

```

//00000000 JOB (000000000),SNYDER,CLASS=B,MSGLEVEL=(0,0)
/*JOBPARM T=2,L=5
/*ROUTE PRINT RMT14
// EXEC FORTXCG,REGION.FORT=400K,TIME.GO=(,15),REGION.GO=400K
//FORT.SYSIN DD *
C
C THIS PROGRAM DETERMINES COEFFICIENTS FOR A CONFORMAL MAP
C PROJECTION WITH MINIMUM SCALE ERROR THROUGH A GIVEN NUMBER
C OF POINTS.
C
C IMPLICIT REAL*8(A-H,O-Z)
C DIMENSION PHI(168),CLON(168),A(15),B(15),DKDA(15),DKDB(15)
C DIMENSION FA(15),FB(15),DFADA(15,15),DFBDB(15,15),DFADB(15,15)
C DIMENSION C(500),D(30),DBEQ(1,30),WT(168)
C DIMENSION DFADP(15),DFADL(15),DFADN(15),DFBDP(15),DFBDL(15),DFBDN(
*15)
C DIMENSION DF1A(15),DF2A(15),CHI(168)
C
C READ NUMBER OF POINTS M (MAX=168), NUMBER OF COEFFICIENTS N
C (MAX=15) OF A AND NUMBER N OF B, AND LAT/LONG OF FIXED POINTS.
C
C READ (5,5) M,E2,CHIR,CLONR
5 FORMAT (I3,7X,3F10.8)
PI=3.14159265359
DGRAD=PI/180.00
ECC=DSQRT(E2)
E4=E2*E2
E6=E4*E2
WRITE (6,6) M
6 FORMAT (I1,2X,'CONFORMAL MAP PROJECTION COEFFICIENTS CALCULATION'
*(I10,2X,'LAT./LONG. AND WTS. OF ',I3,' POINTS FOR MINIMUM SCALE E
*RROR'/))
CHIR=CHIR*DGRAD
CLONR=CLONR*DGRAD
DO 20 I=1,M
READ (5,10) PHI(I),CLON(I),WT(I)
10 FORMAT (F4.1,F6.1,F10.5)
WRITE (6,15) I,PHI(I),CLON(I),WT(I)
15 FORMAT (I1,2X,I3,'.',2F10.1,F14.5)
PHI(I)=PHI(I)*DGRAD
CLON(I)=CLON(I)*DGRAD
ES=DSIN(PHI(I))*ECC
CHI(I)=2.00*DATAN(DTAN(PI/4.00+PHI(I)/2.00)*((1.00-ES)/(1.00+ES))
**((ECC/2.00))-PI/2.00
20 CONTINUE
22 CONTINUE
C
C IF KODE IS 0, CONE CONSTANT EN CAN VARY. IF KODE IS NOT 0, CONE
C CONSTANT IS FIXED.
C
C IF KODE1 IS 0, POLE CAN VARY. IF KODE1 IS 1, LONG. (CLONP) ONLY
C OF POLE CAN VARY. IF KODE1 IS 2, LAT. (PHIP)/LONG. OF POLE IS
C FIXED.
C
C IF KODE2 IS 0, GRATICULE WILL BE PRINTED. IF KODE2 IS NOT 0,
C SCALE FACTORS OF THE M POINTS WILL BE PRINTED AFTER COMPUTATION
C OF CONSTANTS.
C ENTER THE INITIAL ESTIMATE (OR FINAL VALUE) OF PHIP AND CLONP IN
C DEG. AND DECIMALS, WITH OTHER VALUES ON CARD. THE CONE CONSTANT

```

```

00000010
00000020
00000030
00000040
00000050
00000060
00000070
00000080
00000090
00000100
00000110
00000120
00000130
00000140
00000150
00000160
00000170
00000180
00000190
00000200
00000210
00000220
00000230
00000240
00000250
00000260
00000270
00000280
00000290
00000300
00000310
00000320
00000330
00000340
00000350
00000360
00000370
00000380
00000390
00000400
00000410
00000420
00000430
00000440
00000450
00000460
00000470
00000480
00000490
00000500
00000510
00000520
00000530
00000540
00000550
00000560
00000570
00000580
00000590

```

```

C      WITH KODE NOT EQUAL TO 0 WILL ESTABLISH WHETHER THE PROJECTION
C      IS OBLIQUE MERCATOR OR STEREOGRAPHIC.
C
25  READ (5,25) N,PHIP,CLONP,EN,KODE,KODE1,KODE2
    FORMAT (I2,8X,3F10.6,3I2)
    IF (N.EQ.0) GO TO 510
    WRITE (6,30) PHIP,CLONP,EN,KODE,KODE1
30  FORMAT (1H0,2X,'INIT. EST. LAT./LONG. OF TRANSF. POLE FOR OBLIQUE
*ONF. CONIC, TRANS. OR OBLIQUE MERCATOR (IF CONE CONST. IS 0), OBLI
*QUE STEREOG.'/(1H,2X,'(IF CONE CONSTANT IS 1):',2F10.5,' CONE CON
*ST.:',F10.7,' CODE=',1I,' CODE1=',1I))
    WRITE (6,32)
32  FORMAT (1H0,2X,'IF CODE IS NOT 0, CONE CONST. IS FIXED. IF CODE
* IS 1, LAT. OF POLE IS FIXED. IF CODE1 IS 2, LAT/LONG OF POLE
* IS FIXED.')
    PHIP=PHIP*DGRAD
    ES=DSIN(PHIP)*ECC
    CHIP=2.00*DATAN(DTAN(P1/4.00+PHIP/2.00)*((1.00-ES)/(1.00+ES))*
*(ECC/2.00))-PI/2.00
    CLONP=CLONP*DGRAD
    IF (KODE.EQ.0.AND.KODE1.EQ.0) N4=N+N+2
    IF (KODE.NE.0.AND.KODE1.EQ.0) N4=N+N+1
    IF (KODE.EQ.0.AND.KODE1.EQ.1) N4=N+N+1
    IF (KODE.NE.0.AND.KODE1.EQ.1) N4=N+N
    IF (KODE.EQ.0.AND.KODE1.EQ.2) N4=N+N
    IF (KODE.NE.0.AND.KODE1.EQ.2) N4=N+N-1
    N1=N+1
    N2=N+N-1
    N3=N4+1
    N5=N+N
    IF (KODE1.EQ.1) N6=N+N
    IF (KODE1.EQ.0) N6=N+N+1
    NN=N-1

C      TLAT IS TOP LAT, BLAT IS BOTTOM LAT, WLONG IS WESTMOST LONG, ELONG
C      IS EASTMOST LONG, DLAT IS LAT INTERVAL AND DLONG IS LONG INTERVAL
C      FOR GRATICULE IN DEGREES.
C
    READ (5,35) TLAT,BLAT,WLONG,ELONG,DLAT,DLONG
35  FORMAT (10X,6F10.3)

C      INITIALIZE COEFFICIENTS: A(1)=1 AND THE REST ARE 0.
C
    DO 40 I=1,N
    A(I)=0.00
    B(I)=0.00
40  CONTINUE
    A(1)=1.00
    NE=0
    IF (N.EQ.1) GO TO 418
    IF (KODE.EQ.0.AND.KODE1.EQ.0) WRITE (6,41) N,N
41  FORMAT (1H0,2X,'SUCCESSIVE ADJUSTMENTS TO COEFFICIENTS A(1)-A(',1200001110
*,')', B(2)-B(',12,')', LAT/LONG OF POLE, AND CONE CONST.:')
    IF (KODE.EQ.0.AND.KODE1.EQ.1) WRITE (6,421) N,N
421 FORMAT (1H0,2X,'SUCCESSIVE ADJUSTMENTS TO COEFFICIENTS A(1)-A(',1200001140
*,')', B(2)-B(',12,')', LONG OF POLE, AND CONE CONST.:')
    IF (KODE.EQ.0.AND.KODE1.EQ.2) WRITE (6,411) N,N
411 FORMAT (1H0,2X,'SUCCESSIVE ADJUSTMENTS TO COEFFICIENTS A(1)-A(',1200001170
*,')', B(2)-B(',12,')', AND CONE CONST.:')

```

```

IF (KODE.NE.0.AND.KODE1.EQ.0) WRITE (6,412) N,N
412 FORMAT (1H0,2X,'SUCCESSIVE ADJUSTMENTS TO COEFFICIENTS A(1)-A(',1200001200
* ' ', B(2)-B(' ',12,' '), AND LAT/LONG OF POLE: ') 00001210
IF (KODE.NE.0.AND.KODE1.EQ.1) WRITE (6,422) N,N 00001220
422 FORMAT (1H0,2X,'SUCCESSIVE ADJUSTMENTS TO COEFFICIENTS A(1)-A(',1200001230
* ' ', B(2)-B(' ',12,' '), AND LONG OF POLE: ') 00001240
IF (KODE.NE.0.AND.KODE1.EQ.2) WRITE (6,413) N,N 00001250
413 FORMAT (1H0,2X,'SUCCESSIVE ADJUSTMENTS TO COEFFICIENTS A(1)-A(',1200001260
* ' ') AND B(2)-B(' ',12,' '): ') 00001270
GO TO 419 00001280
418 CONTINUE 00001290
IF (KODE.EQ.0.AND.KODE1.EQ.0) WRITE (6,414) N 00001300
414 FORMAT (1H0,2X,'SUCCESSIVE ADJUSTMENTS TO COEFFICIENTS A(1)-A(',1200001310
* ' ') LAT/LONG OF POLE, AND CONE CONST.: ') 00001320
IF (KODE.EQ.0.AND.KODE1.EQ.1) WRITE (6,423) N 00001330
423 FORMAT (1H0,2X,'SUCCESSIVE ADJUSTMENTS TO COEFFICIENTS A(1)-A(',1200001340
* ' '), LONG OF POLE, AND CONE CONST.: ') 00001350
IF (KODE.EQ.0.AND.KODE1.EQ.2) WRITE (6,415) N 00001360
415 FORMAT (1H0,2X,'SUCCESSIVE ADJUSTMENTS TO COEFFICIENTS A(1)-A(',1200001370
* ' ') AND CONE CONST.: ') 00001380
IF (KODE.NE.0.AND.KODE1.EQ.0) WRITE (6,416) N 00001390
416 FORMAT (1H0,2X,'SUCCESSIVE ADJUSTMENTS TO COEFFICIENTS A(1)-A(',1200001400
* ' ') AND LAT/LONG OF POLE: ') 00001410
IF (KODE.NE.0.AND.KODE1.EQ.1) WRITE (6,424) N 00001420
424 FORMAT (1H0,2X,'SUCCESSIVE ADJUSTMENTS TO COEFFICIENTS A(1)-A(',1200001430
* ' ') AND LONG OF POLE: ') 00001440
IF (KODE.NE.0.AND.KODE1.EQ.2) WRITE (6,417) N 00001450
417 FORMAT (1H0,2X,'SUCCESSIVE ADJUSTMENTS TO COEFFICIENTS A(1)-A(',1200001460
* ' '): ') 00001470
419 CONTINUE 00001480
42 CONTINUE 00001490
SP=DSIN(CHIP) 00001500
CP=DCOS(CHIP) 00001510
IF (DABS(EN).LT.1.D-7) GO TO 43 00001520
IF (DABS(EN).GT.1.D-7) GO TO 432 00001530
00001540
CC COMPUTE COORDINATES OF CENTER FOR OBLIQUE CONFORMAL CONIC. 00001550
00001560
SL=DSIN(CLONR-CLONP) 00001580
CL=DCOS(CLONR-CLONP) 00001590
SPP=DSIN(CHIR) 00001600
CPP=DCOS(CHIR) 00001610
CZ=SP*SPP+CP*CPP*CL 00001620
FZ=DSQRT(1.D-CZ**2) 00001630
F=DSQRT((1.D+EN)**(1.D+EN)*(1.D-EN)**(1.D-EN))/EN 00001640
RHO=F*((1.D-CZ)/(1.D+CZ))*(EN/2.D) 00001650
THP=EN*DATAN2(CPP*SL,CP*SPP-SP*CPP*CL) 00001660
XPO=-RHO*DSIN(THP) 00001670
YPO=-RHO*DCOS(THP) 00001680
GO TO 431 00001690
43 CONTINUE 00001700
CC COMPUTE COORDINATES OF CENTER FOR OBLIQUE MERCATOR. 00001710
00001720
CLONO=CLONP+90.D*DGRAD 00001730
SL=DSIN(CLONR-CLONO) 00001740
CL=DCOS(CLONR-CLONO) 00001750
SPP=DSIN(CHIR) 00001760
CPP=DCOS(CHIR) 00001770

```

```

XPO=DATAN2(SPP*CP+CPP*SP*SL, CPP*CL)
CZ=SP*SPP-CP*CPP*SL
YPO=DLOG((1.D0+CZ)/(1.D0-CZ))/2.D0
GO TO 431
432 CONTINUE
C
C   COMPUTE COORDINATES OF CENTER FOR OBLIQUE STEREOGRAPHIC.
C
XPO=0.D0
YPO=0.D0
431 CONTINUE
SE=0.D0
SWT=0.D0
J=(N4*N4+3*N4)/2
DO 44 I=1,J
C(I)=0.D0
44 CONTINUE
DO 45 I=1,N
FA(I)=0.D0
FB(I)=0.D0
DFADP(I)=0.D0
DFADL(I)=0.D0
DFADN(I)=0.D0
DFBDP(I)=0.D0
DFBDL(I)=0.D0
DFBDN(I)=0.D0
DO 45 J=1,N
DFADA(I,J)=0.D0
DFBDB(I,J)=0.D0
DFADB(I,J)=0.D0
45 CONTINUE
FCHIP=0.D0
FLATP=0.D0
FN=0.D0
DFPDP=0.D0
DFPDL=0.D0
DFPDN=0.D0
DFLDL=0.D0
DFLDN=0.D0
DFNDN=0.D0
C
C   CALCULATE POLAR COORDINATES OF THE M POINTS BASED ON THE OBLIQUE
C   CONF. CONIC PROJECTION.
C
DO 80 I=1,M
IF (DABS(EN).LT.1.D-7) GO TO 453
IF (DABS(EN).GT.1.D0-1.D-7) GO TO 451
C
C   COMPUTE FUNCTIONS FOR OBLIQUE CONFORMAL CONIC.
C
SL=DSIN(CLON(I)-CLONP)
CL=DCOS(CLON(I)-CLONP)
SPP=DSIN(CHI(I))
CPP=DCOS(CHI(I))
CZ=SP*SPP+CP*CPP*CL
SZ=DSQRT(1.D0-CZ**2)
RHO=F*((1.D0-CZ)/(1.D0+CZ))* (EN/2.D0)
C
C   SKPR IS SCALE FACTOR ON OBLIQUE CONFORMAL CONIC BASE.

```

```

00001780
00001790
00001800
00001810
00001820
00001830
00001840
00001850
00001860
00001870
00001880
00001890
00001900
00001910
00001920
00001930
00001940
00001950
00001960
00001970
00001980
00001990
00002000
00002010
00002020
00002030
00002040
00002050
00002060
00002070
00002080
00002090
00002100
00002110
00002120
00002130
00002140
00002150
00002160
00002170
00002180
00002190
00002200
00002210
00002220
00002230
00002240
00002250
00002260
00002270
00002280
00002290
00002300
00002310
00002320
00002330
00002340
00002350
00002360

```



```

SZ=DSQRT(1.D0-CZ**2)
S=2.D0/(1.D0+CZ)
SKPR=S*CPP/(DCOS(PHI(1))/DSQRT(1.D0-E2*DSIN(PHI(1))**2))
XP=S*CPP*SL
YP=S*(CP*SPP-SP*CPP*CL)
FAC=-YP/2.D0
CC=XP*FAC
DD=-S*CZ+YP*FAC
DKPDP=SKPR*FAC
FAC=-XP*CP/2.D0
EE=-S*CPP*CL+XP*FAC
GG=-XP*SP+YP*FAC
DKPDL=SKPR*FAC
452 CONTINUE
XP=XP-XPO
YP=YP-YPO
R=DSQRT(XP**2+YP**2)
IF (R.LT.1.D-9) GO TO 46
TH=DATAN2(YP,XP)
GO TO 47
46 TH=0.D0
47 CONTINUE
F2=0.D0
F1=0.D0
DF1P=0.D0
DF1L=0.D0
DF1N=0.D0
DF2P=0.D0
DF2L=0.D0
DF2N=0.D0
C IF (1.LE.30.AND.NE.EQ.0) WRITE (6,92) CC,DD,EE,GG,HH,HJ,XPO,YPO,RH
C *0,R,TH
DO 50 J=1,N
STH=DSIN(DFLOAT(J-1)*TH)
CTH=DCOS(DFLOAT(J-1)*TH)
STH2=DSIN(DFLOAT(J-2)*TH)
CTH2=DCOS(DFLOAT(J-2)*TH)
IF (R.LT.1.D-7) GO TO 48
RPOW=R*(J-1)
RPOW2=R*(J-2)
GO TO 49
48 IF (J.EQ.1) RPOW=1.D0
IF (J.NE.1) RPOW=0.D0
IF (J.EQ.2) RPOW2=1.D0
IF (J.NE.2) RPOW2=0.D0
49 CONTINUE
FAC=DFLOAT(J)*RPOW
FAC2=DFLOAT(J*(J-1))*RPOW2
F2=F2+FAC*(A(J)*CTH-B(J)*STH)
F1=F1+FAC*(A(J)*STH+B(J)*CTH)
IF (KODE1.EQ.2) GO TO 482
DF1P=DF1P+FAC2*((A(J)*CC-B(J)*DD)*STH2+(A(J)*DD+B(J)*CC)*CTH2)
DF2P=DF2P+FAC2*((A(J)*CC-B(J)*DD)*CTH2-(A(J)*DD+B(J)*CC)*STH2)
DF1L=DF1L+FAC2*((A(J)*EE-B(J)*GG)*STH2+(A(J)*GG+B(J)*EE)*CTH2)
DF2L=DF2L+FAC2*((A(J)*EE-B(J)*GG)*CTH2-(A(J)*GG+B(J)*EE)*STH2)
482 CONTINUE
IF (KODE.NE.0) GO TO 491
DF1N=DF1N+FAC2*((A(J)*HH-B(J)*HJ)*STH2+(A(J)*HJ+B(J)*HH)*CTH2)
DF2N=DF2N+FAC2*((A(J)*HH-B(J)*HJ)*CTH2-(A(J)*HJ+B(J)*HH)*STH2)

```

```

00002960
00002970
00002980
00002990
00003000
00003010
00003020
00003030
00003040
00003050
00003060
00003070
00003080
00003090
00003100
00003110
00003120
00003130
00003140
00003150
00003160
00003170
00003180
00003190
00003200
00003210
00003220
00003230
00003240
00003250
00003260
00003270
00003280
00003290
00003300
00003310
00003320
00003330
00003340
00003350
00003360
00003370
00003380
00003390
00003400
00003410
00003420
00003430
00003440
00003450
00003460
00003470
00003480
00003490
00003500
00003510
00003520
00003530
00003540

```

```

491 CONTINUE                                00003550
    DF1A(J)=FAC*STH                        00003560
    DF2A(J)=FAC*CTH                        00003570
50 CONTINUE                                00003580
    SK=DSQRT(F2**2+F1**2)*SKPR             00003590
    IF (1.LE.30.AND.NE.EQ.0) WRITE (6,92) SK,SKPR,DF1P,DF2P,DF1L,DF2L, 00003600
    *F1,F2                                00003610
    SK IS SCALE FACTOR ON TRANSFORMED MAP PROJECTION. 00003620
    SE=SE+WT(I)*(SK-1.00)**2              00003630
    SWT=SWT+WT(I)                          00003640
    DO 60 J=1,N                            00003650
    DKDA(J)=(SKPR**2/SK)*(F1*DF1A(J)+F2*DF2A(J)) 00003660
    DKDB(J)=(SKPR**2/SK)*(F1*DF2A(J)-F2*DF1A(J)) 00003670
    FA(J)=FA(J)+WT(I)*(SK-1.00)*DKDA(J)    00003680
    FB(J)=FB(J)+WT(I)*(SK-1.00)*DKDB(J)    00003690
    IF (1.LE.30.AND.NE.EQ.0) WRITE (6,69) J,DKDA(J),DKDB(J) 00003700
    69 FORMAT (1H ,2X,I3,2F13.7)          00003710
    60 CONTINUE                            00003720
    DKDP=(SKPR**2/SK)*(F1*DF1P+F2*DF2P)+(SK/SKPR)*DKPDL 00003730
    DKDL=(SKPR**2/SK)*(F1*DF1L+F2*DF2L)+(SK/SKPR)*DKPDN 00003740
    IF (KODE.EQ.0) DKDN=(SKPR**2/SK)*(F1*DF1N+F2*DF2N)+(SK/SKPR)*DKPDN 00003750
    FCHIP=FCHIP+WT(I)*(SK-1.00)*DKDP      00003760
    IF (1.LE.30.AND.NE.EQ.0) WRITE (6,92) DKDP,DKDL,DKDN 00003770
    FLATP=FLATP+WT(I)*(SK-1.00)*DKDL      00003780
    IF (KODE.EQ.0) FN=FN+WT(I)*(SK-1.00)*DKDN 00003790
    IF (1.LE.30.AND.NE.EQ.0) WRITE (6,92) DKDP,DKDL,DKDN,DKDA(1), 00003800
    *DKDP,DKDL,DKDN                      00003810
    A FACTOR OF 2 IS REMOVED FROM ALL MULTIPLIERS IN THE 00003820
    SIMULTANEOUS EQUATIONS.                00003830
    IF (KODE1.EQ.0) DFPDP=DFPDP+WT(I)*DKDP**2 00003840
    IF (KODE1.LE.1) DFPDL=DFPDL+WT(I)*DKDP*DKDL 00003850
    IF (KODE.EQ.0) DFPDN=DFPDN+WT(I)*DKDP*DKDN 00003860
    IF (KODE1.LE.1) DFLDL=DFLDL+WT(I)*DKDL**2 00003870
    IF (KODE.EQ.0) DFLDN=DFLDN+WT(I)*DKDL*DKDN 00003880
    IF (KODE.EQ.0) DFNDN=DFNDN+WT(I)*DKDN**2 00003890
    DO 80 J=1,N                            00003900
    IF (KODE1.EQ.0) DFADP(J)=DFADP(J)+WT(I)*DKDP*DKDA(J) 00003910
    IF (KODE1.EQ.0) DFBDP(J)=DFBDP(J)+WT(I)*DKDP*DKDB(J) 00003920
    IF (KODE1.LE.1) DFADL(J)=DFADL(J)+WT(I)*DKDL*DKDA(J) 00003930
    IF (KODE1.LE.1) DFBDL(J)=DFBDL(J)+WT(I)*DKDL*DKDB(J) 00003940
    IF (KODE.EQ.0) DFADN(J)=DFADN(J)+WT(I)*DKDN*DKDA(J) 00003950
    IF (KODE.EQ.0) DFBDN(J)=DFBDN(J)+WT(I)*DKDN*DKDB(J) 00003960
    DO 80 K=1,N                            00003970
    DFADA(J,K)=DFADA(J,K)+WT(I)*DKDA(J)*DKDA(K) 00003980
    DFBDB(J,K)=DFBDB(J,K)+WT(I)*DKDB(J)*DKDB(K) 00003990
    DFADB(J,K)=DFADB(J,K)+WT(I)*DKDA(J)*DKDB(K) 00004000
    80 CONTINUE                            00004010
    NOTE: DFBDA(J,K) IS NOT STORED SEPARATELY, SINCE DFBDA(J,K)= 00004020
    DFADB(K,J).                            00004030
    DO 95 I=1,N                            00004040
    OBEQ (1,M5)=-FA(I)                    00004050
    DO 85 J=1,N                            00004060
    OBEQ(1,J)=DFADA(1,J)                  00004070
    85 CONTINUE                            00004080
    95 CONTINUE                            00004090
    00004100
    00004110
    00004120
    00004130

