

```

Jan 16, 07 14:13          geomain2.f          Page 1/8
PROGRAM GEOMAIN
C THE PROGRAM SOLVES THE 2 MAIN GEODETIC PROBLEMS FOR THE ELLIPSOID.
C PROGRAMMED 1993.07.04 BY C.C.TSCHERNING, BASED ON SUBROUTINES FROM
C T.VINCENY, FORMERLY OF NGS/NOAA, USA. LAST CHANGE 2007-01-15 BY CCT.
  IMPLICIT REAL*8 (A-H,O-Z), LOGICAL (L)
  COMMON /CONST/D0,D1,D2,D3,D4,RADDEG,DEGRAD,PI,TWOPI,TRIFLE
  COMMON /ELL/ ELLIP(15,2),ELLIPTX(16)
  CHARACTER*25 ELLIPTX
  CHARACTER*72 INAME,PNAME

C
  LSTOP=.FALSE.
  WRITE(*,111)
111  FORMAT(' GEODETIC MAIN PROBLEMS, VER. 1993.07.07' )
  WRITE(*,*) ' INTERACTIVE RUN ? (T/F) '
  READ(*,*)LINTER
  IF (LINTER) WRITE(*,112)
112  FORMAT(' SELECT SPHEROID:/'
* 1: SPHERE, 2: CLARKE 1866, /
* 3: HAYFORD 1909 (INTERNATIONAL), 4: GRS 1980' /
* 5: CLARK1880, 6: BESSEL 1841, 7: KRASOVSKY 1940, 8: WGS 1972' /
* 9: AUSTRALIAN 1965, 10: AIRY1849, 11: EVEREST 1830' /
* 12: HOUGH 1956., 13: FISHER 1960, 14&15: SPHERE, 16: INPUT ' /)
  READ(*,*) ISPHER
  IF (ISPHER.LT.17) THEN
  WRITE(*,*) ' ELLIPSOID: ', ELLIPTX (ISPHER), ISPHER
  SEMIA=ELLIP (ISPHER,1)
  SEMIB=ELLIP (ISPHER,2)
  ELSE
  WRITE(*,*) ' INPUT SEMI MAJOR AND MINOR AXIS IN M '
  READ(*,*) SEMIA, SEMIB
  END IF
  FLA1=(SEMIA-SEMIB)/SEMIA
  IF (FLA1.GT.D0) FLA=D1/FLA1
  WRITE(*,100) FLA
100  FORMAT(' 1/FLATTENING =', F12.7)

C
  PI4=ATAN2(D1,D1)
  PI=D4*PI4
  TWOPI=PI4*8.0D0
  DEGRAD=PI/180.0D0
  RADDEG=180.0D0/PI
  IF (LINTER) WRITE(*,*)
* ' ANGLES IN 1: DD MM SS.S, 2: DD MM.M 3: DD.D, 4: CC.C ? '
  READ(*,*) IANG

C
  IF (LINTER) WRITE(*,*) ' ARE INPUT DATA FROM THE KEY-BOARD (T/F) ? '
  READ(*,*) LKEYB
  IF (LKEYB) THEN
  IFILE=5
  ELSE
  IFILE=12
  IF (LINTER) WRITE(*,*)
* ' INPUT NAME OF FILE WITH INPUT DATA '
  READ(*, '(A)') INAME
  OPEN(12, FILE=INAME, STATUS='OLD')
  END IF

C
798  IF (LINTER) WRITE(*,*)
* ' RESULTING DATA TO BE STORED ON FILE ? (T/F) '
  READ(*,*) LSTORE
  IF (.NOT.LSTORE) GO TO 790
  IF (LINTER) WRITE(*,*)
* ' INPUT NAME OF FILE TO HOLD RESULTING DATA '
  READ(*, '(A)') PNAME
  OPEN(14, FILE=PNAME, STATUS='UNKNOWN')

C
790  IF (LINTER) THEN
  WRITE(*,*) ' DIRECT (LAT., LONG, AZIM., DIST GIVEN) (T), '
  WRITE(*,*) ' OR REVERSE (LAT., LONG., LAT. LONG. GIVEN) (F) ? '

