
      NPTS=362
      W1=0.0D0
      W11=0.0D0
      IKJ=362
   15 DO 1 K=IKJ,NPTS
      I=K+II
      SIN1=S(I,1)
      COS1=C(I,1)
      TH2=TH(I,2)
      SIN2=S(I,2)
      COS2=C(I,2)
      SIN3=S(I,3)
      TH3=TH(I,3)
      COS3=C(I,3)
      W2=W(I,2)
      W3=W(I,3)
      ALP2=ALP(I,2)
      ALP3=ALP(I,3)
      SANG1=SIN1*COSG1+COS1*SING1
      SANG2=SIN2*COSG2+COS2*SING2
      SANG3=SIN3*COSG3+COS3*SING3
      SANGD3=-SIN3*COSD3+COS3*SIND3
      CANG1=COS1*COSG1-SIN1*SING1
      CANG2=COS2*COSG2-SIN2*SING2
      CANG3=COS3*COSG3-SIN3*SING3
      CANGD3=-COS3*COSD3-SIN3*SIND3
      W22=W2*W2
      W33=W3*W3
      IF(IDOT.NE.0)GOTO31
      W222=W22*W2
      ALPW2=ALP2*W2
      W333=W33*W3
      ALPW3=ALP3*W3
      AJ2=AJ(I,2)
      AJ3=AJ(I,3)
   31 AY1=-RR1*W11*SANG1+GRAV
      AX1=-RR1*W11*CANG1
      TP1=-AL1*W11*SIN1
      TP2=-AL1*W11*COS1
      AY2=TP1+RR2*(ALP2*CANG2-W22*SANG2)+GRAV
      AX2=TP2-RR2*(ALP2*SANG2+W22*CANG2)
      AY3=D3*(ALP3*CANGD3-W33*SANGD3)+GRAV
      AX3=-D3*(ALP3*SANGD3+W33*CANGD3)
      IF(ILINK.EQ.1)GOTO3
      TH6=TH(I,4)
      SIN6=S(I,4)
```

```
      COS6=C(I,4)
      TH5=TH(I,5)
      SIN5=S(I,5)
      COS5=C(I,5)
      W6=W(I,4)
      W5=W(I,5)
      ALP6=ALP(I,4)
      ALP5=ALP(I,5)
      SANG5=SIN5*COSG5+CCS5*SINC5
      SANG6=SIN6*COSG6+CCS6*SING6
      SANGT3=-SIN3*COSE3+COS3*SINE3
      SANGD6=-SIN6*COSD6+COS6*SIND6
      CANG5=CCS5*COSG5-SIN5*SINC5
      CANG6=COS6*COSG6-SIN6*SING6
      CANGT3=-COS3*COSE3-SIN3*SINE3
      SANGB3=SIN3*CCSB3+CCS3*SINB3
      CANGB3=COS3*CCSB3-SIN3*SINB3
      CANGD6=-COS6*COSD6-SIN6*SIND6
      SANGE2=SIN2*CCSE2+CCS2*SINE2
      CANGE2=COS2*CCSE2-SIN2*SINE2
      SANGE3=SIN3*COSE3-COS3*SINE3
      CANGE3=COS3*COSE3+SIN3*SINE3
      W55=W5*W5
      W66=W6*W6
      AY5=TP1+AL2A*(ALP2*CANGE2-W22*SANGE2)+RR5*(ALP5*CANG5-W55*SANG5)+G
     1RAV
      AX5=TP2-AL2A*(ALP2*SANGE2+W22*CANGE2)-RR5*(ALP5*SANG5+W55*CANG5)
      AY6=AL3B*(ALP3*CANGT3-W33*SANGT3)+D6*(ALP6*CANGD6-W65*SANGD6)+GRAV
      AX6=-AL3B*(ALP3*SANGT3+W33*CANGT3)-D6*(ALP6*SANGD6+W66*CANGD6)
      T5=0.CDC
      T6=0.CDC
      T7=0.0D0
      IF(IDNSF.EC.0)GOTC17
      TH6=TH(I,4)
      TH5=TH(I,5)
      TH3=TH(I,3)
      TH2=TH(I,2)
      ZK=TH3-TH6
      T7=AAK6*(W3-W6)+AAS6*(ZK+ADELT6)+AS6*(ZK+DELT6)
      ZK=TH6-TH5
      T6=AAK5*(W6-W5)+AAS5*(ZK+ADELT5)+AS5*(ZK+DELT5)
      ZK=TH5-TH2
      T5=AAK4*(W5-W2)+AAS4*(ZK+ADELT4)+AS4*(ZK+DELT4)
   17 TOR(K,6)=T5
      TOR(K,7)=T6
      TCR(K,8)=T7
      TAN5=SIN5/COS5
      TAN6=SIN6/COS6
      BT1=TAN6-TAN5