```

```
85 CONTINUE
  IF (N.EQ.1) GO TO 91
  DO 90 J=1,NM
    OBEQ(1,(J+N))=DFADB(I,(J+1))
90 CONTINUE
91 CONTINUE
  IF (KODE1.EQ.0) OBEQ(1,M5)=DFADP(I)
  IF (KODE1.LE.1) OBEQ(1,M6)=DFADL(I)
  IF (KODE.EQ.0) OBEQ(1,M4)=DFADN(I)
C   WRITE (6,92) (OBEQ(1,J),J=1,16)
C   IF (M3.GT.16) WRITE (6,93) (OBEQ(1,J),J=17,M3)
C 92 FORMAT (1H0,2X,'OBEQ=',8F13.7/(1H,2X,8F13.7))
C 93 FORMAT (1H,2X,8F13.7/(1H,2X,8F13.7))
  CALL NRME (OBEQ,1,M4,0,C,M4,1,0)
95 CONTINUE
  K=1
  IF (N.EQ.1) GO TO 111
  DO 110 I=M1,M2
    K=K+1
    OBEQ(1,M3)=-FB(K)
    DO 100 J=1,N
      OBEQ(1,J)=DFADB(J,K)
100 CONTINUE
    DO 105 J=1,NM
      OBEQ(1,(J+N))=DFBDB(K,(J+1))
105 CONTINUE
    IF (KODE1.EQ.0) OBEQ(1,M5)=DFBDP(K)
    IF (KODE1.LE.1) OBEQ(1,M6)=DFBDL(K)
    IF (KODE.EQ.0) OBEQ(1,M4)=DFBDN(K)
C   WRITE (6,92) (OBEQ(1,J),J=1,16)
C   IF (M3.GT.16) WRITE (6,93) (OBEQ(1,J),J=17,M3)
  CALL NRME (OBEQ,1,M4,0,C,M4,1,0)
110 CONTINUE
111 CONTINUE
  IF (KODE1.EQ.2) GO TO 119
  IF (KODE1.EQ.1) GO TO 121
  OBEQ(1,M3)=-FCHIP
  DO 112 J=1,N
    OBEQ(1,J)=DFADP(J)
112 CONTINUE
  IF (N.EQ.1) GO TO 108
  DO 113 J=1,NM
    OBEQ(1,J+N)=DFBDP(J+1)
113 CONTINUE
108 CONTINUE
  OBEQ(1,M5)=DFPDP
  OBEQ(1,M6)=DFPOL
  IF (KODE.EQ.0) OBEQ(1,M4)=DFPDN
C   WRITE (6,92) (OBEQ(1,J),J=1,16)
C   IF (M3.GT.16) WRITE (6,93) (OBEQ(1,J),J=17,M3)
  CALL NRME (OBEQ,1,M4,0,C,M4,1,0)
121 CONTINUE
  OBEQ(1,M3)=-FLATP
  DO 114 J=1,N
    OBEQ(1,J)=DFADL(J)
114 CONTINUE
  IF (N.EQ.1) GO TO 109
  DO 115 J=1,NM
    OBEQ(1,J+N)=DFBDL(J+1)
```

```
00004140
00004150
00004160
00004170
00004180
00004190
00004200
00004210
00004220
00004230
00004240
00004250
00004260
00004270
00004280
00004290
00004300
00004310
00004320
00004330
00004340
00004350
00004360
00004370
00004380
00004390
00004400
00004410
00004420
00004430
00004440
00004450
00004460
00004470
00004480
00004490
00004500
00004510
00004520
00004530
00004540
00004550
00004560
00004570
00004580
00004590
00004600
00004610
00004620
00004630
00004640
00004650
00004660
00004670
00004680
00004690
00004700
00004710
00004720
```

```

115 CONTINUE                                00004730
109 CONTINUE                                00004740
    IF (KODE1.EQ.0) OBEQ(1,N5)=DFPDL        00004750
    OBEQ(1,N6)=DFLDL                        00004760
    IF (KODE.EQ.0) OBEQ(1,N4)=DFLDN        00004770
C    WRITE (6,92) (OBEQ(1,J),J=1,16)      00004780
C    IF (N3.GT.16) WRITE (6,93) (OBEQ(1,J),J=17,N3) 00004790
    CALL NRME (OBEQ,1,N4,0,C,N4,1,0)      00004800
119 CONTINUE                                00004810
    IF (KODE.NE.0) GO TO 120                00004820
    OBEQ(1,N3)=-FM                          00004830
    DO 116 J=1,N                            00004840
    OBEQ(1,J)=BFADN(J)                      00004850
116 CONTINUE                                00004860
    IF (N.EQ.1) GO TO 118                  00004870
    DO 117 J=1,NM                           00004880
    OBEQ(1,J+N)=DFBDN(J+1)                 00004890
117 CONTINUE                                00004900
118 CONTINUE                                00004910
    IF (KODE1.EQ.0) OBEQ(1,N5)=DFPDM        00004920
    IF (KODE1.LE.1) OBEQ(1,N6)=DFLDN        00004930
    OBEQ(1,N4)=DFNDN                        00004940
C    WRITE (6,92) (OBEQ(1,J),J=1,16)      00004950
C    IF (N3.GT.16) WRITE (6,93) (OBEQ(1,J),J=17,N3) 00004960
    CALL NRME (OBEQ,1,N4,0,C,N4,1,0)      00004970
120 CONTINUE                                00004980
    CALL SMLE (C,N4,D,IERR)                 00004990
    IF (IERR.NE.0) WRITE (6,130) IERR       00005000
    IF (IERR.NE.0) GO TO 22                 00005010
130 FORMAT (1H0,3X,'IERR=',I6,' CANNOT SOLVE USING THESE CONDITIONS.' 00005020
*)                                           00005030
    E=0.D0                                  00005040
    IF (N4.LE.16) WRITE (6,132) (D(I),I=1,N4) 00005050
    IF (N4.GT.16) WRITE (6,132) (D(I),I=1,16) 00005060
    IF (N4.GT.16) WRITE (6,135) (D(I),I=17,N4) 00005070
132 FORMAT (1H0,3X,8F13.7/(1H,3X,8F13.7)) 00005080
135 FORMAT (1H,3X,8F13.7/(1H,3X,8F13.7)) 00005090
    DO 140 I=1,N4                          00005100
    E=E+DABS(D(I))                         00005110
140 CONTINUE                                00005120
    DO 150 I=1,N                           00005130
    A(I)=A(I)+D(I)                          00005140
150 CONTINUE                                00005150
    IF (N.EQ.1) GO TO 165                  00005160
    DO 160 I=2,N                           00005170
    B(I)=B(I)+D(I+N-1)                     00005180
160 CONTINUE                                00005190
165 CONTINUE                                00005200
    IF (KODE1.EQ.0) CHIP=CHIP+D(N5)         00005210
    IF (KODE1.LE.1) CLONP=CLONP+D(N6)       00005220
    IF (KODE.EQ.0) EN=EN+D(N4)              00005230
    NE=NE+1                                00005240
    IF (NE.GT.20) GO TO 480                 00005250
    IF (E.LT.1.D-5) GO TO 170              00005260
    GO TO 42                                00005270
170 PHIP=CHIP*(E2/2.D0+5.D0+E4/24.D0+E6/12.D0)*DSIN(2.D0*CHIP)+(7.D0+E 00005280
*4/48.D0+29.D0+E6/240.D0)*DSIN(4.D0*CHIP)+7.D0+E6*DSIN(6.D0*CHIP)/ 00005290
*120.D0                                     00005300
    PHIPD=PHIP/DGRAD                       00005310

```

```

      CLONPD=CLONP/DGRAD                                00005320
      IF (N.GT.1) WRITE (6,180) N,M                     00005330
180  FORMAT (1H0,3X,'COEFFICIENTS A(1) THROUGH A(' ,I2,')', FOLLOWED BY 00005340
      *OEFF. B(2) THROUGH B(' ,I2,')', LAT/LONG OF POLE, AND CONE CONST.') 00005350
      IF (N.EQ.1) WRITE (6,182) N                       00005360
182  FORMAT (1H0,3X,'COEFFICIENTS A(1) THROUGH A(' ,I2,')', FOLLOWED BY 00005370
      *AT/LONG OF POLE, AND CONE CONST.')               00005380
      WRITE (6,190) (A(I),I=1,M)                        00005390
      IF (N.GT.1) WRITE (6,190) (B(I),I=2,M)            00005400
      WRITE (6,192) PHIPD,CLONPD,EN                     00005410
190  FORMAT (1H0,2X,9F13.9/(1H ,2X,9F13.9))            00005420
192  FORMAT (1H0,2X,3F13.5)                             00005430
      CHID=CHIR/DGRAD                                    00005440
      CLOND=CLONR/DGRAD                                 00005450
      IF (DABS(EN).GT.1.D0-1.D-7) CHID=CHIP/DGRAD      00005460
      IF (DABS(EN).GT.1.D0-1.D-7) CLOND=CLONPD         00005470
      WRITE (6,194) CHID,CLOND                          00005480
194  FORMAT (1H0,3X,'COEFFICIENTS ARE BASED ON USING ' ,F13.5,      00005490
      *' AS CONFORMAL LAT. AND ' ,F13.5,' AS LONG. OF ORIGIN OF BASE PROJE00005500
      *CTION.')                                         00005510
      WRITE (6,195) E2                                   00005520
195  FORMAT (1H0,3X,'SQUARE OF ECCENTRICITY OF REFERENCE ELLIPSOID=' ,F100005530
      *3.8)                                             00005540
      RMS=DSQRT(SE/SMT)                                 00005550
      WRITE (6,200) M,RMS                               00005560
200  FORMAT (1H0,2X,'RMS ERROR IN SCALE FOR ' ,I3,' POINTS SELECTED=' ,      00005570
      *F14.7)                                           00005580
      PH1=TLAT                                          00005590
      CLON1=MLONG                                       00005600
      SP=DSIN(CHIP)                                    00005610
      CP=DCOS(CHIP)                                    00005620
      IF (DABS(EN).LT.1.D-7) GO TO 207                  00005630
      IF (DABS(EN).GT.1.D0-1.D-7) GO TO 208             00005640
C                                                     00005650
C COMPUTE COORDINATES OF CENTER FOR OBLIQUE CONFORMAL CONIC. 00005660
C                                                     00005670
      SL=DSIN(CLONR-CLONP)                             00005680
      CL=DCOS(CLONR-CLONP)                             00005690
      SPP=DSIN(CHIR)                                   00005700
      CPP=DCOS(CHIR)                                   00005710
      CZ=SP*SPP+CP*CPP*CL                              00005720
      SZ=DSQRT(1.D0-CZ**2)                             00005730
      F=DSQRT(((1.D0+EN)**(1.D0+EN))*(1.D0-EN)**(1.D0-EN))/EN 00005740
      RHO=F*((1.D0-CZ)/(1.D0+CZ))**(EN/2.D0)           00005750
      THP=EN*DATAN2(CPP*SL,CP*SPP-SP*CPP*CL)           00005760
      XPO=-RHO*DSIN(THP)                               00005770
      YPO=-RHO*DCOS(THP)                               00005780
      GO TO 209                                          00005790
207 CONTINUE                                           00005800
C                                                     00005810
C COMPUTE COORDINATES OF CENTER FOR OBLIQUE MERCATOR. 00005820
C                                                     00005830
      CLONO=CLONP+90.D0*DGRAD                          00005840
      SL=DSIN(CLONR-CLONO)                             00005850
      CL=DCOS(CLONR-CLONO)                             00005860
      SPP=DSIN(CHIR)                                   00005870
      CPP=DCOS(CHIR)                                   00005880
      XPO=DATAN2(SPP*CP*CPP*SP*SL,CPP*CL)              00005890
      CZ=SP*SPP-CP*CPP*SL                              00005900

```

```

      YPO=DLG((1.D0+CZ)/(1.D0-CZ))/2.D0
      GO TO 209
208 CONTINUE
C
C      COMPUTE COORDINATES OF CENTER FOR OBLIQUE STEREOGRAPHIC.
      XPO=0.D0
      YPO=0.D0
209 CONTINUE
      IF (KODE2.NE.0) WRITE (6,210)
210 FORMAT (1H0,3X,'      LONG.      LAT.      K' /)
      IF (KODE2.EQ.0) WRITE (6,211)
211 FORMAT (1H0,3X,'      LONG.      LAT.      K      X
      *Y' /)
214 CONTINUE
      PHI1=PHI1+DGRAD
      CLON1=CLON1+DGRAD
      IF (KODE2.EQ.0) GO TO 215
      DO 265 I=1,M
215 IF (DABS(EN).LT.1.D-7) GO TO 220
      IF (DABS(EN).GT.1.D-7) GO TO 230
      IF (KODE2.EQ.0) GO TO 212
      CL=DCOS(CLON(I)-CLONP)
      ES=DSIN(PHI(I))*ECC
      CPH=DCOS(PHI(I))
      SPP=DSIN(CHI(I))
      CPP=DCOS(CHI(I))
      SL=DSIN(CLON(I)-CLONP)
      GO TO 213
212 ES=DSIN(PHI1)*ECC
      CPH=DCOS(PHI1)
      CHI1=2.D0*ATAN(DTAN(PHI/4.D0+PHI1/2.D0)*((1.D0-ES)/(1.D0+ES))*
      *(ECC/2.D0))-PI/2.D0
      SL=DSIN(CLON1-CLONP)
      CL=DCOS(CLON1-CLONP)
      SPP=DSIN(CHI1)
      CPP=DCOS(CHI1)
C
C      COMPUTE COORDINATES FOR FINAL OBLIQUE CONFORMAL CONIC.
C
213 CZ=SP+SPP*CP+CPP*CL
      SZ=DSQRT(1.D0-CZ**2)
      RHO=F*((1.D0-CZ)/(1.D0+CZ))**(EN/2.D0)
      THP=EN*ATAN2(CPP*SL,CP*SPP-SP*CPP*CL)
      XP=-RHO*DSIN(THP)-XPO
      YP=-RHO*DCOS(THP)-YPO
      R=DSQRT(XP**2+YP**2)
      SKPR=EN*RHO*CPP*DSQRT(1.D0-ES**2)/SZ/CPH
      GO TO 240
220 CONTINUE
      IF (KODE2.EQ.0) GO TO 222
      SL=DSIN(CLON(I)-CLON0)
      CL=DCOS(CLON(I)-CLON0)
      SPP=DSIN(CHI(I))
      CPP=DCOS(CHI(I))
      ESQ=E2*DSIN(PHI(I))*2
      CPH=DCOS(PHI(I))
      GO TO 223
222 SL=DSIN(CLON1-CLON0)

```

```

00005910
00005920
00005930
00005940
00005950
00005960
00005970
00005980
00005990
00006000
00006010
00006020
00006030
00006040
00006050
00006060
00006070
00006080
00006090
00006100
00006110
00006120
00006130
00006140
00006150
00006160
00006170
00006180
00006190
00006200
00006210
00006220
00006230
00006240
00006250
00006260
00006270
00006280
00006290
00006300
00006310
00006320
00006330
00006340
00006350
00006360
00006370
00006380
00006390
00006400
00006410
00006420
00006430
00006440
00006450
00006460
00006470
00006480
00006490

```

```
CL=DCOS(CLON1-CLON0)
ES=DSIN(PHI1)*ECC
CHI1=2.00*DATAN(DTAN(PI/4.00+PHI1/2.00)*((1.00-ES)/(1.00+ES))*
*(ECC/2.00))-PI/2.00
SPP=DSIN(CHI1)
CPP=DCOS(CHI1)
ESQ=E2*DSIN(PHI1)**2
CPH=DCOS(PHI1)
00006500
00006510
00006520
00006530
00006540
00006550
00006560
00006570
00006580
00006590
00006600
00006610
00006620
00006630
00006640
00006650
00006660
00006670
00006680
00006690
00006700
00006710
00006720
00006730
00006740
00006750
00006760
00006770
00006780
00006790
00006800
00006810
00006820
00006830
00006840
00006850
00006860
00006870
00006880
00006890
00006900
00006910
00006920
00006930
00006940
00006950
00006960
00006970
00006980
00006990
00007000
00007010
00007020
00007030
00007040
00007050
00007060
00007070
00007080

C
C   COMPUTE COORDINATES FOR FINAL OBLIQUE MERCATOR.
C
223 XP=DATAN2(SPP*CP+CPP*SP*SL, CPP*CL)-XPO
    CZ=SP*SPP-CP*CPP*SL
    YP=LOG((1.00+CZ)/(1.00-CZ))/2.00-YPO
    SKPR=CPP*DSQRT(1.00-ESQ)/DSQRT(1.00-CZ**2)/CPH
    GO TO 240
230 CONTINUE
    IF (KODE2.EQ.0) GO TO 232
    SL=DSIN(CLON(I)-CLONP)
    CL=DCOS(CLON(I)-CLONP)
    SPP=DSIN(CHI(I))
    CPP=DCOS(CHI(I))
    ESQ=E2*DSIN(PHI(I))**2
    CPH=DCOS(PHI(I))
    GO TO 233
232 SL=DSIN(CLON1-CLONP)
    CL=DCOS(CLON1-CLONP)
    ES=DSIN(PHI1)*ECC
    CHI1=2.00*DATAN(DTAN(PI/4.00+PHI1/2.00)*((1.00-ES)/(1.00+ES))*
*(ECC/2.00))-PI/2.00
    SPP=DSIN(CHI1)
    CPP=DCOS(CHI1)
    ESQ=E2*DSIN(PHI1)**2
    CPH=DCOS(PHI1)
00006840
00006850
00006860
00006870
00006880
00006890
00006900
00006910
00006920
00006930
00006940
00006950
00006960
00006970
00006980
00006990
00007000
00007010
00007020
00007030
00007040
00007050
00007060
00007070
00007080

C
C   COMPUTE COORDINATES FOR FINAL OBLIQUE STEREOGRAPHIC.
C
233 CZ=SP*SPP+CP*CPP*CL
    SZ=DSQRT(1.00-CZ**2)
    S=2.00/(1.00+CZ)
    SKPR=S*CPP/(CPH/DSQRT(1.00-ESQ))
    XP=S*CPP*SL
    YP=S*(CP*SPP-SP*CPP*CL)
240 CONTINUE
C
C   SKPR IS SCALE FACTOR ON OBLIQUE CONFORMAL CONIC BASE.
C
C   COMPUTE COMPLEX SERIES USING KNUTH ALGORITHM.
C
C
    IF (N.EQ.1) GO TO 263
    AR=A(N)
    AI=B(N)
    BR=A(N-1)
    BI=B(N-1)
    CR=DFLOAT(N)*A(N)
    CI=DFLOAT(N)*B(N)
    DR=DFLOAT(N-1)*A(N-1)
    DI=DFLOAT(N-1)*B(N-1)
```