```

```

Jan 16, 07 14:13          geomain2.f          Page 2/8
  END IF
  READ(*,*) LDIREC
  M=0
C
1000 M=M+1
  IF (LDIREC) THEN
  IF (M.EQ.1) THEN
  IF (.NOT.LINTER) WRITE(*,*)
* ' STATION NO., LAT., LONG., AZIMUTH AND DISTANCE OUTPUT '
  WRITE(*,*)
* ' OUTPUT FORMAT IS: LATITUDE, LONGITUDE, BACKWARD AZIMUTH '
  END IF
  IF (LKEYB.AND.LINTER) WRITE(*,*)
* ' INPUT NO. LAT, LONG, AZIMUTH AND DIST. '
  GO TO (401,402,403,403), IANG
401  READ (IFILE, *, END=999) ISTNO, IDLAT, MLAT, SLAT,
* IDLON, MLON, SLON, IDAZ, MAZ, SAZ, S
  GO TO 410
402  READ (IFILE, *, END=999) ISTNO, IDLAT, SLAT, IDLON, SLON, IDAZ, SAZ, S
  GO TO 410
403  READ (IFILE, *, END=999) ISTNO, SLAT, SLON, SAZ, S

C
410  CALL DDEG (IDLAT, MLAT, SLAT, B1, IANG)
  CALL DDEG (IDLON, MLON, SLON, RL1, IANG)
  CALL DDEG (IDAZ, MAZ, SAZ, FAZ, IANG)

C
  GO TO (441,442,443,443), IANG
441  WRITE(*,446) ISTNO, IDLAT, MLAT, SLAT, IDLON, MLON, SLON,
* IDAZ, MAZ, SAZ, S
446  FORMAT(I8, I4, I3, F7.3, I5, I3, F7.3,
* 7X, I5, I3, F7.3, F15.3)
  GO TO 445
442  WRITE(*,447) ISTNO, IDLAT, SLAT, IDLON, SLON,
* IDAZ, SAZ, S
447  FORMAT(I8, I4, F7.3, I5, F7.3, /7X, I5, F7.3, F15.3)
  GO TO 445
443  WRITE(*,308) ISTNO, SLAT, SLON, SAZ, S
308  FORMAT(I8, 2F12.6, /7X, F13.6, F15.3)
445  CONTINUE
  ELSE
  IF (M.EQ.1) THEN
  IF (.NOT.LINTER) WRITE(*,*)
* ' STATION NO., 1. LAT., LONG., STATION NO. 2, LAT., LONG OUTPUT '
  WRITE(*,*)
* ' OUTPUT FORMAT IS: FORWARD & BACKWARD AZIMUTH & DISTANCE '
  END IF
  IF (LKEYB.AND.LINTER) WRITE(*,*)
* ' INPUT NO. LAT, LONG, NO. LAT, LONG. '
  GO TO (1401,1402,1403,1403), IANG
1401  READ (IFILE, *, END=999) ISTNO, IDLAT, MLAT, SLAT,
* IDLON, MLON, SLON, ISTNO2, IDLAT2, MLAT2, SLAT2, IDLON2, MLON2, SLON2
  GO TO 1410
1402  READ (IFILE, *, END=999) ISTNO, IDLAT, SLAT, IDLON, SLON,
* ISTNO2, IDLAT2, SLAT2, IDLON2, SLON2
  GO TO 1410
1403  READ (IFILE, *, END=999) ISTNO, SLAT, SLON, ISTNO2, SLAT2, SLON2

C
1410  CALL DDEG (IDLAT, MLAT, SLAT, B1, IANG)
  CALL DDEG (IDLON, MLON, SLON, RL1, IANG)
  CALL DDEG (IDLAT2, MLAT2, SLAT2, B2, IANG)
  CALL DDEG (IDLON2, MLON2, SLON2, RL2, IANG)
  GO TO (1441,1442,1443,1443), IANG
1441  WRITE(*,1446) ISTNO, IDLAT, MLAT, SLAT, IDLON, MLON, SLON,
* ISTNO2, IDLAT2, MLAT2, SLAT2, IDLON2, MLON2, SLON2
1446  FORMAT(I8, I4, I3, F8.4, I5, I3, F8.4,
* I8, I4, I3, F8.4, I5, I3, F8.4)
  GO TO 1445
1442  WRITE(*,1447) ISTNO, IDLAT, SLAT, IDLON, SLON,
* ISTNO2, IDLAT2, SLAT2, IDLON2, SLON2