      A1=(AAI5*ALP5+AAM5*RR5*(AY5*CANG5-AX5*SANG5)-T6+T5)/(AL5*COS5)
      A2=(AAI6*ALP6+AAM6*(AY6*(RR6*CANG6-AL6*COS6)+AX6*(AL6*SIN6-RR6*SAN
     1G6))-T7+T6)/(AL6*COS6)
      X6=(A1-A2)/BT1
      Y6=A1+X6*TAN5
      Y5=Y6-AAM5*AY5
      X5=X6-AAM5*AX5
      Y7=Y6+AAM6*AY6
      X7=X6+AAM6*AX6
      X(K,6)=X6
      Y(K,6)=Y6

      X(K,7)=X7
      Y(K,7)=Y7
      X(K,5)=X5
      Y(K,5)=Y5
      IF(IDOT.NE.0)GOTO3
      W555=W55*W5
      ALPW5=ALP5*W5
      W666=W66*W6
      ALPW6=ALP6*W6
      AJ5=AJ(I,5)
      AJ6=AJ(I,4)
      AJY5=TP2*W1+AL2A*(AJ2*CANCE2-3.0D0*ALPW2*SANGE2-W222*CANGE2)
      AJY5=AJY5+RR5*(AJ5*CANG5-3.0D0*ALPW5*SANG5-W555*CANG5)
      AJX5=-TP1*W1-AL2A*(AJ2*SANGE2+3.0D0*ALPW2*CANGE2-W222*SANGE2)
      AJX5=AJX5-RR5*(AJ5*SANG5+3.0D0*ALPW5*CANG5-W555*SANG5)
      AJY6=AL3B*(AJ3*CANGT3-3.0D0*ALPW3*SANGT3-W333*CANGT3)
      AJY6=AJY6+D6*(AJ6*CANGD6-3.CD0*ALPW6*SANGD6-W666*CANGD5)
      AJX6=-AL3B*(AJ3*SANGT3+3.0D0*ALPW3*CANGT3-W333*SANGT3)
      AJX6=AJX6-D6*(AJ6*SANGD6+3.0D0*ALPW6*CANGD6-W666*SANGD6)
      SEC6=1.CD0+TAN6*TAN6
      SEC5=1.0D0+TAN5*TAN5
      DT7=AAK6*(ALP3-ALP6)
      DT6=AAK5*(ALP6-ALP5)
      DT5=AAK4*(ALP5-ALP2)
      DA1=(AAI5*AJ5+AAM5*RR5*(AJY5*CANG5-AY5*W5*SANG5-AJX5*SANG5-AX5*W5*
     1CANG5)+DT5-DT6+A1*AL5*W5*SIN5)/(AL5*CCS5)
      DA2=AAI6*AJ6+AAM6*(AJY6*(RR6*CANG6-AL6*COS6)+AY6*W5*(AL6*SIN6-RR6*
     1SANG6)+AJX6*(AL6*SIN6-RR6*SANG6)+AX6*W6*(AL6*COS6-RR6*CANG6))
      DA2=(DA2-DT7+DT6+A2*AL6*W6*SIN6)/(AL6*COS6)
      DX6=(BT1*(DA1-DA2)-(A1-A2)*(W6*SEC6-W5*SEC5))/(BT1*BT1)
      DY6=DA1+DX6*TAN5+X6*W5*SEC5
      DY5=DY6-AAM5*AJY5
      DX5=DX6-AAM5*AJX5
      DY7=DY6+AAM6*AJY6
      DX7=DX6+AAM6*AJX6
      DX(K,6)=DX6
      DY(K,6)=DY6
      DX(K,7)=DX7
      DY(K,7)=DY7
      DX(K,5)=DX5
      DY(K,5)=DY5
    3 T4=0.0D0
      T3=0.CD0
      T2=0.0D0
      IF(IDNSF.EC.0)GOTC18
      ZK=TH3-TH2
      T4=-AAK3*W3-AAS3*(TH3+ADELT3)-AS3*(TH3+DELT3)
      T3=AAK2*(W3-W2)+AAS2*(ZK+ADELT3)+AS2*(ZK+DELT2)
      T2=AAK1*(W2-W1)
      TOR(K,3)=T2
      TCR(K,4)=T3
      TCR(K,5)=T4
   18 TAN3=SIN3/COS3
      TAN2=SIN2/COS2
      BT2=TAN3-TAN2
      A3=AAI2*ALP2+AAM2*RR2*(AY2*CANG2-AX2*SANG2)+T2-T3
      IF(ILINK.NE.1)A3=A3+AL2A*(X5*SANGE2-Y5*CANGE2)-T5
      A3=A3/(AL2C*CCS2)
      Z1=AAM3*AY3+Y7
      Z2=AAM3*AX3+X7

      A4=AAI3*ALP3+AAM3*RR3*(AY3*CANG3-AX3*SANG3)
      A4=A4+AL3C*(Z2*SIN3-Z1*CCS3)-T4+T3
      IF(ILINK.NE.1)A4=A4+AL3A*(Y7*CANGB3-X7*SANGB3)+T7
      A4=A4/(AL3C*COS3)
      X3=(A3-A4)/BT2
      Y3=A3+X3*TAN2
      Y4=Y7+Y3+AAM3*AY3
      X4=X7+X3+AAM3*AX3
      Y2=Y5+Y3-AAM2*AY2