R=XP+XP	00007090
S=XP*XP+YP*YP	00007100
DO 260 J=2,N	00007110
ARN=BR+R*AR	00007120
AIN=BI+R*AI	00007130
IF (J.EQ.N) GO TO 260	00007140
BR=A(N-J)-S*AR	00007150
BI=B(N-J)-S*AI	00007160
AR=ARN	00007170
AI=AIN	00007180
CRN=DR+R*CR	00007190
CIN=DI+R*CI	00007200
DR=DFLOAT(N-J)*A(N-J)-S*CR	00007210
DI=DFLOAT(N-J)*B(N-J)-S*CI	00007220
CR=CRN	00007230
CI=CIN	00007240
260 CONTINUE	00007250
BR=-S*AR	00007260
BI=-S*AI	00007270
AR=ARN	00007280
AI=AIN	00007290
X=XP*AR-YP*AI+BR	00007300
Y=YP*AR+XP*AI+BI	00007310
U=XP*CR-YP*CI+DR	00007320
V=YP*CR+XP*CI+DI	00007330
SK=DSQRT(U+U+V+V)*SKPR	00007340
IF (KODE2.EQ.0) GO TO 268	00007350
GO TO 264	00007360
263 CONTINUE	00007370
X=XP	00007380
Y=YP	00007390
SK=SKPR*A(1)	00007400
IF (KODE2.EQ.0) GO TO 268	00007410
264 CONTINUE	00007420
PHI1=PHI(1)/DGRAD	00007430
CLON1=CLON(1)/DGRAD	00007440
WRITE (6,270) CLON1,PHI1,SK	00007450
270 FORMAT (1H,2X,3F11.5)	00007460
265 CONTINUE	00007470
GO TO 282	00007480
268 PHI1=PHI1/DGRAD	00007490
CLON1=CLON1/DGRAD	00007500
WRITE (6,271) CLON1,PHI1,SK,X,Y	00007510
271 FORMAT (1H,2X,5F11.5)	00007520
PHI1=PHI1-D(LAT)	00007530
IF (PHI1.GT.BLAT-0.01D0) GO TO 214	00007540
PHI1=TLAT	00007550
CLON1=CLON1+DLONG	00007560
WRITE (6,280)	00007570
IF (CLON1.LE.ELONG+0.01D0) GO TO 214	00007580
280 FORMAT (1H)	00007590
282 GO TO 22	00007600
480 WRITE (6,490)	00007610
490 FORMAT (1H0,' TOO MANY ITERATIONS.')	00007620
GO TO 22	00007630
510 WRITE (6,520)	00007640
520 FORMAT (1H0,2X,'NORMAL END OF RUN.')	00007650
STOP	00007660
END	00007670

```

SUBROUTINE NRME (OBEQ,N,M,M1,A,NT,ICOL,JCOL)
REAL*8 OBEQ,A
DIMENSION OBEQ(1,30),A(1)
II=(ICOL-1)*(2*NT+4-ICOL)/2
IK=(ICOL*(2*NT+3-ICOL)/2)
MM=M+M1+1
IJ=II+JCOL-ICOL
DO 206 IN=1,N
ICT=II
IIK=IK
DO 201 IM=1,M
DO 200 JM=IM,M
ICT=ICT+1
200 A(ICT)=A(ICT)+OBEQ(IN,IM)*OBEQ(IN,JM)
A(IIK)=A(IIK)+OBEQ(IN,IM)*OBEQ(IN,MM)
ICT=(ICOL+IM-1)*(2*NT+4-ICOL-IM)/2
201 IIK=ICT-ICOL-IM+NT+2
IF (M1) 206,206,202
202 KM=M+1
KM1=M+M1
IIJ=IJ
DO 203 IM=1,M
DO 2202 JM=KM,KM1
IIJ=IIJ+1
2202 A(IIJ)=A(IIJ)+OBEQ(IN,IM)*OBEQ(IN,JM)
203 IIJ=IJ+((2*NT+3-2*ICOL-IM)*IM)/2
JJ=(JCOL-1)*(2*NT+4-JCOL)/2
JK=(JCOL*(2*NT+3-JCOL)/2)
ICT=JJ
IIK=JK
DO 205 LM=KM,KM1
DO 204 JM=LM,KM1
ICT=ICT+1
204 A(ICT)=A(ICT)+OBEQ(IN,LM)*OBEQ(IN,JM)
A(IIK)=A(IIK)+OBEQ(IN,LM)*OBEQ(IN,MM)
ICT=(JCOL+LM-KM)*(2*NT-JCOL-LM+KM+3)/2
205 IIK=ICT-JCOL-LM+KM+NT+1
206 CONTINUE
RETURN
END
SUBROUTINE SMLE (A,N,B,IERR)
DIMENSION A(1),B(1)
REAL*8 A,B,FAC,FMUL
C
C BEGIN FORWARD SOLUTION
C
N1=N+1
DO 5 I=1,N
II=1+((I-1)*(2*N+4-I))/2
IF (A(II)) 1,1,2
1 IERR=II
GO TO 10
2 FMUL=1.00/A(II)
JCOL=I+1
DO 5 J=JCOL,N1
IIJI=II+J-I
FAC=A(IIJI)
A(IIJI)=FAC*FMUL
JJ=J
00007680
00007690
00007700
00007710
00007720
00007730
00007740
00007750
00007760
00007770
00007780
00007790
00007800
00007810
00007820
00007830
00007840
00007850
00007860
00007870
00007880
00007890
00007900
00007910
00007920
00007930
00007940
00007950
00007960
00007970
00007980
00007990
00008000
00008010
00008020
00008030
00008040
00008050
00008060
00008070
00008080
00008090
00008100
00008110
00008120
00008130
00008140
00008150
00008160
00008170
00008180
00008190
00008200
00008210
00008220
00008230
00008240
00008250
00008260

```

```

      IF (JCOL-N1) 2002,5,5
2002 IF (N-JJ) 1002,3,3
1002 JJ=N
      DO 4 K=JCOL,JJ
      KJ=1+J-K+((K-1)*(2*N+4-K))/2
      IIKI=II+K-I
      4 A(KJ)=A(KJ)-FAC*A(IIKI)
      5 CONTINUE
C
C      BEGIN BACK SOLUTION
C
      I=N+1
      I=I-1
      6 IJ=(I*(2*N+3-I))/2
      B(I)=A(IJ)
      J=N
      L=2+((I-1)*(2*N+4-I))/2
      7 IJ=IJ-1
      IF (L-IJ) 1007,1007,8
1007 B(I)=B(I)-A(IJ)*B(J)
      J=J-1
      GO TO 7
      8 IF (I-1) 9,9,6
      9 IERR=0
      10 RETURN
      END

```

```

/*
//GO.SYSIN DD *
162      0.      50.      -100.
81.5-85.  13.3028
81.5-55.  13.3028
81.5-25.  13.3028
78. -120. 16.6329
78. -100. 16.6329
78. -80.  16.6329
78. -60.  16.6329
78. -40.  16.6329
78. -20.  16.6329
74. -120. 22.0510
74. -100. 22.0510
74. -80.  22.0510
74. -60.  22.0510
74. -40.  22.0510
74. -20.  22.0510
69.5-162.5 29.2656
69.5-145.  35.0207
69.5-127.5 26.2656
69.5-112.5 26.2656
69.5-97.5  26.2656
69.5-82.5  26.2656
69.5-67.5  26.2656
69.5-52.5  26.2656
69.5-37.5  26.2656
69.5-25.   17.5104
64. -172.  19.2883
64.5-160.  25.8307
64.5-148.  25.8307
64.5-136.  25.8307
64.5-124.  25.8307

```

```

00008270
00008280
00008290
00008300
00008310
00008320
00008330
00008340
00008350
00008360
00008370
00008380
00008390
00008400
00008410
00008420
00008430
00008440
00008450
00008460
00008470
00008480
00008490
00008500
00008510
00008520
00008530
00008540
00008550
00008560
00008570
00008580
00008590
00008600
00008610
00008620
00008630
00008640
00008650
00008660
00008670
00008680
00008690
00008700
00008710
00008720
00008730
00008740
00008750
00008760
00008770
00008780
00008790
00008800
00008810
00008820
00008830
00008840
00008850

```

64.5-112. 25.8307
64.5-100. 25.8307
64.5-88. 25.8307
64.5-76. 25.8307
64.5-64. 25.8307
64.5-52. 25.8307
64. -42. 21.9186
58.5-175. 36.5749
58.5-165. 36.5749
58.5-155. 36.5749
59.5-145. 25.3769
58.5-135. 36.5749
58.5-125. 36.5749
58.5-115. 36.5749
58.5-105. 36.5749
58.5-95. 36.5749
58.5-85. 36.5749
58.5-75. 36.5749
58.5-65. 36.5749
58.5-55. 36.5749
61. -45. 9.6962
52.5-174. 24.3505
52.5-178. 24.3505
52.5-170. 24.5305
52.5-163. 18.2628
52.5-132.5 15.2190
52.5-127.5 15.2190
52.5-122.5 15.2190
52.5-117.5 15.2190
52.5-112.5 15.2190
52.5-107.5 15.2190
52.5-102.5 15.2190
52.5-97.5 15.2190
52.5-92.5 15.2190
52.5-87.5 15.2190
52.5-82.5 15.2190
52.5-77.5 15.2190
52.5-72.5 15.2190
52.5-67.5 15.2190
52.5-62.5 15.2190
52.5-57.5 15.2190
52.5-52.5 15.2190
47.5-127.5 16.8898
47.5-122.5 16.8898
47.5-117.5 16.8898
47.5-112.5 16.8898
47.5-107.5 16.8898
47.5-102.5 16.8898
47.5-97.5 16.8898
47.5-92.5 16.8898
47.5-87.5 16.8898
47.5-82.5 16.8898
47.5-77.5 16.8398
47.5-72.5 16.8898
47.5-67.5 16.8898
47.5-62.5 16.8898
47.5-57.5 16.8898
47.5-52.5 16.8898
42.5-127.5 18.4319

00008860
00008870
00008880
00008890
00008900
00008910
00008920
00008930
00008940
00008950
00008960
00008970
00008980
00008990
00009000
00009010
00009020
00009030
00009040
00009050
00009060
00009070
00009080
00009090
00009100
00009110
00009120
00009130
00009140
00009150
00009160
00009170
00009180
00009190
00009200
00009210
00009220
00009230
00009240
00009250
00009260
00009270
00009280
00009290
00009300
00009310
00009320
00009330
00009340
00009350
00009360
00009370
00009380
00009390
00009400
00009410
00009420
00009430
00009440

42.5-122.5 18.4319
42.5-117.5 18.4319
42.5-112.5 18.4319
42.5-107.5 18.4319
42.5-102.5 18.4319
42.5-97.5 18.4319
42.5-92.5 18.4319
42.5-87.5 18.4319
42.5-82.5 18.4319
42.5-77.5 18.4319
42.5-72.5 18.4319
42.5-67.5 18.4319
43.5-62.5 10.8806
38.5-127.5 11.7391
37.5-122.5 19.8338
37.5-117.5 19.8338
37.5-112.5 19.8388
37.5-107.5 19.8338
37.5-102.5 19.8338
37.5-97.5 19.8338
37.5-92.5 19.8338
37.5-87.5 19.8338
37.5-82.5 19.8338
37.5-77.5 19.8338
37.5-73.5 11.9003
33.5-122.5 12.5083
32.5-117.5 21.0848
32.5-112.5 21.0848
32.5-107.5 21.0848
32.5-102.5 21.0848
32.5-97.5 21.0848
32.5-92.5 21.0848
32.5-87.5 21.0848
32.5-82.5 21.0848
32.5-77.5 21.0848
28. -117.5 17.6590
26. -112.5 35.9518
27.5-107.5 22.1753
27.5-102.5 22.1753
27.5-97.5 22.1753
27.5-92.5 22.1753
27.5-87.5 22.1753
27.5-82.5 22.1753
27.5-77.5 22.1753
26. -72.5 8.9879
22.5-107.5 23.0970
22.5-102.5 23.0970
22.5-97.5 23.0970
22.5-92.5 23.0970
22.5-87.5 23.0970
22.5-82.5 23.0970
22.5-77.5 23.0970
22.5-72.5 23.0970
19. -107.5 9.4552
18. -102.5 19.0211
17.5-97.5 23.8429
17.5-92.5 23.8429
17.5-87.5 23.8429
17.5-82.5 23.8429

00009450
00009460
00009470
00009480
00009490
00009500
00009510
00009520
00009530
00009540
00009550
00009560
00009570
00009580
00009590
00009600
00009610
00009620
00009630
00009640
00009650
00009660
00009670
00009680
00009690
00009700
00009710
00009720
00009730
00009740
00009750
00009760
00009770
00009780
00009790
00009800
00009810
00009820
00009830
00009840
00009850
00009860
00009870
00009880
00009890
00009900
00009910
00009920
00009930
00009940
00009950
00009960
00009970
00009980
00009990
00010000
00010010
00010020
00010030

LEVEL 2.3.0 (JUNE 78)

OS/360 FORTRAN H EXTENDED

DATE 84.250/13.02.32

PAGE 1

REQUESTED OPTIONS:

OPTIONS IN EFFECT: NAME(MAIN) OPTIMIZE(1) LINECOUNT(54) SIZE(MAX) AUTODBL(NONE)
 SOURCE EBCDIC NOLIST NODECK OBJECT MAP NOFORMAT GOSTMT NOXREF NOALC NOANSF NOTERM IBM FLAG(I)

```

C      00000060
C      THIS PROGRAM DETERMINES COEFFICIENTS FOR A CONFORMAL MAP      00000070
C      PROJECTION WITH MINIMUM SCALE ERROR THROUGH A GIVEN NUMBER    00000080
C      OF POINTS.                                                    00000090
C      00000100
ISN 0002      IMPLICIT REAL*8(A-H,O-Z)                                00000110
ISN 0003      DIMENSION PHI(168),CLON(168),A(15),B(15),DKDA(15),DKDB(15) 00000120
ISN 0004      DIMENSION FA(15),FB(15),DFADA(15,15),DFBDB(15,15),DFADB(15,15) 00000130
ISN 0005      DIMENSION C(500),D(30),DBEQ(1,30),WT(168)              00000140
ISN 0006      DIMENSION DFADP(15),DFADL(15),DFADN(15),DFBDP(15),DFBDL(15),DFBDN(15) 00000150
ISN 0007      *15)                                                    00000160
      DIMENSION DF1A(15),DF2A(15),CHI(168)                            00000170
C      00000180
C      READ NUMBER OF POINTS M (MAX=168), NUMBER OF COEFFICIENTS N    00000190
C      (MAX=15) OF A AND NUMBER N OF B, AND LAT/LONG OF FIXED POINTS. 00000200
C      00000210
ISN 0008      READ (5,5) M,E2,CHIR,CLONR                               00000220
ISN 0009      5 FORMAT (I3,7X,3F10.8)                                00000230
ISN 0010      PI=3.1415926535D0                                         00000240
ISN 0011      DGRAD=PI/180.D0                                           00000250
ISN 0012      ECC=DSQRT(E2)                                              00000260
ISN 0013      E4=E2*E2                                                  00000270
ISN 0014      E6=E4*E2                                                  00000280
ISN 0015      WRITE (6,6) M                                             00000290
ISN 0016      6 FORMAT (I1,2X,'CONFORMAL MAP PROJECTION COEFFICIENTS CALCULATION' 00000300
      */(I10,2X,'LAT./LONG. AND WTS. OF ',I3,' POINTS FOR MINIMUM SCALE ERROR') 00000310
      *RROR'))                                                         00000320
ISN 0017      CHIR=CHIR*DGRAD                                           00000330
ISN 0018      CLONR=CLONR*DGRAD                                         00000340
ISN 0019      DO 20 I=1,M                                              00000350
ISN 0020      READ (5,10) PHI(I),CLON(I),WT(I)                        00000360
ISN 0021      10 FORMAT (F4.1,F6.1,F10.5)                              00000370
ISN 0022      WRITE (6,15) I,PHI(I),CLON(I),WT(I)                    00000380
ISN 0023      15 FORMAT (I1,2X,I3,'.',2F10.1,F14.5)                  00000390
ISN 0024      PHI(I)=PHI(I)*DGRAD                                       00000400
ISN 0025      CLON(I)=CLON(I)*DGRAD                                     00000410
ISN 0026      ES=DSIN(PHI(I))*ECC                                       00000420
ISN 0027      CHI(I)=2.D0*DATAN(DTAN(PI/4.D0+PHI(I)/2.D0)*((1.D0-ES)/(1.D0+ES))* 00000430
      *(ECC/2.D0))-PI/2.D0                                             00000440
ISN 0028      20 CONTINUE                                              00000450
ISN 0029      22 CONTINUE                                              00000460
C      00000470
C      IF KODE IS 0, CONE CONSTANT EN CAN VARY. IF KODE IS NOT 0, CONE 00000480
C      CONSTANT IS FIXED.                                              00000490
C      00000500
C      IF KODE1 IS 0, POLE CAN VARY. IF KODE1 IS 1, LONG. (CLONP) ONLY 00000510
C      OF POLE CAN VARY. IF KODE1 IS 2, LAT. (PHIP)/LONG. OF POLE IS 00000520

```

LEVEL 2.3.0 (JUNE 78)

MAIN

OS/360 FORTRAN H EXTENDED

DATE 84.250/13.02.32

PAGE 2

```
C      FIXED.                                00000530
C      IF KODE2 IS 0, GRATICULE WILL BE PRINTED. IF KODE2 IS NOT 0, 00000540
C      SCALE FACTORS OF THE M POINTS WILL BE PRINTED AFTER COMPUTATION 00000550
C      OF CONSTANTS.                                00000560
C      ENTER THE INITIAL ESTIMATE (OR FINAL VALUE) OF PHIP AND CLONP IN 00000570
C      DEG. AND DECIMALS, WITH OTHER VALUES ON CARD. THE CONE CONSTANT 00000580
C      WITH KODE NOT EQUAL TO 0 WILL ESTABLISH WHETHER THE PROJECTION 00000590
C      IS OBLIQUE MERCATOR OR STEREOGRAPHIC.        00000600
C                                                    00000610
C                                                    00000620
ISN 0030      READ (5,25) N,PHIP,CLONP,EN,KODE,KODE1,KODE2 00000630
ISN 0031      25 FORMAT (I2,8X,3F10.6,3I2) 00000640
ISN 0032      IF (N.EQ.0) GO TO 510 00000650
ISN 0034      WRITE (6,30) PHIP,CLONP,EN,KODE,KODE1 00000660
ISN 0035      30 FORMAT(1H0,2X,'INIT. EST. LAT./LONG. OF TRANSF. POLE FOR OBLIQUE 00000670
*ONF. CONIC, TRANS. OR OBLIQUE MERCATOR (IF CONE CONST. IS 0), OBLI00000680
*QUE STEREOG.'/(1H,2X,'(IF CONE CONSTANT IS 1):',2F10.5,' CONE CON00000690
*ST.:',F10.7,' CODE=',I1,' CODE1=',I1)) 00000700
ISN 0036      WRITE (6,32) 00000710
ISN 0037      32 FORMAT (1H0,2X,'IF CODE IS NOT 0, CONE CONST. IS FIXED. IF CODE1 00000720
* IS 1, LAT. OF POLE IS FIXED. IF CODE1 IS 2, LAT/LONG OF POLE 00000730
* IS FIXED.') 00000740
ISN 0038      PHIP=PHIP*DGRAD 00000750
ISN 0039      ES=DSIN(PHIP)*ECC 00000760
ISN 0040      CHIP=2.D0*DATAN(DTAN(PI/4.D0+PHIP/2.D0)*((1.D0-ES)/(1.D0+ES))* 00000770
* (ECC/2.D0))-PI/2.D0 00000780
ISN 0041      CLONP=CLONP*DGRAD 00000790
ISN 0042      IF (KODE.EQ.0.AND.KODE1.EQ.0) N4=N+N+2 00000800
ISN 0044      IF (KODE.NE.0.AND.KODE1.EQ.0) N4=N+N+1 00000810
ISN 0046      IF (KODE.EQ.0.AND.KODE1.EQ.1) N4=N+N+1 00000820
ISN 0048      IF (KODE.NE.0.AND.KODE1.EQ.1) N4=N+N 00000830
ISN 0050      IF (KODE.EQ.0.AND.KODE1.EQ.2) N4=N+N 00000840
ISN 0052      IF (KODE.NE.0.AND.KODE1.EQ.2) N4=N+N-1 00000850
ISN 0054      N1=N+1 00000860
ISN 0055      N2=N+N-1 00000870
ISN 0056      N3=N4+1 00000880
ISN 0057      N5=N+N 00000890
ISN 0058      IF (KODE1.EQ.1) N6=N+N 00000900
ISN 0060      IF (KODE1.EQ.0) N6=N+N+1 00000910
ISN 0062      NN=N-1 00000920
C                                                    00000930
C      TLAT IS TOP LAT, BLAT IS BOTTOM LAT, WLONG IS WESTMOST LONG, ELONG00000940
C      IS EASTMOST LONG, DLAT IS LAT INTERVAL AND DLONG I' LONG INTERVAL 00000950
C      FOR GRATICULE IN DEGREES. 00000960
C                                                    00000970
ISN 0063      READ (5,35) TLAT,BLAT,WLONG,ELONG,DLAT,DLONG 00000980
ISN 0064      35 FORMAT (10X,6F10.3) 00000990
C                                                    00001000
C      INITIALIZE COEFFICIENTS: A(1)=1 AND THE REST ARE 0. 00001010
C                                                    00001020
ISN 0065      DO 40 I=1,N 00001030
ISN 0066      A(I)=0.D0 00001040
```