```

```

Jan 16, 07 14:13          geomain2.f          Page 3/8
1447  FORMAT(I8,I4,F10.6,I5,F10.6,/I8,I4,F10.6,I5,F10.6)
      GO TO 1445
1443  WRITE(*,1308)ISTNO,SLAT,SLON,ISTNO2,SLAT2,SLON2
1308  FORMAT(I8,2F14.8,/I8,2F14.8)
1445  CONTINUE
      END IF
C
      IF (LDIREC) THEN
      CALL DIRECT(SEMIA,SEMIB,B1,RL1,B2,RL2,FAZ,BAZ,S,IT)
      FAZN=FAZ
      SN=S
      CALL INVERS(SEMIA,SEMIB,B1,RL1,B2,RL2,FAZ,BAZ,S,IT)
      IF (ABS(FAZN-FAZ-TWOPI).LT.1.0D-8) FAZ=FAZ+TWOPI
      IF (ABS(FAZN-FAZ+TWOPI).LT.1.0D-8) FAZ=FAZ-TWOPI
      IF (ABS(SN-S).GT.1.0D-3.OR.ABS(FAZN-FAZ).GT.1.0D-9)
      *WRITE(*,*)'CHECK',SN,S,FAZN,FAZ,IT
      CALL ANG(IDEQ,MIN,SEC,B2,IANG)
      CALL ANG(IDLO,MLO,SLO,RL2,IANG)
      CALL ANG(IDAZ,MAZ,SAZ,BAZ,IANG)
      IOU=6
600  GO TO (601,602,603,603), IANG
601  WRITE(IOU,611)IDEQ,MIN,SEC,IDLO,MLO,SLO,IDAZ,MAZ,SAZ
611  FORMAT(3(I5,I3,F8.4))
      GO TO 610
602  WRITE(IOU,612)IDEQ,SEC,IDLO,SLO,IDAZ,SAZ
612  FORMAT(3(I5,F10.6))
      GO TO 610
603  WRITE(IOU,613)SEC,SLO,SAZ
613  FORMAT(3F16.9)
610  IF (IOU.EQ.6.AND.LSTORE) THEN
      IOU=14
      GO TO 600
      END IF
      ELSE
      CALL INVERS(SEMIA,SEMIB,B1,RL1,B2,RL2,FAZ,BAZ,S,IT)
      B2N=B2
      RL2N=RL2
      CALL DIRECT(SEMIA,SEMIB,B1,RL1,B2,RL2,FAZ,BAZ,S,IT)
      IF (ABS(RL2N-RL2-TWOPI).LT.1.0D-8) RL2=RL2+TWOPI
      IF (ABS(RL2N-RL2+TWOPI).LT.1.0D-8) RL2=RL2-TWOPI
      IF (ABS(B2N-B2).GT.1.0D-9.OR.ABS(RL2N-RL2).GT.1.0D-9)
      *WRITE(*,*)'CHECK',B2N,B2,RL2N,RL2,IT
      CALL ANG(IDAZ,MAZ,SAZ,FAZ,IANG)
      CALL ANG(IDBAZ,MBAZ,SBAZ,BAZ,IANG)
      IOU=6
620  GO TO (621,622,623,623), IANG
621  WRITE(IOU,631)IDAZ,MAZ,SAZ,IDBAZ,MBAZ,SBAZ,S
631  FORMAT(7X,2(I5,I3,F8.4),F16.3)
      GO TO 630
622  WRITE(IOU,632)IDAZ,SAZ,IDBAZ,SBAZ,S
632  FORMAT(7X,2(I5,F10.6),F16.3)
      GO TO 630
623  WRITE(IOU,633)SAZ,SBAZ,S
633  FORMAT(7X,2F16.9,F16.3)
630  IF (IOU.EQ.6.AND.LSTORE) THEN
      IOU=14
      GO TO 620
      END IF
      END IF
      IF (LKEYB.AND.LINTER) THEN
      WRITE(*,*)'STOP?(T/F)'
      READ(*,*)LSTOP
      IF (.NOT.LSTOP) GO TO 1000
      ELSE
      GO TO 1000
      END IF
999  IF (LSTORE) CLOSE(14)
      IF (LKEYB) CLOSE(12)
      STOP