      X2=X5+X3-AAM2*AX2
      Y1=Y2-AAM1*AY1
      X1=X2-AAM1*AX1
      X(K,1)=X1
      Y(K,1)=Y1
      X(K,2)=X2
      Y(K,2)=Y2
      X(K,3)=X3
      Y(K,3)=Y3
      X(K,4)=X4
      Y(K,4)=Y4
      IF(IDCT.NE.0)GOTO30
      AJY1=AX1*W1
      AJX1=RR1*W11*SANG1*W1
      AJY2=TP2*W1+RR2*(AJ2*CANG2-3.0D0*ALPW2*SANG2-W222*CANG2)
      AJX2=-TP1*W1-RR2*(AJ2*SANG2+3.0D0*ALPW2*CANG2-W222*SANG2)
      AJY3=D3*(AJ3*CANGD3-3.0D0*ALPW3*SANGD3-W333*CANGD3)
      AJX3=-D3*(AJ3*SANGD3+3.0D0*ALPW3*CANGD3-W333*SANGD3)
      SEC3=1.0D0+TAN3*TAN3
      SEC2=1.CD0+TAN2*TAN2
      DT4=-AAK3*ALP3
      DT3=AAK2*(ALP3-ALP2)
      DT2=AAK1*ALP2
      DA3=AAI2*AJ2+AAM2*RR2*((AJY2-AX2*W2)*CANG2-(AJX2+AY2*W2)*SANG2)
      IF(ILINK.NE.1)DA3=DA3+AL2A*((DX5+Y5*W2)*SANGE2+(X5*W2-DY5)*CANGE2)
     1-DT5
      DA3=(DA3+DT2-DT3+A3*AL2C*W2*SIN2)/(AL2C*CCS2)
      DA4=AAI3*AJ3+AAM3*RR3*((AJY3-AX3*W3)*CANG3-(AY3*W3+AJX3)*SANG3)
      DA4=DA4+AL3C*((DX7+AAM3*AJX3+Z1*W3)*SIN3+(Z2*W3-DY7-AAM3*AJY3)*COS
     13)-DT4+DT3
      IF(ILINK.NE.1)DA4=DA4+AL3A*((DY7-X7*W3)*CANGB3-(DX7+Y7*W3)*SANGB3)
     1+DT7
      DA4=(DA4+A4*AL3C*W3*SIN3)/(AL3C*CCS3)
      DX3=(BT2*(DA3-DA4)-(A3-A4)*(W3*SEC3-W2*SEC2))/(BT2*BT2)
      DY3=DA3+DX3*TAN2+X3*W2*SEC2
      DY4=DY7+DY3+AAM3*AJY3
      DX4=DX7+DX3+AAM3*AJX3
      DY2=DY5+DY3-AAM2*AJY2
      DX2=DX5+DX3-AAM2*AJX2
      DY1=DY2-AAM1*AJY1
      DX1=DX2-AAM1*AJX1
      DX(K,1)=DX1
      DY(K,1)=DY1
      DX(K,2)=DX2
      DY(K,2)=DY2
      DX(K,3)=DX3
      DY(K,3)=DY3
      DX(K,4)=DX4
      DY(K,4)=DY4
   30 CONTINUE
      TOR(K,1)=T2+AL1*(Y2*COS1-X2*SIN1)

      TCR(K,2)=-Y4*AL4-T4
    1 CONTINUE
      NPTS=NPT
      RETURN
      END
      FUNCTION SOL(I,A,B,C,D,Q,E,F,G,H,T,R,Z,P)
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON/CONST/PIE,PIE2,RAC,DEG,PY
      U=A*B
      S=E*Q
      V1=DCCS(F)
      V2=DSIN(F)
      X=E*V1+U*DCOS(C)
      Y=E*V2+U*DSIN(C)
      Z=DSQRT(X*X+Y*Y)
      T=D+A
      IF(T.NE.0.0D0)GOTC1
      R=0.CD0
      Z=0.0D0
      P=0.0D0
      SCL=0.0D0
      RETURN
    1 R=Z/T
      P=1.0
      IF(I.EQ.1)GOTC2
      P=H+A*B*B+G+S-Z*R
    2 SOL=DATAN2(Y,X)
    4 IF(SCL.LE.PIE2)GOTO3
      SOL=SOL-PIE2
      GOTO4
    3 IF(SCL.GE.0.0D0)GOTO5
      SCL=SOL+PIE2
      GOTO3
    5 CONTINUE
      RETURN
      END
END OF FILE
```

S8 NO. QMCG8 UNIVERSITY OF MICHIGAN TERMINAL SYSTEM AT NUMAC 13:04:28 MON 5-FEB-79

USER: MCG8
PROJECT NO: NMC0

**** OBTAINED AT 13:04:28 MON 5-FEB-79
**** RELEASED AT 13:04:55 MON 5-FEB-79
**** LINES PRINTED 2365
**** PAGES PRINTED 41

QMCG8

MCG8