PAGE 3

ISN 0067	B(1)=0.D0	00001050
ISN 0068	40 CONTINUE	00001060
ISN 0069	A(1)=1.D0	00001070
ISN 0070	NE=0	00001080
ISN 0071	IF (N.EQ.1) GO TO 418	00001090
ISN 0072	IF (KODE.EQ.0.AND.KODE1.EQ.0) WRITE (6,41) N,N	00001100
ISN 0073	41 FORMAT (1H0,2X,'SUCCESSIVE ADJUSTMENTS TO COEFFICIENTS A(1)-A('	00001110
ISN 0074	*,') B(2)-B('12,') , LAT/LONG OF POLE, AND CONE CONST.:')	00001120
ISN 0075	IF (KODE.EQ.0.AND.KODE1.EQ.1) WRITE (6,421) N,N	00001130
ISN 0076	421 FORMAT (1H0,2X,'SUCCESSIVE ADJUSTMENTS TO COEFFICIENTS A(1)-A('	00001140
ISN 0077	*,') B(2)-B('12,') , LONG OF POLE, AND CONE CONST.:')	00001150
ISN 0078	IF (KODE.EQ.0.AND.KODE1.EQ.2) WRITE (6,411) N,N	00001160
ISN 0079	411 FORMAT (1H0,2X,'SUCCESSIVE ADJUSTMENTS TO COEFFICIENTS A(1)-A('	00001170
ISN 0080	*,') B(2)-B('12,') , AND CONE CONST.:')	00001180
ISN 0081	IF (KODE.NE.0.AND.KODE1.EQ.0) WRITE (6,412) N,N	00001190
ISN 0082	412 FORMAT (1H0,2X,'SUCCESSIVE ADJUSTMENTS TO COEFFICIENTS A(1)-A('	00001200
ISN 0083	*,') B(2)-B('12,') , AND LAT/LONG OF POLE:')	00001210
ISN 0084	IF (KODE.NE.0.AND.KODE1.EQ.1) WRITE (6,422) N,N	00001220
ISN 0085	422 FORMAT (1H0,2X,'SUCCESSIVE ADJUSTMENTS TO COEFFICIENTS A(1)-A('	00001230
ISN 0086	*,') B(2)-B('12,') , AND LONG OF POLE:')	00001240
ISN 0087	IF (KODE.NE.0.AND.KODE1.EQ.2) WRITE (6,413) N,N	00001250
ISN 0088	413 FORMAT (1H0,2X,'SUCCESSIVE ADJUSTMENTS TO COEFFICIENTS A(1)-A('	00001260
ISN 0089	*,') , AND B(2)-B('12,') :')	00001270
ISN 0090	GO TO 419	00001280
ISN 0091	418 CONTINUE	00001290
ISN 0092	IF (KODE.EQ.0.AND.KODE1.EQ.0) WRITE (6,414) N	00001300
ISN 0093	414 FORMAT (1H0,2X,'SUCCESSIVE ADJUSTMENTS TO COEFFICIENTS A(1)-A('	00001310
ISN 0094	*,') , LAT/LONG OF POLE, AND CONE CONST.:')	00001320
ISN 0095	IF (KODE.EQ.0.AND.KODE1.EQ.1) WRITE (6,423) N	00001330
ISN 0096	423 FORMAT (1H0,2X,'SUCCESSIVE ADJUSTMENTS TO COEFFICIENTS A(1)-A('	00001340
ISN 0097	*,') , LONG OF POLE, AND CONE CONST.:')	00001350
ISN 0098	IF (KODE.EQ.0.AND.KODE1.EQ.2) WRITE (6,415) N	00001360
ISN 0099	415 FORMAT (1H0,2X,'SUCCESSIVE ADJUSTMENTS TO COEFFICIENTS A(1)-A('	00001370
ISN 0100	*,') , AND CONE CONST.:')	00001380
ISN 0101	IF (KODE.NE.0.AND.KODE1.EQ.0) WRITE (6,416) N	00001390
ISN 0102	416 FORMAT (1H0,2X,'SUCCESSIVE ADJUSTMENTS TO COEFFICIENTS A(1)-A('	00001400
ISN 0103	*,') , AND LAT/LONG OF POLE:')	00001410
ISN 0104	IF (KODE.NE.0.AND.KODE1.EQ.1) WRITE (6,424) N	00001420
ISN 0105	424 FORMAT (1H0,2X,'SUCCESSIVE ADJUSTMENTS TO COEFFICIENTS A(1)-A('	00001430
ISN 0106	*,') , AND LONG OF POLE:')	00001440
ISN 0107	IF (KODE.NE.0.AND.KODE1.EQ.2) WRITE (6,417) N	00001450
ISN 0108	417 FORMAT (1H0,2X,'SUCCESSIVE ADJUSTMENTS TO COEFFICIENTS A(1)-A('	00001460
ISN 0109	*,') :')	00001470
ISN 0110	419 CONTINUE	00001480
ISN 0111	42 CONTINUE	00001490
ISN 0112	SP=DSIN(CHIP)	00001500
ISN 0113	CP=DCOS(CHIP)	00001510
ISN 0114	IF (DABS(EN).LT.1.0-7) GO TO 43	00001520
ISN 0115	IF (DABS(EN).GT.1.0-7) GO TO 432	00001530
ISN 0116		00001540
ISN 0117	COMPUTE COORDINATES OF CENTER FOR OBLIQUE CONFORMAL CONIC.	00001550
ISN 0118		00001560

LEVEL 2.3.0 (JUNE 78)

MAIN

05/360 FORTRAN H EXTENDED

DATE 84.250/13.02.32

PAGE 4

```
ISN 0119      SL=DSIN(CLONR-CLONP)                00001570
ISN 0120      CL=DCOS(CLONR-CLONP)                00001580
ISN 0121      SPP=DSIN(CHIR)                      00001590
ISN 0122      CPP=DCOS(CHIR)                      00001600
ISN 0123      CZ=SP*SPP+CP*CPP*CL                 00001610
ISN 0124      SZ=DSQRT(1.00-CZ**2)                00001620
ISN 0125      F=DSQRT((1.00+EN)**(1.00+EN)*(1.00-EN)**(1.00-EN))/EN 00001630
ISN 0126      RHO=F*((1.00-CZ)/(1.00+CZ))*(EN/2.00) 00001640
ISN 0127      THP=EN*ATAN2(CPP*SL,CP*SPP-SP*CPP*CL) 00001650
ISN 0128      XPO=-RHO*DSIN(THP)                  00001660
ISN 0129      YPO=-RHO*DCOS(THP)                  00001670
ISN 0130      GO TO 431                            00001680
ISN 0131      43 CONTINUE                          00001690
C                                                    00001700
C      COMPUTE COORDINATES OF CENTER FOR OBLIQUE MERCATOR. 00001710
C                                                    00001720
ISN 0132      CLONO=CLONP+90.00*DGRAD              00001730
ISN 0133      SL=DSIN(CLONR-CLONO)                00001740
ISN 0134      CL=DCOS(CLONR-CLONO)                00001750
ISN 0135      SPP=DSIN(CHIR)                      00001760
ISN 0136      CPP=DCOS(CHIR)                      00001770
ISN 0137      XPO=ATAN2(SPP*CP+CPP*SP*SL,CPP*CL)   00001780
ISN 0138      CZ=SP*SPP-CP*CPP*SL                 00001790
ISN 0139      YPO=DLOG((1.00+CZ)/(1.00-CZ))/2.00   00001800
ISN 0140      GO TO 431                            00001810
ISN 0141      432 CONTINUE                         00001820
C                                                    00001830
C      COMPUTE COORDINATES OF CENTER FOR OBLIQUE STEREOGRAPHIC. 00001840
C                                                    00001850
ISN 0142      XPO=0.00                            00001860
ISN 0143      YPO=0.00                            00001870
ISN 0144      431 CONTINUE                         00001880
ISN 0145      SE=0.00                             00001890
ISN 0146      SWT=0.00                             00001900
ISN 0147      J=(N4*N4+3*N4)/2                    00001910
ISN 0148      DO 44 I=1,J                          00001920
ISN 0149      C(I)=0.00                            00001930
ISN 0150      44 CONTINUE                         00001940
ISN 0151      DO 45 I=1,N                          00001950
ISN 0152      FA(I)=0.00                          00001960
ISN 0153      FB(I)=0.00                          00001970
ISN 0154      DFADP(I)=0.00                       00001980
ISN 0155      DFADL(I)=0.00                       00001990
ISN 0156      DFADN(I)=0.00                       00002000
ISN 0157      DFBDP(I)=0.00                       00002010
ISN 0158      DFBDL(I)=0.00                       00002020
ISN 0159      DFBDN(I)=0.00                       00002030
ISN 0160      DO 45 J=1,N                          00002040
ISN 0161      DFADA(I,J)=0.00                     00002050
ISN 0162      DFADB(I,J)=0.00                     00002060
ISN 0163      DFADB(I,J)=0.00                     00002070
ISN 0164      45 CONTINUE                         00002080
```

LEVEL 2.3.0 (JUNE 78)

MAIN

OS/360 FORTRAN H EXTENDED

DATE 84.250/13.02.32

PAGE 5

```

ISN 0165      FCHIP=0.D0      00002090
ISN 0166      FLATP=0.D0      00002100
ISN 0167      FN=0.D0         00002110
ISN 0168      DFPDP=0.D0      00002120
ISN 0169      DFPDL=0.D0      00002130
ISN 0170      DFPDN=0.D0      00002140
ISN 0171      DFLDL=0.D0      00002150
ISN 0172      DFLDN=0.D0      00002160
ISN 0173      DFN DN=0.D0     00002170
                C             00002180
                C             00002190
                C      CALCULATE POLAR COORDINATES OF THE M POINTS BASED ON THE OBLIQUE
                C      CONF. CONIC PROJECTION.      00002200
                C             00002210
ISN 0174      DO 80 I=1,M      00002220
ISN 0175      IF (DABS(EN).LT.1.D-7) GO TO 453      00002230
ISN 0177      IF (DABS(EN).GT.1.D0-1.D-7) GO TO 451  00002240
                C             00002250
                C      COMPUTE FUNCTIONS FOR OBLIQUE CONFORMAL CONIC.      00002260
                C             00002270
ISN 0179      SL=DSIN(CLON(I)-CLONP)      00002280
ISN 0180      CL=DCOS(CLON(I)-CLONP)      00002290
ISN 0181      SPP=DSIN(CHI(I))      00002300
ISN 0182      CPP=DCOS(CHI(I))      00002310
ISN 0183      CZ=SP*SPP+CP*CPP*CL      00002320
ISN 0184      SZ=DSQRT(1.D0-CZ**2)      00002330
ISN 0185      RHO=F*((1.D0-CZ)/(1.D0+CZ))*((PI/2.D0) 00002340
                C             00002350
                C      SKPR IS SCALE FACTOR ON OBLIQUE CONFORMAL CONIC BASE.      00002360
                C             00002370
ISN 0186      SKPR=EN*RHO*CPP*DSQRT(1.D0-E2*DSIN(PHI(I))**2)/SZ/DCOS(PHI(I)) 00002380
ISN 0187      FAC1=CPP*SL      00002390
ISN 0188      FAC2=CP*SPP-SP*CPP*CL      00002400
ISN 0189      AZ=DATAN2(FAC1,FAC2)      00002410
ISN 0190      THP=AZ*EN      00002420
ISN 0191      CAZ=DCOS(AZ)**2      00002430
ISN 0192      STHP=DSIN(THP)      00002440
ISN 0193      CTHP=DCOS(THP)      00002450
ISN 0194      XP=-RHO*STHP      00002460
ISN 0195      YP=-RHO*CTHP      00002470
ISN 0196      DTD P=EN*CAZ*FAC1*CZ/FAC2**2      00002480
ISN 0197      DRDP=-SKPR*FAC2/SZ      00002490
ISN 0198      DKPDP=-SKPR*(EN-CZ)*FAC2/SZ**2      00002500
ISN 0199      DTD L=EN*CAZ*CPP*(SP*CPP-CP*SPP*CL)/FAC2**2 00002510
ISN 0200      DRDL=-SKPR*CP*FAC1/SZ      00002520
ISN 0201      DKPDL=-SKPR*(EN-CZ)*CP*FAC1/SZ**2      00002530
ISN 0202      FAC=0.5D0*DLOG((1.D0-CZ)*(1.D0+EN)/(1.D0+CZ)/(1.D0-EN))-1.D0/EN 00002540
ISN 0203      DTD N=AZ      00002550
ISN 0204      DRDN=RHO*FAC      00002560
ISN 0205      DKPDN=SKPR*(FAC+1.D0/EN)      00002570
ISN 0206      CC=-RHO*CTHP*DTD P-STHP*DRDP      00002580
ISN 0207      DD=RHO*STHP*DTD P-CTHP*DRDP      00002590
ISN 0208      EE=-RHO*CTHP*DTD L-STHP*DRDL      00002600

```

LEVEL 2.3.0 (JUNE 78)

MAIN

OS/360 FORTRAN H EXTENDED

DATE 84.250/13.02.32

PAGE 6

```
ISN 0209      GG=RHO*STHP*DTD-L-CTHP*DRDL      00002610
ISN 0210      HH=-RHO*CTHP*DTDN-STHP*DRDN      00002620
ISN 0211      HJ=RHO*STHP*DTDN-CTHP*DRDN      00002630
ISN 0212      GO TO 452                        00002640
ISN 0213      453 CONTINUE                      00002650
C                                                    00002660
C      COMPUTE FUNCTIONS FOR OBLIQUE MERCATOR.    00002670
C                                                    00002680
ISN 0214      SL=DSIN(CLON(I)-CLON0)           00002690
ISN 0215      CL=DCOS(CLON(I)-CLON0)           00002700
ISN 0216      SPP=DSIN(CHI(I))                 00002710
ISN 0217      CPP=DCOS(CHI(I))                 00002720
ISN 0218      XP=DATAN2(SPP*CP+CPP*SP*SL, CPP*CL) 00002730
ISN 0219      CZ=SP*SPP-CP*CPP*SL             00002740
ISN 0220      SZ=DSQRT(1.00-CZ**2)             00002750
ISN 0221      YP=DLOG((1.00+CZ)/(1.00-CZ))/2.00 00002760
ISN 0222      SKPR=CPP*DSQRT(1.00-E2*DSIN(PHI(I))**2)/DSQRT(1.00-CZ**2)/DCOS(PHI(I)) 00002770
C                                                    00002780
C      FAC=DCOS(XP)**2                          00002790
ISN 0223      CC=-CZ*FAC/ CPP/CL               00002800
ISN 0224      DD=SKPR**2*(CP*SPP+SP*CPP*SL)    00002810
ISN 0225      DKPDP=SKPR*CZ*(CP*SPP+SP*CPP*SL)/SZ**2 00002820
ISN 0226      EE=FAC*(SPP*CP*CL-SP*CPP)/SZ**2/ CPP 00002830
ISN 0227      GG=SKPR**2*CP*CPP*CL            00002840
ISN 0228      DKPDL=SKPR*CZ*CP*CPP*CL/SZ**2    00002850
ISN 0229      GO TO 452                        00002860
ISN 0230      451 CONTINUE                      00002870
C                                                    00002880
C      COMPUTE FUNCTIONS FOR OBLIQUE STEREOGRAPHIC. 00002890
C                                                    00002900
ISN 0232      SL=DSIN(CLON(I)-CLONP)           00002910
ISN 0233      CL=DCOS(CLON(I)-CLONP)           00002920
ISN 0234      SPP=DSIN(CHI(I))                 00002930
ISN 0235      CPP=DCOS(CHI(I))                 00002940
ISN 0236      CZ=SP*SPP+CP*CPP*CL             00002950
ISN 0237      SZ=DSQRT(1.00-CZ**2)             00002960
ISN 0238      S=2.00/(1.00+CZ)                 00002970
ISN 0239      SKPR=S*CPP/(DCOS(PHI(I))/DSQRT(1.00-E2*DSIN(PHI(I))**2)) 00002980
ISN 0240      XP=S*CPP*SL                     00002990
ISN 0241      YP=S*(CP*SPP-SP*CPP*CL)          00003000
ISN 0242      FAC=-YP/2.00                     00003010
ISN 0243      CC=XP*FAC                       00003020
ISN 0244      DD=-S*CZ+YP*FAC                 00003030
ISN 0245      DKPDP=SKPR*FAC                  00003040
ISN 0246      FAC=-XP*CP/2.00                 00003050
ISN 0247      EE=-S*CPP*CL+XP*FAC             00003060
ISN 0248      GG=-XP*SP+YP*FAC                00003070
ISN 0249      DKPDL=SKPR*FAC                  00003080
ISN 0250      452 CONTINUE                      00003090
ISN 0251      XP=XP-XPO                       00003100
ISN 0252      YP=YP-YPO                       00003110
ISN 0253      R=DSQRT(XP**2+YP**2)             00003120
```

LEVEL 2.3.0 (JUNE 78)

MAIN

OS/360 FORTRAN H EXTENDED

DATE 84.250/13.02.32

PAGE 7

```

ISN 0254      IF (R.LT.1.D-9) GO TO 46      00003130
ISN 0256      TH=DATAN2(YP,XP)              00003140
ISN 0257      GO TO 47                      00003150
ISN 0258      46 TH=0.D0                    00003160
ISN 0259      47 CONTINUE                   00003170
ISN 0260      F2=0.D0                      00003180
ISN 0261      F1=0.D0                      00003190
ISN 0262      DF1P=0.D0                    00003200
ISN 0263      DF1L=0.D0                    00003210
ISN 0264      DF1N=0.D0                    00003220
ISN 0265      DF2P=0.D0                    00003230
ISN 0266      DF2L=0.D0                    00003240
ISN 0267      DF2N=0.D0                    00003250
C      IF (I.LE.30.AND.NE.EQ.0) WRITE (6,92) CC,DD,EE,GG,HH,HJ,XPO,YPO,RH 00003260
C      *O,R,TH                             00003270
ISN 0268      DO 50 J=1,N                  00003280
ISN 0269      STH=DSIN(DFLOAT(J-1)*TH)     00003290
ISN 0270      CTH=DCOS(DFLOAT(J-1)*TH)     00003300
ISN 0271      STH2=DSIN(DFLOAT(J-2)*TH)    00003310
ISN 0272      CTH2=DCOS(DFLOAT(J-2)*TH)    00003320
ISN 0273      IF (R.LT.1.D-7) GO TO 48     00003330
ISN 0275      RPOW=R**(J-1)                00003340
ISN 0276      RPOW2=R**(J-2)               00003350
ISN 0277      GO TO 49                     00003360
ISN 0278      48 IF (J.EQ.1) RPOW=1.D0      00003370
ISN 0280      IF (J.NE.1) RPOW=0.D0        00003380
ISN 0282      IF (J.EQ.2) RPOW2=1.D0       00003390
ISN 0284      IF (J.NE.2) RPOW2=0.D0       00003400
ISN 0286      49 CONTINUE                   00003410
ISN 0287      FAC=DFLOAT(J)*RPOW           00003420
ISN 0288      FAC2=DFLOAT(J*(J-1))*RPOW2   00003430
ISN 0289      F2=F2+FAC*(A(J)*CTH-B(J)*STH) 00003440
ISN 0290      F1=F1+FAC*(A(J)*STH+B(J)*CTH) 00003450
ISN 0291      IF (KODE1.EQ.2) GO TO 482     00003460
ISN 0293      DF1P=DF1P+FAC2*((A(J)*CC-B(J)*DD)*STH2+(A(J)*DD+B(J)*CC)*CTH2) 00003470
ISN 0294      DF2P=DF2P+FAC2*((A(J)*CC-B(J)*DD)*CTH2-(A(J)*DD+B(J)*CC)*STH2) 00003480
ISN 0295      DF1L=DF1L+FAC2*((A(J)*EE-B(J)*GG)*STH2+(A(J)*GG+B(J)*EE)*CTH2) 00003490
ISN 0296      DF2L=DF2L+FAC2*((A(J)*EE-B(J)*GG)*CTH2-(A(J)*GG+B(J)*EE)*STH2) 00003500
ISN 0297      482 CONTINUE                  00003510
ISN 0298      IF (KODE.NE.0) GO TO 491     00003520
ISN 0300      DF1N=DF1N+FAC2*((A(J)*HH-B(J)*HJ)*STH2+(A(J)*HJ+B(J)*HH)*CTH2) 00003530
ISN 0301      DF2N=DF2N+FAC2*((A(J)*HH-B(J)*HJ)*CTH2-(A(J)*HJ+B(J)*HH)*STH2) 00003540
ISN 0302      491 CONTINUE                  00003550
ISN 0303      DF1A(J)=FAC*STH               00003560
ISN 0304      DF2A(J)=FAC*CTH               00003570
ISN 0305      50 CONTINUE                   00003580
ISN 0306      SK=DSQRT(F2**2+F1**2)*SKPR    00003590
C      IF (I.LE.30.AND.NE.EQ.0) WRITE (6,92) SK,SKPR,DF1P,DF2P,DF1L,DF2L, 00003600
C      *F1,F2                             00003610
C      SK IS SCALE FACTOR ON TRANSFORMED MAP PROJECTION. 00003620
C      00003630
C      00003640

```

LEVEL 2.3.0 (JUNE 78)