```

Friday February 29, 2008

```

Jan 16, 07 14:13          geomain2.f          Page 4/8
      END
      BLOCK DATA
      IMPLICIT REAL*8 (A-H,O-Z)
      COMMON /CONST/D0,D1,D2,D3,D4,RADDEG,DEGRAD,PI,TWOPI,TRIFLE
      COMMON /ELL/ ELLIP(15,2),ELLIPTX(16)
      CHARACTER*25 ELLIPTX
C
C SEMI-MAJOR AND MINOR AXES OF VARIOUS ELLIPSOIDS.
      DATA ELLIP/6371000.0D0,6378206.4D0,6378388.0D0,6378137.0D0,
      *6378249.145D0,6377397.155D0,6378245.0D0,6378135.0D0,
      *6378160.0D0,6377563.396D0,6377276.345D0,6378270.0D0,
      *6378166.0D0,2*6371000.0D0,
      *6371000.0D0,6356583.8D0,6356911.946D0,6356752.3142D0,
      *6356514.870D0,6356078.963D0,6356863.019D0,6356750.520D0,
      *6356774.719D0,6356256.909D0,6356075.413D0,6356794.343D0,
      *6356784.284D0,2*6371000.0D0/
C NAMES OF THE VARIOUS ELLIPSOIDS.
      DATA ELLIPTX/'SPHERE','CLARKE 1866','HAYFORD 1909.(INTERNAT.)',
      *'GRS 1980','CLARK1880','BESSEL 1841','KRASOVSKY 1940',
      *'WGS 1972','AUSTRALIAN 1965','AIRY1849','EVEREST 1830',
      *'HOUGH 1956','FISHER 1960','SPHERE','SPHERE','USER DEFINED'/
C
      DATA D0,D1,D2,D3,D4,TRIFLE/0.0D0,1.0D0,2.0D0,3.0D0,4.0D0,.5D-11/
      END
      SUBROUTINE DIRECT(SEMIA,SEMIB,B1,RL1,B2,RL2,FAZ,BAZ,S,N)
C PROGRAMMED BY T.VINCENTY, 1975. REF.: VINCENTY, T.: DIRECT AND
C INVERSE SOLUTIONS OF GEODESICS ON THE ELLIPSOID WITH APPLICATIONS
C OF NESTED EQUATIONS, SURVEY REVIEW, NO. 176, 1975.
C LATITUDE POSITIVE NORTH, LONGITUDE POSITIVE EAST, AZIMUTH CLOCK-
C WISE FROM NORTH. ANGLES IN RADIANS. MODIFIED JULY 1993, BY C.C.TSCHERNING.
C CALL PARAMETERS:
C SEMIA, SEMIB = MAJOR AND MINOR SEMI AXES OF THE ELLIPSOID IN METERS.
C B1, RL1 = LATITUDE, LONGITUDE IN RADIANS OF POINT P.
C FAZ, S = AZIMUTH (RADIANS) AND DISTANCE (METERS) TO A POINT Q.
C RETURN PARAMETERS:
C B2, RL2 = LATITUDE, LONGITUDE IN RADIANS OF Q.
C BAZ = AZIMUTH FROM Q TO P IN RADIANS.
      IMPLICIT REAL*8 (A-H,O-Z)
      COMMON /CONST/D0,D1,D2,D3,D4,RADDEG,DEGRAD,PI,TWOPI,TRIFLE
      EQUIVALENCE (SINU,TANU)
      A = SEMIA
      R = A/(A-SEMIB)
      B0A = SEMIB/SEMIA
      TANU=B0A*SIN(B1)/COS(B1)
      SINAZ = SIN(FAZ)
      COSAZ = COS(FAZ)
      SIG1 = ATAN2(TANU,COSAZ)
      COSU = D1/SQRT(TANU*TANU+D1)
      SINU = TANU*COSU
      SINAL = COSU*SINAZ
      SCOSAL= -SINAL*SINAL+D1
      RK = SQRT((D1/B0A/B0A-D1)*SCOSAL+D1)+D1
      RK = (RK-D2)/RK
      C = (RK*RK/D4+D1)/(D1-RK)
      D = (-0.375D0*RK*RK+D1)*RK
      FIRST = S/B0A/A/C
      SIG = FIRST
      N = 0
1 COSTM = COS(SIG1*D2+SIG)
      N = N+1
      E = COSTM*COSTM*D2-D1
      SINSIG= SIN(SIG)
      COSSIG= COS(SIG)
      TEST = SIG
      SIG = (((-D2*E+D1)*(SINSIG*SINSIG*D4-D3)*COSTM*D/6.0D0
      *+E*COSSIG)*D/D4+COSTM)*SINSIG*D+FIRST
      IF (ABS(SIG-TEST).GT.TRIFLE) GO TO 1
      TEST = SINU*SINSIG-COSU*COSSIG*COSAZ
      B2 = ATAN2(SINU*COSSIG+COSU*SINSIG*COSAZ, B0A*