MAIN

OS/360 FORTRAN H EXTENDED

DATE 84.250/13.02.32

PAGE 8

```

ISN 0307      SE=SE+WT(I)*(SK-1.D0)**2      00003650
ISN 0308      SWT=SWT+WT(I)                  00003660
ISN 0309      DO 60 J=1,N                    00003670
ISN 0310      DKDA(J)=(SKPR**2/SK)*(F1*DF1A(J)+F2*DF2A(J)) 00003680
ISN 0311      DKDB(J)=(SKPR**2/SK)*(F1*DF2A(J)-F2*DF1A(J)) 00003690
ISN 0312      FA(J)=FA(J)+WT(I)*(SK-1.D0)*DKDA(J) 00003700
ISN 0313      FB(J)=FB(J)+WT(I)*(SK-1.D0)*DKDB(J) 00003710
C             IF (I.LE.30.AND.NE.EQ.0) WRITE (6,69) J,DKDA(J),DKDB(J) 00003720
C 69          FORMAT (1H,2X,I3,2F13.7)        00003730
C 60          CONTINUE                        00003740
ISN 0314      DKDP=(SKPR**2/SK)*(F1*DF1P+F2*DF2P)+(SK/SKPR)*DKPDP 00003750
ISN 0315      DKDL=(SKPR**2/SK)*(F1*DF1L+F2*DF2L)+(SK/SKPR)*DKPDL 00003760
ISN 0316      IF (KODE.EQ.0) DKDN=(SKPR**2/SK)*(F1*DF1N+F2*DF2N)+(SK/SKPR)*DKPDN 00003770
ISN 0317      FCHIP=FCHIP+WT(I)*(SK-1.D0)*DKDP 00003780
ISN 0319      C             IF (I.LE.30.AND.NE.EQ.0) WRITE (6,92) DKDP,DKDL,DKDN 00003790
ISN 0320      FLATP=FLATP+WT(I)*(SK-1.D0)*DKDL 00003800
ISN 0321      IF (KODE.EQ.0) FN=FN+WT(I)*(SK-1.D0)*DKDN 00003810
C             IF (I.LE.30.AND.NE.EQ.0) WRITE (6,92) DKDP,DKDL,DKDN,DKDA(1), 00003820
C             *DKPDP,DKPDL,DKPDN              00003830
C             A FACTOR OF 2 IS REMOVED FROM ALL MULTIPLIERS IN THE 00003840
C             SIMULTANEOUS EQUATIONS.         00003850
C             00003860
C             00003870
ISN 0323      IF (KODE1.EQ.0) DFPDP=DFPDP+WT(I)*DKDP**2 00003880
ISN 0325      IF (KODE1.LE.1) DFPDL=DFPDL+WT(I)*DKDP*DKDL 00003890
ISN 0327      IF (KODE.EQ.0) DFPDN=DFPDN+WT(I)*DKDP*DKDN 00003900
ISN 0329      IF (KODE1.LE.1) DFLDL=DFLDL+WT(I)*DKDL**2 00003910
ISN 0331      IF (KODE.EQ.0) DFLDN=DFLDN+WT(I)*DKDL*DKDN 00003920
ISN 0333      IF (KODE.EQ.0) DFNDN=DFNDN+WT(I)*DKDN**2 00003930
ISN 0335      DO 80 J=1,N                    00003940
ISN 0336      IF (KODE1.EQ.0) DFADP(J)=DFADP(J)+WT(I)*DKDP*DKDA(J) 00003950
ISN 0338      IF (KODE1.EQ.0) DFBDP(J)=DFBDP(J)+WT(I)*DKDP*DKDB(J) 00003960
ISN 0340      IF (KODE1.LE.1) DFADL(J)=DFADL(J)+WT(I)*DKDL*DKDA(J) 00003970
ISN 0342      IF (KODE1.LE.1) DFBDL(J)=DFBDL(J)+WT(I)*DKDL*DKDB(J) 00003980
ISN 0344      IF (KODE.EQ.0) DFADN(J)=DFADN(J)+WT(I)*DKDN*DKDA(J) 00003990
ISN 0346      IF (KODE.EQ.0) DFBDN(J)=DFBDN(J)+WT(I)*DKDN*DKDB(J) 00004000
ISN 0348      DO 80 K=1,N                    00004010
ISN 0349      DFADA(J,K)=DFADA(J,K)+WT(I)*DKDA(J)*DKDA(K) 00004020
ISN 0350      DFBDB(J,K)=DFBDB(J,K)+WT(I)*DKDB(J)*DKDB(K) 00004030
ISN 0351      DFADB(J,K)=DFADB(J,K)+WT(I)*DKDA(J)*DKDB(K) 00004040
ISN 0352      80          CONTINUE            00004050
C             00004060
C             NOTE: DFBDA(J,K) IS NOT STORED SEPARATELY, SINCE DFBDA(J,K)= 00004070
C             DFADB(K,J).                     00004080
C             00004090
ISN 0353      DO 95 I=1,N                    00004100
ISN 0354      OBEQ (1,N3)=-FA(I)             00004110
ISN 0355      DO 85 J=1,N                    00004120
ISN 0356      OBEQ(1,J)=DFADA(I,J)           00004130
ISN 0357      85          CONTINUE            00004140
ISN 0358      IF (N.EQ.1) GO TO 91            00004150
ISN 0360      DO 90 J=1,NN                    00004160

```

LEVEL 2.3.0 (JUNE 78)	MAIN	OS/360 FORTRAN H EXTENDED	DATE 84.250/13.02.32	PAGE 9
ISN 0361		OBEQ(1,(J+N))=DFADB(I,(J+1))	00004170	
ISN 0362	90	CONTINUE	00004180	
ISN 0363	91	CONTINUE	00004190	
ISN 0364		IF (KODE1.EQ.0) OBEQ(1,N5)=DFADP(I)	00004200	
ISN 0366		IF (KODE1.LE.1) OBEQ(1,N6)=DFADL(I)	00004210	
ISN 0368		IF (KODE.EQ.0) OBEQ(1,N4)=DFADN(I)	00004220	
	C	WRITE (6,92) (OBEQ(1,J),J=1,16)	00004230	
	C	IF (N3.GT.16) WRITE (6,93) (OBEQ(1,J),J=17,N3)	00004240	
	C	92 FORMAT (1H0,2X,'OBEQ=' ,8F13.7/(1H,2X,8F13.7))	00004250	
	C	93 FORMAT (1H,2X,8F13.7/(1H,2X,8F13.7))	00004260	
ISN 0370		CALL NRME (OBEQ,1,N4,0,C,N4,1,0)	00004270	
ISN 0371	95	CONTINUE	00004280	
ISN 0372		K=1	00004290	
ISN 0373		IF (N.EQ.1) GO TO 111	00004300	
ISN 0375		DO 110 I=N1,N2	00004310	
ISN 0376		K=K+1	00004320	
ISN 0377		OBEQ(1,N3)=-FB(K)	00004330	
ISN 0378		DO 100 J=1,N	00004340	
ISN 0379		OBEQ(1,J)=DFADB(J,K)	00004350	
ISN 0380	100	CONTINUE	00004360	
ISN 0381		DO 105 J=1,NN	00004370	
ISN 0382		OBEQ(1,(J+N))=DFBDB(K,(J+1))	00004380	
ISN 0383	105	CONTINUE	00004390	
ISN 0384		IF (KODE1.EQ.0) OBEQ(1,N5)=DFBDP(K)	00004400	
ISN 0386		IF (KODE1.LE.1) OBEQ(1,N6)=DFBDL(K)	00004410	
ISN 0388		IF (KODE.EQ.0) OBEQ(1,N4)=DFBDN(K)	00004420	
	C	WRITE (6,92) (OBEQ(1,J),J=1,16)	00004430	
	C	IF (N3.GT.16) WRITE (6,93) (OBEQ(1,J),J=17,N3)	00004440	
ISN 0390		CALL NRME (OBEQ,1,N4,0,C,N4,1,0)	00004450	
ISN 0391	110	CONTINUE	00004460	
ISN 0392	111	CONTINUE	00004470	
ISN 0393		IF (KODE1.EQ.2) GO TO 119	00004480	
ISN 0395		IF (KODE1.EQ.1) GO TO 121	00004490	
ISN 0397		OBEQ(1,N3)=-FCHIP	00004500	
ISN 0398		DO 112 J=1,N	00004510	
ISN 0399		OBEQ(1,J)=DFADP(J)	00004520	
ISN 0400	112	CONTINUE	00004530	
ISN 0401		IF (N.EQ.1) GO TO 108	00004540	
ISN 0403		DO 113 J=1,NN	00004550	
ISN 0404		OBEQ(1,J+N)=DFBDP(J+1)	00004560	
ISN 0405	113	CONTINUE	00004570	
ISN 0406	108	CONTINUE	00004580	
ISN 0407		OBEQ(1,N5)=DFPDP	00004590	
ISN 0408		OBEQ(1,N6)=DFPDL	00004600	
ISN 0409		IF (KODE.EQ.0) OBEQ(1,N4)=DFPDN	00004610	
	C	WRITE (6,92) (OBEQ(1,J),J=1,16)	00004620	
	C	IF (N3.GT.16) WRITE (6,93) (OBEQ(1,J),J=17,N3)	00004630	
ISN 0411		CALL NRME (OBEQ,1,N4,0,C,N4,1,0)	00004640	
ISN 0412	121	CONTINUE	00004650	
ISN 0413		OBEQ(1,N3)=-FLATP	00004660	
ISN 0414		DO 114 J=1,N	00004670	
ISN 0415		OBEQ(1,J)=DFADL(J)	00004680	

LEVEL 2.3.0 (JUNE 78)

MAIN

OS/360 FORTRAN H EXTENDED

DATE 84.250/13.02.32

PAGE 10

```

ISN 0416      114 CONTINUE                                00004690
ISN 0417      IF (N.EQ.1) GO TO 109                        00004700
ISN 0419      DO 115 J=1,NN                                00004710
ISN 0420      OBEQ(1,J+N)=DFBDL(J+1)                       00004720
ISN 0421      115 CONTINUE                                00004730
ISN 0422      109 CONTINUE                                00004740
ISN 0423      IF (KODE1.EQ.0) OBEQ(1,N5)=DFPDL            00004750
ISN 0425      OBEQ(1,N6)=DFLDL                             00004760
ISN 0426      IF (KODE.EQ.0) OBEQ(1,N4)=DFLDN             00004770
C             WRITE (6,92) (OBEQ(1,J),J=1,16)            00004780
C             IF (N3.GT.16) WRITE (6,93) (OBEQ(1,J),J=17,N3) 00004790
ISN 0428      CALL NRME (OBEQ,1,N4,0,C,N4,1,0)            00004800
ISN 0429      119 CONTINUE                                00004810
ISN 0430      IF (KODE.NE.0) GO TO 120                    00004820
ISN 0432      OBEQ(1,N3)=-FN                               00004830
ISN 0433      DO 116 J=1,N                                00004840
ISN 0434      OBEQ(1,J)=DFADN(J)                          00004850
ISN 0435      116 CONTINUE                                00004860
ISN 0436      IF (N.EQ.1) GO TO 118                      00004870
ISN 0438      DO 117 J=1,NN                                00004880
ISN 0439      OBEQ(1,J+N)=DFBDN(J+1)                     00004890
ISN 0440      117 CONTINUE                                00004900
ISN 0441      118 CONTINUE                                00004910
ISN 0442      IF (KODE1.EQ.0) OBEQ(1,N5)=DFPDN            00004920
ISN 0444      IF (KODE1.LE.1) OBEQ(1,N6)=DFLDN           00004930
ISN 0446      OBEQ(1,N4)=DFNDN                             00004940
C             WRITE (6,92) (OBEQ(1,J),J=1,16)            00004950
C             IF (N3.GT.16) WRITE (6,93) (OBEQ(1,J),J=17,N3) 00004960
ISN 0447      CALL NRME (OBEQ,1,N4,0,C,N4,1,0)            00004970
ISN 0448      120 CONTINUE                                00004980
ISN 0449      CALL SMLE (C,N4,0,IERR)                     00004990
ISN 0450      IF (IERR.NE.0) WRITE (6,130) IERR           00005000
ISN 0452      IF (IERR.NE.0) GO TO 22                     00005010
ISN 0454      130 FORMAT (1H0,3X,'IERR=',I6,' CANNOT SOLVE USING THESE CONDITIONS.' 00005020
          *)                                              00005030
ISN 0455      E=0.D0                                      00005040
ISN 0456      IF (N4.LE.16) WRITE (6,132) (D(I),I=1,N4) 00005050
ISN 0458      IF (N4.GT.16) WRITE (6,132) (D(I),I=1,16) 00005060
ISN 0460      IF (N4.GT.16) WRITE (6,135) (D(I),I=17,N4) 00005070
ISN 0462      132 FORMAT (1H0,3X,8F13.7/(1H,3X,8F13.7)) 00005080
ISN 0463      135 FORMAT (1H,3X,8F13.7/(1H,3X,8F13.7)) 00005090
ISN 0464      DO 140 I=1,N4                               00005100
ISN 0465      E=E+DABS(D(I))                              00005110
ISN 0466      140 CONTINUE                                00005120
ISN 0467      DO 150 I=1,N                                00005130
ISN 0468      A(I)=A(I)+D(I)                              00005140
ISN 0469      150 CONTINUE                                00005150
ISN 0470      IF (N.EQ.1) GO TO 165                      00005160
ISN 0472      DO 160 I=2,N                                00005170
ISN 0473      B(I)=B(I)+D(I+N-1)                         00005180
ISN 0474      160 CONTINUE                                00005190
ISN 0475      165 CONTINUE                                00005200
```

LEVEL 2.3.0 (JUNE 78)

MAIN

OS/360 FORTRAN H EXTENDED

DATE 84.250/13.02.32

PAGE 11

```

ISM 0470      IF (KODE1.EQ.0) CHIP=CHIP*D(N5)                00005210
ISM 0478      IF (KODE1.LE.1) CLONP=CLONP*D(N6)              00005220
ISM 0480      IF (KODE.EQ.0) EN=EN*D(N4)                      00005230
ISM 0482      NE=NE+1                                         00005240
ISM 0483      IF (NE.GT.20) GO TO 480                         00005250
ISM 0485      IF (E.LT.1.D-5) GO TO 170                      00005260
ISM 0487      GO TO 42                                         00005270
ISM 0488      170 PHIP=CHIP*(E2/2.D0+5.D0+E4/24.D0+E6/12.D0)*DSIN(2.D0*CHIP)+(7.D0+E
    *4/48.D0+29.D0+E6/240.D0)*DSIN(4.D0*CHIP)+7.D0+E6*DSIN(6.D0*CHIP)/
    *120.D0                                                    00005280
ISM 0489      PHIPD=PHIP/DGRAD                                00005290
ISM 0490      CLONPD=CLONP/DGRAD                              00005300
ISM 0491      IF (N.GT.1) WRITE (6,180) N,N                  00005310
ISM 0493      180 FORMAT (1H0,3X,'COEFFICIENTS A(1) THROUGH A(' ,12,') , FOLLOWED BY
    *OEFF. B(2) THROUGH B(' ,12,') , LAT/LONG OF POLE, AND CONE CONST.') 00005320
ISM 0494      IF (N.EQ.1) WRITE (6,182) N                     00005330
ISM 0496      182 FORMAT (1H0,3X,'COEFFICIENTS A(1) THROUGH A(' ,12,') , FOLLOWED BY
    *AT/LONG OF POLE, AND CONE CONST.')                        00005340
ISM 0497      WRITE (6,190) (A(I),I=1,N)                      00005350
ISM 0498      IF (N.GT.1) WRITE (6,190) (B(I),I=2,N)          00005360
ISM 0500      WRITE (6,192) PHIPD,CLONPD,EN                   00005370
ISM 0501      190 FORMAT (1H0,2X,9F13.9/(1H ,2X,9F13.9))      00005380
ISM 0502      192 FORMAT (1H0,2X,3F13.5)                      00005390
ISM 0503      CHID=CHIR/DGRAD                                  00005400
ISM 0504      CLOND=CLONR/DGRAD                                00005410
ISM 0505      IF (DABS(EN).GT.1.D0-1.D-7) CHID=CHIP/DGRAD    00005420
ISM 0507      IF (DABS(EN).GT.1.D0-1.D-7) CLOND=CLONPD        00005430
ISM 0509      WRITE (6,194) CHID,CLOND                        00005440
ISM 0510      194 FORMAT (1H0,3X,'COEFFICIENTS ARE BASED ON USING ' ,F13.5,
    *' AS CONFORMAL LAT. AND ' ,F13.5, ' AS LONG. OF ORIGIN OF BASE PROJE
    *CTION.')                                                  00005450
ISM 0511      WRITE (6,195) E2                                  00005460
ISM 0512      195 FORMAT (1H0,3X,'SQUARE OF ECCENTRICITY OF REFERENCE ELLIPSOID=' ,F1
    *3.8)                                                       00005470
ISM 0513      RMS=DSQRT(SE/SMT)                                00005480
ISM 0514      WRITE (6,200) M,RMS                             00005490
ISM 0515      200 FORMAT (1H0,2X,'RMS ERROR IN SCALE FOR ' ,13, ' POINTS SELECTED=' ,
    *F14.7)                                                      00005500
ISM 0516      PHI1=TLAT                                        00005510
ISM 0517      CLON1=MLONG                                       00005520
ISM 0518      SP=DSIN(CHIP)                                    00005530
ISM 0519      CP=DCOS(CHIP)                                    00005540
ISM 0520      IF (DABS(EN).LT.1.D-7) GO TO 207                 00005550
ISM 0522      IF (DABS(EN).GT.1.D0-1.D-7) GO TO 208            00005560
C                                                           00005570
C                                                           00005580
C                                                           00005590
C                                                           00005600
C                                                           00005610
C                                                           00005620
C                                                           00005630
C                                                           00005640
C                                                           00005650
C                                                           00005660
C                                                           00005670
ISM 0524      SL=DSIN(CLONR-CLONP)                             00005680
ISM 0525      CL=DCOS(CLONR-CLONP)                             00005690
ISM 0526      SPP=DSIN(CHIR)                                   00005700
ISM 0527      CPP=DCOS(CHIR)                                   00005710
ISM 0528      C2=SP*SPP+CP*CPP*CL                             00005720

```

LEVEL 2.3.0 (JUNE 78)

MAIN

OS/360 FORTRAN H EXTENDED

DATE 84.250/13.02.32

PAGE 12

```

ISN 0529      SZ=DSQRT(1.D0-CZ**2)
ISN 0530      F=DSQRT((1.D0+EN)**(1.D0+EN)*(1.D0-EN)**(1.D0-EN))/EN
ISN 0531      RHO=F*((1.D0-CZ)/(1.D0+CZ))**((EN/2.D0)
ISN 0532      THP=EN*DATAN2(CPP*SL,CP*SPP-SP*CPP*CL)
ISN 0533      XPO=-RHO*DSIN(THP)
ISN 0534      YPO=-RHO*DCOS(THP)
ISN 0535      GO TO 209
ISN 0536      207 CONTINUE

C
C      COMPUTE COORDINATES OF CENTER FOR OBLIQUE MERCATOR.
C
ISN 0537      CLONO=CLONP+90.D0*DGRAD
ISN 0538      SL=DSIN(CLONR-CLONO)
ISN 0539      CL=DCOS(CLONR-CLONO)
ISN 0540      SPP=DSIN(CHIR)
ISN 0541      CPP=DCOS(CHIR)
ISN 0542      XPO=DATAN2(SPP*CP+CPP*SP*SL,CPP*CL)
ISN 0543      CZ=SP*SPP-CP*CPP*SL
ISN 0544      YPO=DLOG((1.D0+CZ)/(1.D0-CZ))/2.D0
ISN 0545      GO TO 209
ISN 0546      208 CONTINUE

C
C      COMPUTE COORDINATES OF CENTER FOR OBLIQUE STEREOGRAPHIC.
C
ISN 0547      XPO=0.D0
ISN 0548      YPO=0.D0
ISN 0549      209 CONTINUE
ISN 0550      IF (KODE2.NE.0) WRITE (6,210)
ISN 0552      210 FORMAT (1H0,3X,'      LONG.      LAT.      K'//)
ISN 0553      IF (KODE2.EQ.0) WRITE (6,211)
ISN 0555      211 FORMAT (1H0,3X,'      LONG.      LAT.      K      X
      *Y'//)
ISN 0556      214 CONTINUE
ISN 0557      PHI1=PHI1*DGRAD
ISN 0558      CLON1=CLON1*DGRAD
ISN 0559      IF (KODE2.EQ.0) GO TO 215
ISN 0561      DO 265 I=1,M
ISN 0562      215 IF (DABS(EN).LT.1.D-7) GO TO 220
ISN 0564      IF (DABS(EN).GT.1.D-7) GO TO 230
ISN 0566      IF (KODE2.EQ.0) GO TO 212
ISN 0568      CL=DCOS(CLON(I)-CLONP)
ISN 0569      ES=DSIN(PHI(I))*ECC
ISN 0570      CPH=DCOS(PHI(I))
ISN 0571      SPP=DSIN(CHI(I))
ISN 0572      CPP=DCOS(CHI(I))
ISN 0573      SL=DSIN(CLON(I)-CLONP)
ISN 0574      GO TO 213
ISN 0575      212 ES=DSIN(PHI1)*ECC
ISN 0576      CPH=DCOS(PHI1)
ISN 0577      CHI1=2.D0*DATAN(DTAN(PI/4.D0+PHI1/2.D0)*((1.D0-ES)/(1.D0+ES))*
      *(ECC/2.D0))-PI/2.D0
ISN 0578      SL=DSIN(CLON1-CLONP)

```

```

00005730
00005740
00005750
00005760
00005770
00005780
00005790
00005800
00005810
00005820
00005830
00005840
00005850
00005860
00005870
00005880
00005890
00005900
00005910
00005920
00005930
00005940
00005950
00005960
00005970
00005980
00005990
00006000
00006010
00006020
00006030
00006040
00006050
00006060
00006070
00006080
00006090
00006100
00006110
00006120
00006130
00006140
00006150
00006160
00006170
00006180
00006190
00006200
00006210
00006220
00006230
00006240

```

LEVEL 2.3.0 (JUNE 78)

MAIN

OS/360 FORTRAN H EXTENDED

DATE 84.250/13.02.32

PAGE 13

```
ISN 0579      CL=DCOS(CLON1-CLONP)      00006250
ISN 0580      SPP=DSIN(CHI1)            00006260
ISN 0581      CPP=DCOS(CHI1)            00006270
C'           00006280
C           COMPUTE COORDINATES FOR FINAL OBLIQUE CONFORMAL CONIC. 00006290
C           00006300
ISN 0582      213 CZ=SP*SPP+CP*CPP*CL    00006310
ISN 0583      SZ=DSQRT(1.D0-CZ**2)      00006320
ISN 0584      RHO=F*((1.D0-CZ)/(1.D0+CZ))* (EN/2.D0) 00006330
ISN 0585      THP=EN*DATAN2(CPP*SL,CP*SPP-SP*CPP*CL) 00006340
ISN 0586      XP=-RHO*DSIN(THP)-XP0    00006350
ISN 0587      YP=-RHO*DCOS(THP)-YP0    00006360
ISN 0588      R=DSQRT(XP**2+YP**2)      00006370
ISN 0589      SKPR=EN*RHO*CPP*DSQRT(1.D0-ES**2)/SZ/CPH 00006380
ISN 0590      GO TO 240                  00006390
ISN 0591      220 CONTINUE                00006400
ISN 0592      IF (KODE2.EQ.0) GO TO 222  00006410
ISN 0594      SL=DSIN(CLON(I)-CLON0)    00006420
ISN 0595      CL=DCOS(CLON(I)-CLON0)    00006430
ISN 0596      SPP=DSIN(CHI(I))          00006440
ISN 0597      CPP=DCOS(CHI(I))          00006450
ISN 0598      ESQ=E2*DSIN(PHI(I))*2     00006460
ISN 0599      CPH=DCOS(PHI(I))          00006470
ISN 0600      GO TO 223                  00006480
ISN 0601      222 SL=DSIN(CLON1-CLON0)  00006490
ISN 0602      CL=DCOS(CLON1-CLON0)      00006500
ISN 0603      ES=DSIN(PHI1)*ECC         00006510
ISN 0604      CHI1=2.D0*DATAN(DTAN(PI/4.D0+PHI1/2.D0)*((1.D0-ES)/(1.D0+ES))* 00006520
            * (ECC/2.D0))-PI/2.D0      00006530
ISN 0605      SPP=DSIN(CHI1)            00006540
ISN 0606      CPP=DCOS(CHI1)            00006550
ISN 0607      ESQ=E2*DSIN(PHI1)*2       00006560
ISN 0608      CPH=DCOS(PHI1)            00006570
C           00006580
C           COMPUTE COORDINATES FOR FINAL OBLIQUE MERCATOR. 00006590
C           00006600
ISN 0609      223 XP=DATAN2(SPP*CP+CPP*SP*SL,CPP*CL)-XP0 00006610
ISN 0610      CZ=SP*SPP-CP*CPP*SL      00006620
ISN 0611      YP=DLOG((1.D0+CZ)/(1.D0-CZ))/2.D0-YP0 00006630
ISN 0612      SKPR=CPP*DSQRT(1.D0-ESQ)/DSQRT(1.D0-CZ**2)/CPH 00006640
ISN 0613      GO TO 240                  00006650
ISN 0614      230 CONTINUE                00006660
ISN 0615      IF (KODE2.EQ.0) GO TO 232  00006670
ISN 0617      SL=DSIN(CLON(I)-CLONP)    00006680
ISN 0618      CL=DCOS(CLON(I)-CLONP)    00006690
ISN 0619      SPP=DSIN(CHI(I))          00006700
ISN 0620      CPP=DCOS(CHI(I))          00006710
ISN 0621      ESQ=E2*DSIN(PHI(I))*2     00006720
ISN 0622      CPH=DCOS(PHI(I))          00006730
ISN 0623      GO TO 233                  00006740
ISN 0624      232 SL=DSIN(CLON1-CLONP)  00006750
ISN 0625      CL=DCOS(CLON1-CLONP)      00006760
```

PAGE 14

ISN 0626	ES=DSIN(PHI1)*ECC	00006770
ISN 0627	CHI1=2.00*DATAN(OTAN(PI/4.00+PHI1/2.00)*((1.00-ES)/(1.00+ES))* *(ECC/2.00))-PI/2.00	00006780
		00006790
ISN 0628	SPP=DSIN(CHI1)	00006800
ISN 0629	CPP=DCOS(CHI1)	00006810
ISN 0630	ESQ=E2*DSIN(PHI1)**2	00006820
ISN 0631	CPH=DCOS(PHI1)	00006830
		00006840
	C	00006850
	C	00006860
	C	00006870
ISN 0632	233 CZ=SP*SPP+CP*CPP*CL	00006880
ISN 0633	SZ=DSQRT(1.00-CZ**2)	00006890
ISN 0634	S=2.00/(1.00+CZ)	00006900
ISN 0635	SKPR=S*CPP/(CPH/DSQRT(1.00-ESQ))	00006910
ISN 0636	XP=S*CPP*SL	00006920
ISN 0637	YP=S*(CP*SPP-SP*CPP*CL)	00006930
ISN 0638	240 CONTINUE	00006940
		00006950
	C	00006960
	C	00006970
	C	00006980
	C	00006990
	C	00007000
ISN 0639	IF (N.EQ.1) GO TO 263	00007010
ISN 0641	AR=A(N)	00007020
ISN 0642	AI=B(N)	00007030
ISN 0643	BR=A(N-1)	00007040
ISN 0644	BI=B(N-1)	00007050
ISN 0645	CR=DFLOAT(N)*A(N)	00007060
ISN 0646	CI=DFLOAT(N)*B(N)	00007070
ISN 0647	DR=DFLOAT(N-1)*A(N-1)	00007080
ISN 0648	DI=DFLOAT(N-1)*B(N-1)	00007090
ISN 0649	R=XP*XP	00007100
ISN 0650	S=XP*XP+YP*YP	00007110
ISN 0651	DO 260 J=2,N	00007120
ISN 0652	ARN=BR+R*AR	00007130
ISN 0653	AIN=BI+R*AI	00007140
ISN 0654	IF (J.EQ.N) GO TO 260	00007150
ISN 0656	BR=A(N-J)-S*AR	00007160
ISN 0657	BI=B(N-J)-S*AI	00007170
ISN 0658	AR=ARN	00007180
ISN 0659	AI=AIN	00007190
ISN 0660	CRN=DR+R*CR	00007200
ISN 0661	CIN=DI+R*CI	00007210
ISN 0662	DR=DFLOAT(N-J)*A(N-J)-S*CR	00007220
ISN 0663	DI=DFLOAT(N-J)*B(N-J)-S*CI	00007230
ISN 0664	CR=CRN	00007240
ISN 0665	CI=CIN	00007250
ISN 0666	260 CONTINUE	00007260
ISN 0667	BR=-S*AR	00007270
ISN 0668	BI=-S*AI	00007280
ISN 0669	AR=ARN	

LEVEL 2.3.0 (JUNE 78)

MAIN

OS/360 FORTRAN H EXTENDED

DATE 84.250/13.02.32

PAGE 15

```

ISN 0670      AI=AIN                      00007290
ISN 0671      X=XP*AR-YP*AI+BR           00007300
ISN 0672      Y=YP*AR+XP*AI+BI           00007310
ISN 0673      U=XP*CR-YP*CI+DR           00007320
ISN 0674      V=YP*CR+XP*CI+DI           00007330
ISN 0675      SK=DSQRT(U*U+V*V)*SKPR      00007340
ISN 0676      IF (KODE2.EQ.0) GO TO 268   00007350
ISN 0678      GO TO 264                   00007360
ISN 0679      263 CONTINUE                00007370
ISN 0680      X=XP                        00007380
ISN 0681      Y=YP                        00007390
ISN 0682      SK=SKPR*A(1)                00007400
ISN 0683      IF (KODE2.EQ.0) GO TO 268   00007410
ISN 0685      264 CONTINUE                00007420
ISN 0686      PHI1=PHI(I)/DGRAD           00007430
ISN 0687      CLON1=CLON(I)/DGRAD         00007440
ISN 0688      WRITE (6,270) CLON1,PHI1,SK 00007450
ISN 0689      270 FORMAT (1H,2X,3F11.5)   00007460
ISN 0690      265 CONTINUE                00007470
ISN 0691      GO TO 282                   00007480
ISN 0692      268 PHI1=PHI1/DGRAD         00007490
ISN 0693      CLON1=CLON1/DGRAD           00007500
ISN 0694      WRITE (6,271) CLON1,PHI1,SK,X,Y 00007510
ISN 0695      271 FORMAT (1H,2X,5F11.5)   00007520
ISN 0696      PHI1=PHI1-DLAT              00007530
ISN 0697      IF (PHI1.GT.BLAT-0.01D0) GO TO 214 00007540
ISN 0699      PHI1=TLAT                   00007550
ISN 0700      CLON1=CLON1+DLONG           00007560
ISN 0701      WRITE (6,280)                00007570
ISN 0702      IF (CLON1.LE.ELONG+0.01D0) GO TO 214 00007580
ISN 0704      280 FORMAT (1H)              00007590
ISN 0705      282 GO TO 22                 00007600
ISN 0706      480 WRITE (6,490)            00007610
ISN 0707      490 FORMAT (1H0,' TOO MANY ITERATIONS.') 00007620
ISN 0708      GO TO 22                     00007630
ISN 0709      510 WRITE (6,520)            00007640
ISN 0710      520 FORMAT (1H0,2X,'NORMAL END OF RUN.') 00007650
ISN 0711      STOP                        00007660
ISN 0712      END                          00007670

```

/ MAIN /

SIZE OF PROGRAM 0090E0 HEXADECIMAL BYTES

NAME	TAG	TYPE	ADD.	NAME	TAG	TYPE	ADD.	NAME	TAG	TYPE	ADD.	NAME	TAG	TYPE	ADD.
A SF		R*8	0010D8	B SF		R*8	001150	C SFA		R*8	0011C8	D SFA		R*8	002168
E SF		R*8	000D30	F SF		R*8	000D38	I SFA		I*4	000CEC	J SFA		I*4	000CF0
K SF		I*4	000CF4	M SF		I*4	000CF8	N SFA		I*4	000CFC	R SF		R*8	000D40
S SF		R*8	000D48	U SFA		R*8	000D50	V SFA		R*8	000D58	X SF		R*8	000D60
Y SF		R*8	000D68	AI SF		R*8	000D70	AR SF		R*8	000D78	AZ SFA		R*8	000D80
BI SF		R*8	000D88	BR SF		R*8	000D90	CC SF		R*8	000D98	CI SF		R*8	000DA0
CL SFA		R*8	000DA8	CP SFA		R*8	000DB0	CR SF		R*8	000DB8	CZ SFA		R*8	000DC0
DD SF		R*8	000DC8	DI SF		R*8	000DD0	DR SF		R*8	000DD8	EE SF		R*8	000DE0

LEVEL 2.3.0 (JUNE 78)

MAIN

OS/360 FORTRAN H EXTENDED

DATE 24.250/13.02.32

PAGE 16

EN SFA	R*8	000DE8	ES SFA	R*8	000DF0	E2 SFA	R*8	000DF8	E4 SF	R*8	000E00
E6 SF	R*8	000E08	FA SF	R*8	002258	FB SF	R*8	0022D0	FM SF	R*8	000E10
F1 SFA	R*8	000E18	F2 SFA	R*8	000E20	GG SF	R*8	000E28	HM SF	R*8	000E30
HJ SF	R*8	000E38	NE SF	I*4	000D00	NN SF	I*4	000D04	N1 SF	I*4	000D08
N2 SF	I*4	000D0C	N3 SF	I*4	000D10	N4 SFA	I*4	000D14	N5 SF	I*4	000D18
N6 SF	I*4	000D1C	PI SFA	R*8	000E40	SE SFA	R*8	000E48	SK SF	R*8	000E50
SL SFA	R*8	000E58	SP SFA	R*8	000E60	SZ SF	R*8	000E68	TH SFA	R*8	000E70
WT SF	R*8	002348	XP SFA	R*8	000E78	YP SFA	R*8	000E80	AIN SF	R*8	000E88
ARN SF	R*8	000E90	CAZ SF	R*8	000E98	CHI SFA	R*8	002888	CIN SF	R*8	000EA0
CPH SF	R*8	000EA8	CPP SFA	R*8	000E80	CRN SF	R*8	000E88	CTH SF	R*8	000EC0
ECC SFA	R*8	000EC8	ESQ SFA	R*8	000ED0	FAC SF	R*8	000ED8	PHI SFA	R*8	002DC8
RHO SF	R*8	000EE0	RMS SF	R*8	000EE8	SPP SFA	R*8	000EF0	STH SF	R*8	000EF8
SWT SFA	R*8	000F00	THP SFA	R*8	000F08	XPO SF	R*8	000F10	YPO SF	R*8	000F18
BLAT S	R*8	000F20	CHID SF	R*8	000F28	CHIP SFA	R*8	000F30	CHIR SFA	R*8	000F38
CHI1 SFA	R*8	000F40	CLON SFA	R*8	003308	CTHP SF	R*8	000F48	CTH2 SF	R*8	000F50
DCOS F	XF	000000	DF1A SF	R*8	003848	DF1L SF	R*8	000F58	DF1N SF	R*8	000F60
DF1P SF	R*8	000F68	DF2A SF	R*8	0038C0	DF2L SF	R*8	000F70	DF2N SF	R*8	000F78
DF2P SF	R*8	000F80	DKDA SF	R*8	003938	DKDB SF	R*8	0039J0	DKDL SF	R*8	000F88
DKDN SF	R*8	000F90	DKDP SF	R*8	000F98	DLAT SF	R*8	000FA0	DLOG F	XF	000000
DRDL SF	R*8	000FA8	DRDN SF	R*8	000F80	DRDP SF	R*8	000F88	DSIN FA	XF	000000
DTAN FA	XF	000000	DTDL SF	R*8	000FC0	DTDN SF	R*8	000FC8	DTDP SF	R*8	000FD0
FAC1 SFA	R*8	000FD8	FAC2 SFA	R*8	000FE0	IERR SFA	I*4	000D20	KODE SF	I*4	000D24
NRME SF	XF	000000	OBEQ SFA	R*8	003A28	PHIP SFA	R*8	000FE8	PHI1 SFA	R*8	000FF0
RPOW SF	R*8	000FF8	SKPR SF	R*8	001000	SMLE SF	XF	000000	STHP SF	R*8	001008
STH2 SF	R*8	001010	TLAT SF	R*8	001018	CLOND SF	R*8	001020	CLONO SFA	R*8	001028
CLONP SFA	R*8	001030	CLONR SFA	R*8	001038	CLON1 SFA	R*8	001040	DATAN F	XF	000000
DFADA SF	R*8	003B18	DFADB SF	R*8	004220	DFADL SF	R*8	004928	DFADM SF	R*8	0049A0
DFADP SF	R*8	004A18	DFBDB SF	R*8	004A90	DFBDL SF	R*8	005198	DFBDN SF	R*8	005210
DFBDP SF	R*8	005288	DFLDL SF	R*8	001048	DFLDN SF	R*8	001050	DFNDN SF	R*8	001058
DFPDL SF	R*8	001060	DFPDN SF	R*8	001068	DFPDP SF	R*8	001070	DGRAD SF	R*8	001078
DKPDL SF	R*8	001080	DKPDN SF	R*8	001088	DKPDP SF	R*8	001090	DLONG SF	R*8	001098
DSQRT F	XF	000000	ELGNG S	R*8	0010A0	FCHIP SF	R*8	0010A8	FLATP SF	R*8	0010B0
KODE1 SF	I*4	000D28	KODE2 S	I*4	000D2C	PHIPD SF	R*8	0010B8	RPOW2 SF	R*8	0010C0
WLONG SF	R*8	0010C8	FDXPI#	XF	000000	FDXPD#	XF	000000	CLONPD SF	R*8	0010D0
DATAN2 F	XF	000000	IBCOM#	F	XF	I*4	000000				

SOURCE STATEMENT LABELS

LABEL	ISN	ADDR
20	28	0055A0
419	111	005B78
431	144	005E62
451	231	0066F6
48	278	006AA8
50	305	006E00
90	362	00754C
105	383	0076BC
113	405	007814
115	421	007902
117	440	007A08

LABEL	ISN	ADDR
22	29	0055B4
42	112	005B78
44	150	005EA4
452	250	006904
49	286	006AF8
60	314	006F68
91	363	007560
110	391	00775E
108	406	007828
109	422	007916
118	441	007A1C

LABEL	ISN	ADDR
40	68	0058B4
43	131	005D6E
45	164	005FA6
46	258	006958
482	297	006D12
80	352	007466
95	371	0075EE
111	392	007772
121	412	007878
119	429	007974
120	448	007A7C

LABEL	ISN	ADDR
418	92	005A48
432	141	005E56
453	213	00642C
47	259	006960
491	302	006D04
85	357	0074F6
100	380	00766E
112	400	0077C8
114	416	0078B6
116	435	0079BC
140	466	007B90

LEVEL 2.3.0 (JUNE 78)

MAIN

OS/360 FORTRAN H EXTENDED

DATE 84.250/13.02.32

PAGE 17

150 469 007B00
 207 536 00816A
 215 562 0082C6
 222 601 008740
 233 632 008B66
 264 685 008F58
 480 706 009072

160 474 007C24
 208 546 008252
 212 575 0083E6
 223 609 008866
 240 638 008C10
 265 690 008F80
 510 709 00908C

165 475 007C38
 209 549 00825E
 213 582 0084F0
 230 614 00893C
 260 666 008E4C
 268 692 008FC8

170 488 007CB8
 214 556 00829C
 220 591 00863C
 232 624 008A40
 263 679 008F32
 282 705 00906E

COMPILER GENERATED LABELS

LABEL	ISN	ADDR
100000	1	005364
100004	43	005722
100007	46	005752
200004	49	005780
100012	51	0057AC
100015	54	00570A
100019	62	00583C
200007	74	0058EE
100025	77	005932
100028	82	005994
200011	86	0059D6
100033	89	005A1E
100036	96	005A78
200015	100	005AB6
100041	103	005AF4
100044	108	005B44
100047	119	005BBE
100051	161	005F4C
100055	177	00600C
200021	224	006624
100059	275	006A68
100063	282	006AD0
100067	293	006BA2
100071	315	006F7C
100075	323	00708C
100079	327	0070DE
100083	331	007130
100087	335	007180
100091	339	0071DE
100095	343	00727A
100099	347	007312
100103	353	00748E
100107	358	00750A
100112	366	00758A
100116	370	0075E0
100120	379	007648
100124	385	0076DA
100128	389	007730
100133	399	0077A8
100138	410	007856
100142	419	0078D6

LABEL	ISN	ADDR
100001	20	005410
100005	44	005730
200003	47	00575C
100010	49	00578C
100013	52	0057B6
100016	59	005818
100020	66	005894
100023	74	0058F8
100026	79	005958
200010	83	00599E
100031	86	0059E2
100034	91	005A44
200014	97	005A82
100039	100	005AC2
100042	105	005B10
200018	109	005B4E
100048	149	005E94
100052	165	005FBA
100056	179	006020
200022	248	0068E8
100060	279	006AB4
100064	283	006ADC
100068	300	006D1C
100072	318	006FE6
100076	324	007096
100080	328	0070E8
100084	332	00713A
100088	336	007188
100092	340	007220
100096	344	00728C
100100	348	007354
100104	353	0074A2
100108	360	007516
100113	367	007596
100117	372	007602
100121	381	007682
100125	386	0076FA
100129	390	007750
100134	401	0077DC
100139	411	00786A
100143	420	0078DE

LABEL	ISN	ADDR
100003	34	00560A
200002	45	00573A
100008	47	005768
100011	50	005796
200006	53	0057C0
100017	60	005822
100021	69	0058C8
100024	76	00591C
200009	80	005962
100029	83	0059A8
100032	88	005A08
200013	94	005A52
100037	97	005A8E
100040	102	005AE0
200017	106	005B1A
100045	109	005B5A
100049	151	005EB8
100053	165	005FCE
200019	192	006218
100057	256	006942
100061	280	006ABC
100065	284	006AE4
100069	306	006E14
100073	319	007016
100077	325	007084
100081	329	007106
100085	333	007158
100089	337	007192
100093	341	00722C
100097	345	0072C6
100101	349	00735C
100105	354	0074AA
100109	361	00751E
100114	368	007586
100118	375	007612
100122	382	00768A
100126	387	007706
100131	395	00777E
100135	403	0077E8
100140	415	007896
100145	424	007920

LABEL	ISN	ADDR
200001	43	005718
100006	45	005744
100009	48	005776
200005	51	0057A0
100014	53	0057CC
100018	61	00582C
100022	73	0058E4
200008	77	005926
100027	80	00596E
100030	85	0059CC
200012	89	005A12
100035	94	005A5C
100038	99	005AAC
200016	103	005AEA
100043	106	005B26
100046	117	005BAA
100050	152	005EC0
100054	175	005FFE
200020	210	0063F0
100058	269	00698C
100062	281	006AC8
100066	285	006AF0
100070	310	006E78
100074	322	007068
100078	326	0070C0
100082	330	007112
100086	334	007162
100090	338	0071D4
100094	342	00726E
100098	346	007308
100102	353	00747A
100106	356	0074D0
100111	365	00756A
100115	369	0075C0
100119	376	00761A
100123	384	0076D0
100127	388	007726
100132	397	00778A
100136	404	0077F0
100141	417	0078CA
100146	425	007934

LEVEL 2.3.0 (JUNE 78)

MAIN

OS/360 FORTRAN H EXTENDED

DATE 84.250/13.02.32

PAGE 18

100147	427	007952	100148	428	007966	100149	432	00797E	100150	434	00799C
100151	436	0079D0	100152	438	0079DC	100153	439	0079E4	100155	443	007A26
100156	444	007A3A	100157	445	007A46	100158	446	007A5A	100159	451	007A94
100160	452	007AB0	100161	455	007ABA	100162	457	007ACE	100165	458	007B00
100166	459	007B0C	100169	460	007B34	100170	461	007B40	100173	464	007B70
100174	465	007B78	100175	467	007BA4	100176	468	007BAC	100177	470	007BE4
100178	472	007BF0	100179	473	007BF8	100181	477	007C42	100182	478	007C56
100183	479	007C62	100184	480	007C76	100185	481	007C80	100186	482	007C94
100187	485	007CA8	100188	487	007CB4	100189	492	007DA8	100190	494	007DCC
100191	495	007D08	100192	497	007DF4	100195	499	007E30	100198	500	007E60
100199	506	007EB8	100200	507	007EC4	100201	508	007ED8	100202	509	007EE0
100203	522	007FA6	100204	524	007FBA	100205	551	008268	100206	553	00827C
100207	554	008286	100208	561	0082BE	100209	564	0082D4	100210	566	0082E8
100211	568	0082F2	100212	594	008646	100213	617	008946	100214	641	008C1C
100215	652	008D24	100216	656	008D54	100217	667	008E60	100218	678	008F2E
100219	691	008FC4	100220	699	009036						

FORMAT STATEMENT LABELS

LABEL	ISN	ADDR	LABEL	ISN	ADDR	LABEL	ISN	ADDR	LABEL	ISN	ADDR
5	9	000028	6	16	000033	10	21	0000B3	15	23	0000BE
25	31	0000D2	30	35	0000E1	32	37	0001BA	35	64	000241
41	75	00024A	421	78	0002B8	411	81	000322	412	84	00037E
422	87	0003DF	413	90	00043C	414	95	00048B	423	98	0004EB
415	101	000547	416	104	000595	424	107	0005E8	417	110	000637
130	454	000674	132	462	0006AC	135	463	0006C6	180	493	0006E0
182	496	000756	190	501	0007AF	192	502	0007C9	194	510	0007D5
195	512	000846	200	515	000880	210	552	0008B8	211	555	0008E0
270	689	00091D	271	695	000929	280	704	000935	490	707	00093A
520	710	000957									