```

geomain2.f

2/4

Jan 16, 07 14:13

geomain2.f

Page 5/8

```

*      SQRT(SINAL*SINAL+TEST*TEST)
RLAM = ATAN2(SINSIG*SINAZ, COSU*COSSIG-SINU*SINSIG*COSAZ)
C      = ((-D3*SCOSAL+D4)/R+D4)*SCOSAL/R/16.0D0
RL2 = ((E*COSSIG+C+COSTM)*SINSIG*C+SIG)*SINAL*(C-D1)/R+RLAM+RL1
IF (RL2.GT.PI) RL2=RL2-TWOPI
IF (RL2.LT.-PI) RL2=RL2+TWOPI
BAZ = ATAN2(-SINAL,TEST)
IF (BAZ.LT.D0) BAZ=BAZ+TWOPI
RETURN
END
SUBROUTINE INVERS(SEMIA,SEMIB,B1,RL1,B2,RL2,FAZ,BAZ,S,N)
C PROGRAMMED BY T.VINCENTY, 1975. REF.: VINCENTY, T.: DIRECT AND
C INVERSE SOLUTIONS OF GEODESICS ON THE ELLIPSOID WITH APPLICATIONS
C OF NESTED EQUATIONS, SURVEY REVIEW, NO. 176, 1975.
C LATITUDE POSITIVE NORTH, LONGITUDE POSITIVE EAST, AZIMUTH CLOCK-
C FEATURES TO HANDLE ANTIPODAL PROBLEM ADDED.
C WISE FROM NORTH. ANGLES IN RADIANS. MODIFIED JULY 1993, BY C.C.TSCHERNING.
C CALL PARAMETERS:
C SEMIA, SEMIB = MAJOR AND MINOR SEMI AXES OF THE ELLIPSOID IN METERS.
C B1, RL1 = LATITUDE, LONGITUDE IN RADIANS OF POINT P.
C B2, RL2 = LATITUDE, LONGITUDE IN RADIANS OF A POINT Q.
C FAZ, S = AZIMUTH (RADIANS) AND DISTANCE (METERS) TO A POINT Q.
C RETURN PARAMETERS:
C FAZ, S = AZIMUTH (RADIANS) AND DISTANCE (METERS) FROM P TO Q.
C BAZ = AZIMUTH FROM Q TO P IN RADIANS.
IMPLICIT REAL*8 (A-H,O-Z)
COMMON /CONST/D0,D1,D2,D3,D4,DEGRAD,RADDEG,PI,TWOPI,TRIFLE
EQUIVALENCE (RK,RLAM)
A = SEMIA
RF = A/(A-SEMIB)
BOA = SEMIB/SEMIA
RL = RL2-RL1
D = SQRT((BOA*SIN(B1))**2+COS(B1)**2)
SINU1 = BOA*SIN(B1)/D
COSU1 = SQRT(D1-SINU1**2)
D = SQRT((BOA*SIN(B2))**2+COS(B2)**2)
SINU2 = BOA*SIN(B2)/D
COSU2 = SQRT(D1-SINU2**2)
IF (RL.GT.PI) RL=RL-PI-PI
IF (RL.LT.-PI) RL=RL+PI+PI
TEST = RL
PREV = RL
RLAM = RL
N = -1
C KIND=1 IS THE STANDARD SOLUTION.
KIND=1
2 SINLAM= SIN(RLAM)
IF (ABS(PI-ABS(RL)).LT.0.2D-10) SINLAM=D0
COSLAM= COS(RLAM)
8 TEMP = COSU1*SINU2-SINU1*COSU2*COSLAM
SINSIG=SQRT((COSU2*SINLAM)**2+TEMP**2)
COSSIG=SINU1*SINU2+COSU1*COSU2*COSLAM
SIG =ATAN2(SINSIG,COSSIG)
C
GO TO (3,5), KIND
3 SINAL = COSU1*COSU2*SINLAM/SINSIG
SCOSAL = -SINAL**2+D1
4 IF (ABS(SCOSAL).LT.TRIFLE) THEN
COSTM = -D2*SINU1*SINU2/(SCOSAL+TRIFLE)+COSSIG
ELSE
COSTM = -D2*SINU1*SINU2/SCOSAL+COSSIG
END IF
N=N+1
IF (KIND.EQ.2) N=N-2
GO TO (5,6), KIND
5 C = ((-D3*SCOSAL+D4)/RF+D4)*SCOSAL/RF/16.0D0
GO TO (15,16),KIND
16 IF (MOD(N,50).EQ.0.AND.N.LT.-2)
*WRITE(*,*) ' SINAL,PREV= ',SINAL,PREV,N