*OPTIONS IN EFFECT*NAME(MAIN) OPTIMIZE(1) LINECOUNT(54) SIZE(MAX) AUTODBL(NONE)

*OPTIONS IN EFFECT*SOURCE EBCDIC NOLIST NODECK OBJECT MAP NOFORMAT GOSTMT NOXREF NOALC NOANSF NOTERM IBM FLAG(I)

STATISTICS SOURCE STATEMENTS = 711, PROGRAM SIZE = 37088, SUBPROGRAM NAME = MAIN

STATISTICS NO DIAGNOSTICS GENERATED

***** END OF COMPILATION *****

114K BYTES OF CORE NOT USED

LEVEL 2.3.0 (JUNE 78)

OS/360 FORTRAN H EXTENDED

DATE 84.250/13.02.36

PAGE 1

REQUESTED OPTIONS:

OPTIONS IN EFFECT: NAME(MAIN) OPTIMIZE(1) LINECOUNT(54) SIZE(MAX) AUTODBL(NONE)
SOURCE EBCDIC NOLIST NODECK OBJECT MAP NOFORMAT GOSTMT NOXREF NOALC NOANSF NOTERM IBM FLAG(I)

```

ISN 0002      SUBROUTINE NRME (OBEQ,N,M,M1,A,NT,ICOL,JCOL)      00007680
ISN 0003      REAL*8 OBEQ,A      00007690
ISN 0004      DIMENSION OBEQ(1,30),A(1)      00007700
ISN 0005      II=(ICOL-1)*(2*NT+4-ICOL)/2      00007710
ISN 0006      IK=(ICOL*(2*NT+3-ICOL)/2)      00007720
ISN 0007      MM=M+M1+1      00007730
ISN 0008      IJ=II+JCOL-ICOL      00007740
ISN 0009      DO 206 IM=1,M      00007750
ISN 0010      ICT=II      00007760
ISN 0011      IIK=IK      00007770
ISN 0012      DO 201 IM=1,M      00007780
ISN 0013      DO 200 JM=IM,M      00007790
ISN 0014      ICT=ICT+1      00007800
ISN 0015      200 A(ICT)=A(ICT)+OBEQ(IM,IM)*OBEQ(IM,JM)      00007810
ISN 0016      A(IIK)=A(IIK)+OBEQ(IM,IM)*OBEQ(IM,MM)      00007820
ISN 0017      ICT=(ICOL+IM-1)*(2*NT+4-ICOL-IM)/2      00007830
ISN 0018      201 IIK=ICT-ICOL-IM+NT+2      00007840
ISN 0019      IF (M1) 206,206,202      00007850
ISN 0020      202 KM=M+1      00007860
ISN 0021      KM1=M+M1      00007870
ISN 0022      IIJ=IJ      00007880
ISN 0023      DO 203 IM=1,M      00007890
ISN 0024      DO 2202 JM=KM,KM1      00007900
ISN 0025      IIJ=IIJ+1      00007910
ISN 0026      2202 A(IIJ)=A(IIJ)+OBEQ(IM,IM)*OBEQ(IM,JM)      00007920
ISN 0027      203 IIJ=IJ+((2*NT+3-2*ICOL-IM)*IM)/2      00007930
ISN 0028      JJ=(JCOL-1)*(2*NT+4-JCOL)/2      00007940
ISN 0029      JK=(JCOL*(2*NT+3-JCOL)/2)      00007950
ISN 0030      ICT=JJ      00007960
ISN 0031      IIK=JK      00007970
ISN 0032      DO 205 LM=KM,KM1      00007980
ISN 0033      DO 204 JM=LM,KM1      00007990
ISN 0034      ICT=ICT+1      00008000
ISN 0035      204 A(ICT)=A(ICT)+OBEQ(IM,LM)*OBEQ(IM,JM)      00008010
ISN 0036      A(IIK)=A(IIK)+OBEQ(IM,LM)*OBEQ(IM,MM)      00008020
ISN 0037      ICT=(JCOL+LM-KM)*(2*NT-JCOL-LM+KM+3)/2      00008030
ISN 0038      205 IIK=ICT-JCOL-LM+KM+NT+1      00008040
ISN 0039      206 CONTINUE      00008050
ISN 0040      RETURN      00008060
ISN 0041      END      00008070

```

				/	NRME /				SIZE OF PROGRAM 00052C HEXADECIMAL BYTES						
NAME	TAG	TYPE	ADD.	NAME	TAG	TYPE	ADD.	NAME	TAG	TYPE	ADD.	NAME	TAG	TYPE	ADD.
A SF	XR	R*8	000000	H F		I*4	000094	N F		I*4	000098	II SF		I*4	00009C
IJ SF		I*4	0000A0	IK SF		I*4	0000A4	IM SF		I*4	0000A8	IN SF		I*4	0000AC

LEVEL 2.3.0 (JUNE 78)

NRME

OS/360 FORTRAN H EXTENDED

DATE 84.250/13.02.36

PAGE 2

JJ SF	I*4	0000B0	JK SF	I*4	0000B4	JM SF	I*4	0000B8	KM SF	I*4	0000BC
LM SF	I*4	0000C0	MM SF	I*4	0000C4	M1 F	I*4	0000C8	NT F	I*4	0000CC
ICT SF	I*4	0000D0	I1J SF	I*4	0000D4	I1K SF	I*4	0000D8	KM1 SF	I*4	0000DC
ICOL F	I*4	0000E0	JCOL F	I*4	0000E4	NRME	I*4	0000E8	OBEQ F XR	R*8	000000

SOURCE STATEMENT LABELS

LABEL	ISN	ADDR	LABEL	ISN	ADDR	LABEL	ISN	ADDR	LABEL	ISN	ADDR
200	15	00019E	201	18	00025A	202	20	000282	2202	26	0002B0
203	27	0002FA	204	35	000398	205	38	000462	206	39	00048C

COMPILER GENERATED LABELS

LABEL	ISN	ADDR	LABEL	ISN	ADDR	LABEL	ISN	ADDR	LABEL	ISN	ADDR
100001	2	000108	100002	10	000180	100003	13	000192	100004	14	000194
100005	16	0001E8	100006	19	000278	100007	24	0002A2	100008	25	0002A6
100010	28	000330	100011	33	00038A	100012	34	00038E	100013	36	0003E4
100015	40	00049E									

*OPTIONS IN EFFECT*NAME(MAIN) OPTIMIZE(1) LINECOUNT(54) SIZE(MAX) AUTODBL(NONE)

*OPTIONS IN EFFECT*SOURCE EBCDIC NOLIST NODECK OBJECT MAP NOFORMAT GOSTMT NOXREF NOALC NOANSF MOTERM IBM FLAG(I)

STATISTICS SOURCE STATEMENTS = 40, PROGRAM SIZE = 1324, SUBPROGRAM NAME = NRME

STATISTICS NO DIAGNOSTICS GENERATED

***** END OF COMPILATION *****

258K BYTES OF CORE NOT USED

LEVEL 2.3.0 (JUNE 78)

OS/360 FORTRAN H EXTENDED

DATE 84.250/13.02.37

PAGE 1

REQUESTED OPTIONS:

OPTIONS IN EFFECT: NAME(MAIN) OPTIMIZE(1) LINECOUNT(54) SIZE(MAX) AUTODBL(NONE)
SOURCE EBCDIC NOLIST NODECK OBJECT MAP NOFORMAT GOSTMT NOXREF NOALC NOANSF NOTERM IBM FLAG(I)

```
ISN 0002      SUBROUTINE SMLE (A,N,B,IERR)                00008080
ISN 0003      DIMENSION A(1),B(1)                        00008090
ISN 0004      REAL*8 A,B,FAC,FMUL                        00008100
C                                                     00008110
C      BEGIN FORWARD SOLUTION                            00008120
C                                                     00008130
ISN 0005      N1=N+1                                     00008140
ISN 0006      DO 5 I=1,N                                 00008150
ISN 0007      II=1+((I-1)*(2*N+4-I))/2 .                00008160
ISN 0008      IF (A(II)) 1,1,2                          00008170
ISN 0009      1 IERR=II                                  00008180
ISN 0010      GO TO 10                                   00008190
ISN 0011      2 FMUL=1.D0/A(II)                          00008200
ISN 0012      JCOL=I+1                                  00008210
ISN 0013      DO 5 J=JCOL,N1                             00008220
ISN 0014      IIJI=II+J-I                               00008230
ISN 0015      FAC=A(IIJI)                               00008240
ISN 0016      A(IIJI)=FAC*FMUL                          00008250
ISN 0017      JJ=J                                       00008260
ISN 0018      IF (JCOL-N1) 2002,5,5                     00008270
ISN 0019      2002 IF (N-JJ) 1002,3,3                    00008280
ISN 0020      1002 JJ=N                                  00008290
ISN 0021      3 DO 4 K=JCOL,JJ                          00008300
ISN 0022      KJ=1+J-K+((K-1)*(2*N+4-K))/2              00008310
ISN 0023      IIKI=II+K-I                               00008320
ISN 0024      4 A(KJ)=A(KJ)-FAC*A(IIKI)                 00008330
ISN 0025      5 CONTINUE                                00008340
C                                                     00008350
C      BEGIN BACK SOLUTION                              00008360
C                                                     00008370
ISN 0026      I=N+1                                      00008380
ISN 0027      6 I=I-1                                    00008390
ISN 0028      IJ=(I*(2*N+3-I))/2                        00008400
ISN 0029      B(I)=A(IJ)                                00008410
ISN 0030      J=N                                       00008420
ISN 0031      L=2+((I-1)*(2*N+4-I))/2                  00008430
ISN 0032      7 IJ=IJ-1                                 00008440
ISN 0033      IF (L-IJ) 1007,1007,8                    00008450
ISN 0034      1007 B(I)=B(I)-A(IJ)*B(J)                 00008460
ISN 0035      J=J-1                                    00008470
ISN 0036      GO TO 7                                   00008480
ISN 0037      8 IF (I-1) 9,9,6                          00008490
ISN 0038      9 IERR=0                                  00008500
ISN 0039      10 RETURN                                 00008510
ISN 0040      END                                       00008520
```

LEVEL 2.3.0 (JUNE 78)

SMLE

OS/360 FORTRAN H EXTENDED

DATE 84.250/13.02.37

PAGE 2

/ SMLE / SIZE OF PROGRAM 000388 HEXADECIMAL BYTES

NAME	TAG	TYPE	ADD.	NAME	TAG	TYPE	ADD.	NAME	TAG	TYPE	ADD.	NAME	TAG	TYPE	ADD.
A SF	XR	R*8	000000	B SF	XR	R*8	000000	I SF		I*4	00009C	J SF		I*4	0000A0
K SF		I*4	0000A4	L S		I*4	0000A8	N F		I*4	0000AC	II SF		I*4	0000B0
IJ SF		I*4	0000B4	JJ SF		I*4	0000B8	KJ SF		I*4	0000BC	N1 SF		I*4	0000C0
FAC SF		R*8	0000D8	FMUL SF		R*8	0000E0	IERR S		I*4	0000C4	IIJI SF		I*4	0000C8
IIKI SF		I*4	0000CC	JCOL SF		I*4	0000D0	SMLE		R*4	0000D4				

SOURCE STATEMENT LABELS

LABEL	ISN	ADDR	LABEL	ISN	ADDR	LABEL	ISN	ADDR	LABEL	ISN	ADDR
1	9	000156	2	11	00015E	2002	19	0001B2	1002	20	0001BC
3	21	0001C4	4	24	00020A	5	25	00023A	6	27	000254
7	32	0002B6	1007	34	0002CA	8	37	0002FC	9	38	000304
10	39	00030C									

COMPILER GENERATED LABELS

LABEL	ISN	ADDR	LABEL	ISN	ADDR	LABEL	ISN	ADDR	LABEL	ISN	ADDR
100001	2	000100	100002	7	000120	100003	14	00017C	100004	22	0001CC
100006	26	000244	100007	26	00024E						

*OPTIONS IN EFFECT*NAME(MAIN) OPTIMIZE(1) LINECOUNT(54) SIZE(MAX) AUTODBL(NONE)

*OPTIONS IN EFFECT*SOURCE EBCDIC NOLIST NODECK OBJECT MAP NOFORMAT GOSTMT NOXREF NOALC NOANSF NOTERM IBM FLAG(I)

STATISTICS SOURCE STATEMENTS = 39, PROGRAM SIZE = 904, SUBPROGRAM NAME = SMLE

STATISTICS NO DIAGNOSTICS GENERATED

***** END OF COMPILATION *****

266K BYTES OF CORE NOT USED

STATISTICS NO DIAGNOSTICS THIS STEP

CONFORMAL MAP PROJECTION COEFFICIENTS CALCULATION

LAT./LONG. AND WTS. OF 162 POINTS FOR MINIMUM SCALE ERROR

1.	81.5	-85.0	13.30280
2.	81.5	-55.0	13.30280
3.	81.5	-25.0	13.30280
4.	78.0	-120.0	16.63290
5.	78.0	-100.0	16.63290
6.	78.0	-80.0	16.63290
7.	78.0	-60.0	16.63290
8.	78.0	-40.0	16.63290
9.	78.0	-20.0	16.63290
10.	74.0	-120.0	22.05100
11.	74.0	-100.0	22.05100
12.	74.0	-80.0	22.05100
13.	74.0	-60.0	22.05100
14.	74.0	-40.0	22.05100
15.	74.0	-20.0	22.05100
16.	69.5	-162.5	29.26560
17.	69.5	-145.0	35.02070
18.	69.5	-127.5	26.26560
19.	69.5	-112.5	26.26560
20.	69.5	-97.5	26.26560
21.	69.5	-82.5	26.26560
22.	69.5	-67.5	26.26560
23.	69.5	-52.5	26.26560
24.	69.5	-37.5	26.26560
25.	69.5	-25.0	17.51040
26.	64.0	-172.0	19.28830
27.	64.5	-160.0	25.83070
28.	64.5	-148.0	25.83070
29.	64.5	-136.0	25.83070
30.	64.5	-124.0	25.83070
31.	64.5	-112.0	25.83070
32.	64.5	-100.0	25.83070
33.	64.5	-88.0	25.83070
34.	64.5	-76.0	25.83070
35.	64.5	-64.0	25.83070
36.	64.5	-52.0	25.83070
37.	64.0	-42.0	21.91860
38.	58.5	-175.0	36.57490
39.	58.5	-165.0	36.57490
40.	58.5	-155.0	36.57490
41.	59.5	-145.0	25.37690
42.	58.5	-135.0	36.57490
43.	58.5	-125.0	36.57490
44.	58.5	-115.0	36.57490
45.	58.5	-105.0	36.57490
46.	58.5	-95.0	36.57490
47.	58.5	-85.0	36.57490
48.	58.5	-75.0	36.57490
49.	58.5	-65.0	36.57490
50.	58.5	-55.0	36.57490
51.	61.0	-45.0	9.69620
52.	52.5	174.0	24.35050
53.	52.5	-178.0	24.35050
54.	52.5	-170.0	24.53050
55.	52.5	-163.0	18.26280
56.	52.5	-132.5	15.21900
57.	52.5	-127.5	15.21900

58.	52.5	-122.5	15.21900
59.	52.5	-117.5	15.21900
60.	52.5	-112.5	15.21900
61.	52.5	-107.5	15.21900
62.	52.5	-102.5	15.21900
63.	52.5	-97.5	15.21900
64.	52.5	-92.5	15.21900
65.	52.5	-87.5	15.21900
66.	52.5	-82.5	15.21900
67.	52.5	-77.5	15.21900
68.	52.5	-72.5	15.21900
69.	52.5	-67.5	15.21900
70.	52.5	-62.5	15.21900
71.	52.5	-57.5	15.21900
72.	52.5	-52.5	15.21900
73.	47.5	-127.5	16.88980
74.	47.5	-122.5	16.88980
75.	47.5	-117.5	16.88980
76.	47.5	-112.5	16.88980
77.	47.5	-107.5	16.88980
78.	47.5	-102.5	16.88980
79.	47.5	-97.5	16.88980
80.	47.5	-92.5	16.88980
81.	47.5	-87.5	16.88980
82.	47.5	-82.5	16.88980
83.	47.5	-77.5	16.88980
84.	47.5	-72.5	16.88980
85.	47.5	-67.5	16.88980
86.	47.5	-62.5	16.88980
87.	47.5	-57.5	16.88980
88.	47.5	-52.5	16.88980
89.	42.5	-127.5	18.43190
90.	42.5	-122.5	18.43190
91.	42.5	-117.5	18.43190
92.	42.5	-112.5	18.43190
93.	42.5	-107.5	18.43190
94.	42.5	-102.5	18.43190
95.	42.5	-97.5	18.43190
96.	42.5	-92.5	18.43190
97.	42.5	-87.5	18.43190
98.	42.5	-82.5	18.43190
99.	42.5	-77.5	18.43190
100.	42.5	-72.5	18.43190
101.	42.5	-67.5	18.43190
102.	43.5	-62.5	10.88060
103.	38.5	-127.5	11.73910
104.	37.5	-122.5	19.83380
105.	37.5	-117.5	19.83380
106.	37.5	-112.5	19.83380
107.	37.5	-107.5	19.83380
108.	37.5	-102.5	19.83380
109.	37.5	-97.5	19.83380
110.	37.5	-92.5	19.83380
111.	37.5	-87.5	19.83380
112.	37.5	-82.5	19.83380
113.	37.5	-77.5	19.83380
114.	37.5	-73.5	11.90030
115.	33.5	-122.5	12.50830
116.	32.5	-117.5	21.08480
117.	32.5	-112.5	21.08480
118.	32.5	-107.5	21.08480

119.	32.5	-102.5	21.08480
120.	32.5	-97.5	21.08480
121.	32.5	-92.5	21.08480
122.	32.5	-87.5	21.08480
123.	32.5	-82.5	21.08480
124.	32.5	-77.5	21.08480
125.	28.0	-117.5	17.65900
126.	26.0	-112.5	35.95180
127.	27.5	-107.5	22.17530
128.	27.5	-102.5	22.17530
129.	27.5	-97.5	22.17530
130.	27.5	-92.5	22.17530
131.	27.5	-87.5	22.17530
132.	27.5	-82.5	22.17530
133.	27.5	-77.5	22.17530
134.	26.0	-72.5	8.98790
135.	22.5	-107.5	23.09700
136.	22.5	-102.5	23.09700
137.	22.5	-97.5	23.09700
138.	22.5	-92.5	23.09700
139.	22.5	-87.5	23.09700
140.	22.5	-82.5	23.09700
141.	22.5	-77.5	23.09700
142.	22.5	-72.5	23.09700
143.	19.0	-107.5	9.45520
144.	18.0	-102.5	19.02110
145.	17.5	-97.5	23.84290
146.	17.5	-92.5	23.84290
147.	17.5	-87.5	23.84290
148.	17.5	-82.5	23.84290
149.	17.5	-77.5	23.84290
150.	17.5	-72.5	23.84290
151.	17.5	-67.5	23.84290
152.	17.5	-62.5	23.84290
153.	13.5	-92.5	14.58550
154.	12.5	-87.5	24.40740
155.	12.5	-82.5	24.40740
156.	12.5	-77.5	24.40740
157.	12.5	-72.5	24.40740
158.	12.5	-67.5	24.40740
159.	12.5	-62.5	24.40740
160.	9.0	-87.5	9.87690
161.	8.0	-82.5	19.80540
162.	8.0	-77.5	19.80540

INIT. EST. LAT./LONG. OF TRANSF. POLE FOR OBLIQUE CONF. CONIC, TRANS. OR OBLIQUE MERCATOR (IF CONE CONST. IS 0), OBLIQUE STEREOG.
(IF CONE CONSTANT IS 1): 0.0 -10.00000 CONE CONST.: 0.0 CODE=1 CODE1=1

IF CODE IS NOT 0, CONE CONST. IS FIXED. IF CODE1 IS 1, LAT. OF POLE IS FIXED. IF CODE1 IS 2, LAT/LONG OF POLE IS FIXED.

SUCCESSIVE ADJUSTMENTS TO COEFFICIENTS A(1)-A(1), AND LONG OF POLE:

-0.0399997	0.0860226
-0.0011548	0.0247746
-0.0000432	0.0058424
0.0000087	0.0013885
0.0000031	0.0003315

*** N03 ***

*** N03 ***

0.0000008 0.0000793
 0.0000002 0.0000190
 0.0000000 0.0000045

COEFFICIENTS A(1) THROUGH A(1), FOLLOWED BY LAT/LONG OF POLE, AND CONE CONST.

0.958815134

-0.00000 -3.21261 0.0

COEFFICIENTS ARE BASED ON USING 50.00000 AS CONFORMAL LAT. AND -100.00000 AS LONG. OF ORIGIN OF BASE PROJECTION.