```

Jan 16, 07 14:13

geomain2.f

Page 6/8

```

IF (ABS(SINAL-PREV).LT.TRIFLE*1000.0.OR.N.LT.-5000) GO TO 100
15 GO TO (6,4), KIND
6 D=((D2*COSTM**2-D1)*COSSIG*C+COSTM)*SINSIG*C+SIG*(D1-C)/RF
GO TO (7,9),KIND
7 RLAM = RL+D*SINAL
IF (ABS(RLAM-TEST).LT.TRIFLE) GO TO 100
IF (ABS(RLAM).GT.PI) GO TO 200
IF (N.GT.4.AND.(RLAM-TEST)*(TEST-PREV).LT.D0.AND.
* ABS(RLAM-TEST).LT.ABS(TEST-PREV)) RLAM = (RLAM+TEST)/D2
PREV = TEST
TEST = RLAM
GO TO 2
C ANTIPODAL SOLUTION.
200 KIND = 2
N = 1
RLAM = PI
IF (RL.LT.d0) RLAM=-RLAM
SINAL = D0
SCOSAL= D1
TEST = D1
PREV = TEST
SIG = PI- ABS(B1+B2)
SINSIG= SIN(SIG)
COSSIG= COS(SIG)
GO TO 5
C
9 SINAL = (RLAM-RL)/D
IF (N.LT.-4.AND.(SINAL-TEST)*(TEST-PREV).LT.D0.AND.
. ABS(SINAL-TEST).LT.ABS(TEST-PREV)) SINAL = (SINAL+TEST)/D2
PREV = TEST
TEST = SINAL
SCOSAL= -SINAL**2+d1
SINLAM= SINAL*SINSIG/(COSU1*COSU2)
COSLAM= -sqrt(ABS(-SINLAM**2+D1))
RLAM = ATAN2(SINLAM,COSLAM)
GO TO 8
100 IF (KIND.EQ.2) GO TO 11
FAZ = ATAN2(COSU2*SINLAM,COSU1*SINU2-SINU1*COSU2*COSLAM)
BAZ = ATAN2(-COSU1*SINLAM,SINU1*COSU2-COSU1*SINU2*COSLAM)
GO TO 12
11 FAZ = SINAL/COSU1
BAZ = SQRT(-FAZ**2+d1)
IF (TEMP.LT.D0) BAZ=-BAZ
FAZ = ATAN2(FAZ,BAZ)
BAZ = ATAN2(-SINAL,SINU1*SINSIG-COSU1*COSSIG*BAZ)
12 IF (FAZ.LT.D0) FAZ=FAZ+TWOPI
IF (BAZ.LT.D0) BAZ=BAZ+TWOPI
E = COSTM**2*D2-D1
RK = SQRT((D1/BOA**2-D1)*SCOSAL+D1)+D1
RK = (RK-D2)/RK
C = (RK**2/D4+D1)/(D1-RK)
D = (-0.375D0*RK**2+D1)*RK
S = (((-D2*E+D1)*(SINSIG**2*D4-D3)*COSTM*D/6.0D0+E*COSSIG)
* D/D4+COSTM)*SINSIG*(-D)+SIG)*C*A*BOA
IF (N.LT.-5000) THEN
WRITE(*,*) ' **** WARNING ****, N= ', N
IF (KIND.EQ.1) N = IABS(N)
END IF
RETURN
END
SUBROUTINE DDEG(IDEG,MIN,SEC,RA,IANG)
C THE SUBROUTINE CONVERTS FOR IANG = 1,2,3,4 ANGLES IN (1) DEGREES, MI-
C NUTES, SECONDS, (2) DEGREES, MINUTES, (3) DEGREES AND (4) 400-DEGREES
C TO DECIMAL RADIANS.
C IF DOUBLE PRECISION IS NEEDED, ACTIVATE:
IMPLICIT INTEGER(I,J,K,M,N),REAL *8(A-H,O-Z)
COMMON /CONST/D0,D1,D2,D3,D4,RADDEG,DEGRAD,PI,TWOPI,TRIFLE
I = 1
IF (IDEG .LT. 0 .AND. IANG .LT. 3) I = -1

```