SQUARE OF ECCENTRICITY OF REFERENCE ELLIPSOID= 0.0

RMS ERROR IN SCALE FOR 162 POINTS SELECTED= 0.0486789

LONG.	LAT.	K
-85.00000	81.50000	0.95903
-55.00000	81.50000	0.96285
-25.00000	81.50000	0.96798
-120.00000	78.00000	0.96305
-100.00000	78.00000	0.95910
-80.00000	78.00000	0.95990
-60.00000	78.00000	0.96509
-40.00000	78.00000	0.97239
-20.00000	78.00000	0.97839
-120.00000	74.00000	0.96630
-100.00000	74.00000	0.95932
-80.00000	74.00000	0.96072
-60.00000	74.00000	0.96993
-40.00000	74.00000	0.98307
-20.00000	74.00000	0.99405
-162.50000	69.50000	1.01481
-145.00000	69.50000	0.99731
-127.50000	69.50000	0.97804
-112.50000	69.50000	0.96530
-97.50000	69.50000	0.95914
-82.50000	69.50000	0.96085
-67.50000	69.50000	0.97008
-52.50000	69.50000	0.98485
-37.50000	69.50000	1.00166
-25.00000	69.50000	1.01392
-172.00000	64.00000	1.06201
-160.00000	64.50000	1.04401
-148.00000	64.50000	1.02427
-136.00000	64.50000	1.00265
-124.00000	64.50000	0.98298
-112.00000	64.50000	0.96817
-100.00000	64.50000	0.96006
-88.00000	64.50000	0.95955
-76.00000	64.50000	0.96669
-64.00000	64.50000	0.98071
-52.00000	64.50000	0.99988
-42.00000	64.00000	1.02022
-175.00000	58.50000	1.12024
-165.00000	58.50000	1.10445
-155.00000	58.50000	1.08011
-145.00000	59.50000	1.04555

-135.00000	58.50000	1.02281
-125.00000	58.50000	0.99734
-115.00000	58.50000	0.97737
-105.00000	58.50000	0.96432
-95.00000	58.50000	0.95894
-85.00000	58.50000	0.96150
-75.00000	58.50000	0.97186
-65.00000	58.50000	0.98948
-55.00000	58.50000	1.01320
-45.00000	61.00000	1.02835
174.00000	52.50000	1.20772
-178.00000	52.50000	1.20563
-170.00000	52.50000	1.19039
-163.00000	52.50000	1.16820
-132.50000	52.50000	1.03912
-127.50000	52.50000	1.02071
-122.50000	52.50000	1.00439
-117.50000	52.50000	0.99036
-112.50000	52.50000	0.97881
-107.50000	52.50000	0.96982
-102.50000	52.50000	0.96348
-97.50000	52.50000	0.95981
-92.50000	52.50000	0.95884
-87.50000	52.50000	0.96058
-82.50000	52.50000	0.96501
-77.50000	52.50000	0.97212
-72.50000	52.50000	0.98184
-67.50000	52.50000	0.99412
-62.50000	52.50000	1.00881
-57.50000	52.50000	1.02576
-52.50000	52.50000	1.04470
-127.50000	47.50000	1.03684
-122.50000	47.50000	1.01590
-117.50000	47.50000	0.99813
-112.50000	47.50000	0.98362
-107.50000	47.50000	0.97243
-102.50000	47.50000	0.96457
-97.50000	47.50000	0.96004
-92.50000	47.50000	0.95885
-87.50000	47.50000	0.96099
-82.50000	47.50000	0.96647
-77.50000	47.50000	0.97528
-72.50000	47.50000	0.98742
-67.50000	47.50000	1.00286
-62.50000	47.50000	1.02155
-57.50000	47.50000	1.04337
-52.50000	47.50000	1.06811
-127.50000	42.50000	1.05403
-122.50000	42.50000	1.02801
-117.50000	42.50000	1.00620
-112.50000	42.50000	0.98858
-107.50000	42.50000	0.97509
-102.50000	42.50000	0.96568
-97.50000	42.50000	0.96027
-92.50000	42.50000	0.95886
-87.50000	42.50000	0.96141
-82.50000	42.50000	0.96795
-77.50000	42.50000	0.97852
-72.50000	42.50000	0.99318
-67.50000	42.50000	1.01198
-62.50000	43.50000	1.03227

-127.50000	38.50000	1.06824
-122.50000	37.50000	1.04036
-117.50000	37.50000	1.01434
-112.50000	37.50000	0.99353
-107.50000	37.50000	0.97774
-102.50000	37.50000	0.96677
-97.50000	37.50000	0.96051
-92.50000	37.50000	0.95886
-87.50000	37.50000	0.96182
-82.50000	37.50000	0.96941
-77.50000	37.50000	0.98174
-73.50000	37.50000	0.99511
-122.50000	33.50000	1.05017
-117.50000	32.50000	1.02230
-112.50000	32.50000	0.99834
-107.50000	32.50000	0.98028
-102.50000	32.50000	0.96782
-97.50000	32.50000	0.96073
-92.50000	32.50000	0.95887
-87.50000	32.50000	0.96221
-82.50000	32.50000	0.97082
-77.50000	32.50000	0.98485
-117.50000	28.00000	1.02908
-112.50000	26.00000	1.00408
-107.50000	27.50000	0.98265
-102.50000	27.50000	0.96879
-97.50000	27.50000	0.96093
-92.50000	27.50000	0.95887
-87.50000	27.50000	0.96257
-82.50000	27.50000	0.97212
-77.50000	27.50000	0.98774
-72.50000	26.00000	1.01127
-107.50000	22.50000	0.98475
-102.50000	22.50000	0.96965
-97.50000	22.50000	0.96111
-92.50000	22.50000	0.95888
-87.50000	22.50000	0.96290
-82.50000	22.50000	0.97327
-77.50000	22.50000	0.99031
-72.50000	22.50000	1.01450
-107.50000	19.00000	0.98603
-102.50000	18.00000	0.97031
-97.50000	17.50000	0.96126
-92.50000	17.50000	0.95888
-87.50000	17.50000	0.96316
-82.50000	17.50000	0.97425
-77.50000	17.50000	0.99249
-72.50000	17.50000	1.01851
-67.50000	17.50000	1.05321
-62.50000	17.50000	1.09786
-92.50000	13.50000	0.95889
-87.50000	12.50000	0.96337
-82.50000	12.50000	0.97500
-77.50000	12.50000	0.99419
-72.50000	12.50000	1.02165
-67.50000	12.50000	1.05846
-62.50000	12.50000	1.10613
-87.50000	9.00000	0.96348
-82.50000	8.00000	0.97548
-77.50000	8.00000	0.99527

INIT. EST. LAT./LONG. OF TRANSF. POLE FOR OBLIQUE CONF. CONIC, TRANS. OR OBLIQUE MERCATOR (IF CONE CONST. IS 0), OBLIQUE STEREOG.
(IF CONE CONSTANT IS 1): 5.00000 0.0 CONE CONST.: 0.0 CODE=1 CODE1=0

IF CODE IS NOT 0, CONE CONST. IS FIXED. IF CODE1 IS 1, LAT. OF POLE IS FIXED. IF CODE1 IS 2, LAT/LONG OF POLE IS FIXED.

SUCCESSIVE ADJUSTMENTS TO COEFFICIENTS A(1)-A(1), AND LAT/LONG OF POLE:

-0.0321133	0.0983541	0.1230220
-0.0007752	0.0110274	0.0427917
-0.0001200	0.0055434	0.0133295
-0.0000018	0.0014124	0.0038121
0.0000018	0.0004382	0.0011047
0.0000007	0.0001225	0.0003165
0.0000002	0.0000357	0.0000910
0.0000001	0.0000102	0.0000261
0.0000000	0.0000029	0.0000075
0.0000000	0.0000008	0.0000022

COEFFICIENTS A(1) THROUGH A(1), FOLLOWED BY LAT/LONG OF POLE, AND CONE CONST.

0.966992474

11.70061 10.57127 0.0

COEFFICIENTS ARE BASED ON USING 50.00000 AS CONFORMAL LAT. AND -100.00000 AS LONG. OF ORIGIN OF BASE PROJECTION.

SQUARE OF ECCENTRICITY OF REFERENCE ELLIPSOID= 0.0

RMS ERROR IN SCALE FOR 162 POINTS SELECTED= 0.0325775

LONG.	LAT.	K
-85.00000	81.50000	0.98426
-55.00000	81.50000	1.00155
-25.00000	81.50000	1.02004
-120.00000	78.00000	0.96910
-100.00000	78.00000	0.97486
-80.00000	78.00000	0.98619
-60.00000	78.00000	1.00316
-40.00000	78.00000	1.02350
-20.00000	78.00000	1.04250
-120.00000	74.00000	0.96717
-100.00000	74.00000	0.97187
-80.00000	74.00000	0.98537
-60.00000	74.00000	1.00874
-40.00000	74.00000	1.03925
-20.00000	74.00000	1.06957
-162.50000	69.50000	0.97813
-145.00000	69.50000	0.97430
-127.50000	69.50000	0.96905
-112.50000	69.50000	0.96700
-97.50000	69.50000	0.97039

-82.50000	69.50000	0.98155
-67.50000	69.50000	1.00167
-52.50000	69.50000	1.03035
-37.50000	69.50000	1.06504
-25.00000	69.50000	1.09480
-172.00000	64.00000	0.99780
-160.00000	64.50000	0.99432
-148.00000	64.50000	0.98891
-136.00000	64.50000	0.98107
-124.00000	64.50000	0.97320
-112.00000	64.50000	0.96793
-100.00000	64.50000	0.96758
-88.00000	64.50000	0.97406
-76.00000	64.50000	0.98867
-64.00000	64.50000	1.01209
-52.00000	64.50000	1.04414
-42.00000	64.00000	1.07870
-175.00000	58.50000	1.02680
-165.00000	58.50000	1.02715
-155.00000	58.50000	1.02161
-145.00000	59.50000	1.00660
-135.00000	58.50000	0.99847
-125.00000	58.50000	0.98542
-115.00000	58.50000	0.97460
-105.00000	58.50000	0.96811
-95.00000	58.50000	0.96760
-85.00000	58.50000	0.97442
-75.00000	58.50000	0.98958
-65.00000	58.50000	1.01382
-55.00000	58.50000	1.04752
-45.00000	61.00000	1.08026
174.00000	52.50000	1.06044
-178.00000	52.50000	1.07026
-170.00000	52.50000	1.07403
-163.00000	52.50000	1.07190
-132.50000	52.50000	1.01909
-127.50000	52.50000	1.00809
-122.50000	52.50000	0.99770
-117.50000	52.50000	0.98834
-112.50000	52.50000	0.98033
-107.50000	52.50000	0.97399
-102.50000	52.50000	0.96956
-97.50000	52.50000	0.96727
-92.50000	52.50000	0.96732
-87.50000	52.50000	0.96989
-82.50000	52.50000	0.97513
-77.50000	52.50000	0.98322
-72.50000	52.50000	0.99431
-67.50000	52.50000	1.00855
-62.50000	52.50000	1.02609
-57.50000	52.50000	1.04705
-52.50000	52.50000	1.07156
-127.50000	47.50000	1.02931
-122.50000	47.50000	1.01444
-117.50000	47.50000	1.00099
-112.50000	47.50000	0.98937
-107.50000	47.50000	0.97990
-102.50000	47.50000	0.97287
-97.50000	47.50000	0.96850
-92.50000	47.50000	0.96699
-87.50000	47.50000	0.96855

-82.50000	47.50000	0.97335
-77.50000	47.50000	0.98158
-72.50000	47.50000	0.99347
-67.50000	47.50000	1.00923
-62.50000	47.50000	1.02910
-57.50000	47.50000	1.05337
-52.50000	47.50000	1.08229
-127.50000	42.50000	1.05513
-122.50000	42.50000	1.03479
-117.50000	42.50000	1.01647
-112.50000	42.50000	1.00059
-107.50000	42.50000	0.98750
-102.50000	42.50000	0.97745
-97.50000	42.50000	0.97067
-92.50000	42.50000	0.96733
-87.50000	42.50000	0.96761
-82.50000	42.50000	0.97170
-77.50000	42.50000	0.97982
-72.50000	42.50000	0.99223
-67.50000	42.50000	1.00922
-62.50000	43.50000	1.03082
-127.50000	38.50000	1.07914
-122.50000	37.50000	1.05868
-117.50000	37.50000	1.03461
-112.50000	37.50000	1.01384
-107.50000	37.50000	0.99665
-102.50000	37.50000	0.98323
-97.50000	37.50000	0.97374
-92.50000	37.50000	0.96831
-87.50000	37.50000	0.96709
-82.50000	37.50000	0.97025
-77.50000	37.50000	0.97800
-73.50000	37.50000	0.98770
-122.50000	33.50000	1.08022
-117.50000	32.50000	1.05523
-112.50000	32.50000	1.02892
-107.50000	32.50000	1.00718
-102.50000	32.50000	0.99008
-97.50000	32.50000	0.97765
-92.50000	32.50000	0.96993
-87.50000	32.50000	0.96702
-82.50000	32.50000	0.96902
-77.50000	32.50000	0.97616
-117.50000	28.00000	1.07563
-112.50000	26.00000	1.05081
-107.50000	27.50000	1.01889
-102.50000	27.50000	0.99784
-97.50000	27.50000	0.98229
-92.50000	27.50000	0.97214
-87.50000	27.50000	0.96738
-82.50000	27.50000	0.96807
-77.50000	27.50000	0.97438
-72.50000	26.00000	0.98593
-107.50000	22.50000	1.03151
-102.50000	22.50000	1.00635
-97.50000	22.50000	0.98756
-92.50000	22.50000	0.97489
-87.50000	22.50000	0.96818
-82.50000	22.50000	0.96740
-77.50000	22.50000	0.97269
-72.50000	22.50000	0.98431

-107.50000	19.00000	1.04071
-102.50000	18.00000	1.01445
-97.50000	17.50000	0.99332
-92.50000	17.50000	0.97809
-87.50000	17.50000	0.96938
-82.50000	17.50000	0.96705
-77.50000	17.50000	0.97115
-72.50000	17.50000	0.98191
-67.50000	17.50000	0.99979
-62.50000	17.50000	1.02549
-92.50000	13.50000	0.98093
-87.50000	12.50000	0.97096
-82.50000	12.50000	0.96702
-77.50000	12.50000	0.96980
-72.50000	12.50000	0.97949
-67.50000	12.50000	0.99649
-62.50000	12.50000	1.02149
-87.50000	9.00000	0.97227
-82.50000	8.00000	0.96726
-77.50000	8.00000	0.96879

INIT. EST. LAT./LONG. OF TRANSF. POLE FOR OBLIQUE CONF. CONIC, TRANS. OR OBLIQUE MERCATOR (IF CONE CONST. IS 0), OBLIQUE STEREOG.
(IF CONE CONSTANT IS 1): -10.00000-170.00000 CONE CONST.: 0.0 CODE=1 CODE1=2

IF CODE IS NOT 0, CONE CONST. IS FIXED. IF CODE1 IS 1, LAT. OF POLE IS FIXED. IF CODE1 IS 2, LAT/LONG OF POLE IS FIXED.

SUCCESSIVE ADJUSTMENTS TO COEFFICIENTS A(1)-A(10), AND B(2)-B(10):

-0.0315980	0.0034540	0.0461590	0.0031199	-0.0250259	0.0588764	0.0799607	-0.0874274
-0.0524711	0.0563506	0.0225055	-0.0333458	0.0165190	0.0714492	0.0084677	-0.0337052
0.0075188	0.0464498	-0.0078929					
-0.0003385	0.0005095	-0.0014540	-0.0015949	0.0005632	-0.0004888	0.0035803	0.0019992
-0.0036736	0.0002815	-0.0010862	0.0004935	-0.0012409	-0.0019951	0.0047363	0.0031883
-0.0057198	0.0005889	0.0042804					
0.0000015	-0.0000285	0.0000312	0.0000789	-0.0001191	-0.0000275	0.0001612	-0.0000956
-0.0001341	0.0000222	0.0000350	-0.0000626	0.0000565	0.0001300	-0.0001417	-0.0001519
0.0002952	0.0000843	-0.0002620					
-0.0000004	0.0000019	-0.0000015	-0.0000052	0.0000064	-0.0000003	-0.0000020	0.0000043
-0.0000023	0.0000077	-0.0000020	0.0000042	-0.0000053	-0.0000058	0.0000105	0.0000049
-0.0000195	-0.0000006	0.0000132					
0.0000000	-0.0000001	0.0000001	0.0000003	-0.0000006	0.0000001	0.0000008	-0.0000009
-0.0000006	0.0000004	0.0000001	-0.0000003	0.0000003	0.0000006	-0.0000005	-0.0000009
0.0000012	0.0000005	-0.0000013					

COEFFICIENTS A(1) THROUGH A(10), FOLLOWED BY COEFF. B(2) THROUGH B(10), LAT/LONG OF POLE, AND CONE CONST.

0.968064600 0.003936846 0.044734844 0.001598984 -0.024576031 0.058359795 0.083700981 -0.085520392 -0.056281629
0.056662369

0.021452466 -0.032910990 0.015329612 0.069578922 0.013072309 -0.030664864 0.002075969 0.047122960 -0.003862700

-10.00000 -170.00000 0.0

COEFFICIENTS ARE BASED ON USING 50.00000 AS CONFORMAL LAT. AND -100.00000 AS LONG. OF ORIGIN OF BASE PROJECTION.

SQUARE OF ECCENTRICITY OF REFERENCE ELLIPSOID= 0.0

RMS ERROR IN SCALE FOR 162 POINTS SELECTED= 0.0170209

LONG.	LAT.	K
-85.00000	81.50000	1.02924
-55.00000	81.50000	1.02719
-25.00000	81.50000	1.03053
-120.00000	78.00000	1.02333
-100.00000	78.00000	1.01425
-80.00000	78.00000	1.01053
-60.00000	78.00000	1.01034
-40.00000	78.00000	1.00933
-20.00000	78.00000	1.00212
-120.00000	74.00000	1.00694
-100.00000	74.00000	0.99857
-80.00000	74.00000	0.99685
-60.00000	74.00000	1.00062
-40.00000	74.00000	1.00320
-20.00000	74.00000	0.98345
-162.50000	69.50000	1.02264
-145.00000	69.50000	1.01143
-127.50000	69.50000	0.99987
-112.50000	69.50000	0.99101
-97.50000	69.50000	0.98599
-82.50000	69.50000	0.98635
-67.50000	69.50000	0.99223
-52.50000	69.50000	1.00313
-37.50000	69.50000	1.01581
-25.00000	69.50000	1.01351
-172.00000	64.00000	1.00251
-160.00000	64.50000	1.00787
-148.00000	64.50000	1.00764
-136.00000	64.50000	1.00189
-124.00000	64.50000	0.99242
-112.00000	64.50000	0.98341
-100.00000	64.50000	0.97805
-88.00000	64.50000	0.97775
-76.00000	64.50000	0.98265
-64.00000	64.50000	0.99247
-52.00000	64.50000	1.00854
-42.00000	64.00000	1.03243
-175.00000	58.50000	0.98703
-165.00000	58.50000	1.00082
-155.00000	58.50000	1.01401
-145.00000	59.50000	1.01513
-135.00000	58.50000	1.01102
-125.00000	58.50000	0.99719
-115.00000	58.50000	0.98403
-105.00000	58.50000	0.97522
-95.00000	58.50000	0.97200
-85.00000	58.50000	0.97447
-75.00000	58.50000	0.98192
-65.00000	58.50000	0.99275
-55.00000	58.50000	1.00566
-45.00000	61.00000	1.02775
174.00000	52.50000	1.00100
-178.00000	52.50000	1.00548
-170.00000	52.50000	0.99091
-163.00000	52.50000	1.00848
-132.50000	52.50000	1.03035
-127.50000	52.50000	1.01611

-122.50000	52.50000	1.00305
-117.50000	52.50000	0.99201
-112.50000	52.50000	0.98327
-107.50000	52.50000	0.97683
-102.50000	52.50000	0.97263
-97.50000	52.50000	0.97060
-92.50000	52.50000	0.97071
-87.50000	52.50000	0.97290
-82.50000	52.50000	0.97708
-77.50000	52.50000	0.98304
-72.50000	52.50000	0.99039
-67.50000	52.50000	0.99841
-62.50000	52.50000	1.00597
-57.50000	52.50000	1.01145
-52.50000	52.50000	1.01296
-127.50000	47.50000	1.03250
-122.50000	47.50000	1.01261
-117.50000	47.50000	0.99745
-112.50000	47.50000	0.98626
-107.50000	47.50000	0.97838
-102.50000	47.50000	0.97340
-97.50000	47.50000	0.97113
-92.50000	47.50000	0.97156
-87.50000	47.50000	0.97472
-82.50000	47.50000	0.98068
-77.50000	47.50000	0.98940
-72.50000	47.50000	1.00064
-67.50000	47.50000	1.01366
-62.50000	47.50000	1.02680
-57.50000	47.50000	1.03683
-52.50000	47.50000	1.03801
-127.50000	42.50000	1.04521
-122.50000	42.50000	1.01932
-117.50000	42.50000	1.00182
-112.50000	42.50000	0.98961
-107.50000	42.50000	0.98108
-102.50000	42.50000	0.97560
-97.50000	42.50000	0.97309
-92.50000	42.50000	0.97369
-87.50000	42.50000	0.97766
-82.50000	42.50000	0.98536
-77.50000	42.50000	0.99720
-72.50000	42.50000	1.01359
-67.50000	42.50000	1.03462
-62.50000	43.50000	1.05230
-127.50000	38.50000	1.04586
-122.50000	37.50000	1.01948
-117.50000	37.50000	1.00479
-112.50000	37.50000	0.99402
-107.50000	37.50000	0.98560
-102.50000	37.50000	0.97964
-97.50000	37.50000	0.97658
-92.50000	37.50000	0.97687
-87.50000	37.50000	0.98104
-82.50000	37.50000	0.98976
-77.50000	37.50000	1.00404
-73.50000	37.50000	1.02040
-122.50000	33.50000	1.01568
-117.50000	32.50000	1.01039
-112.50000	32.50000	1.00239
-107.50000	32.50000	0.99347

-102.50000	32.50000	0.98620
-97.50000	32.50000	0.98190
-92.50000	32.50000	0.98118
-87.50000	32.50000	0.98457
-82.50000	32.50000	0.99296
-77.50000	32.50000	1.00779
-117.50000	28.00000	1.02632
-112.50000	26.00000	1.02598
-107.50000	27.50000	1.00582
-102.50000	27.50000	0.99556
-97.50000	27.50000	0.98938
-92.50000	27.50000	0.98701
-87.50000	27.50000	0.98855
-82.50000	27.50000	0.99491
-77.50000	27.50000	1.00767
-72.50000	26.00000	1.02701
-107.50000	22.50000	1.02081
-102.50000	22.50000	1.00680
-97.50000	22.50000	0.99940
-92.50000	22.50000	0.99531
-87.50000	22.50000	0.99386
-82.50000	22.50000	0.99645
-77.50000	22.50000	1.00483
-72.50000	22.50000	1.02016
-107.50000	19.00000	1.02893
-102.50000	18.00000	1.01692
-97.50000	17.50000	1.01355
-92.50000	17.50000	1.00825
-87.50000	17.50000	1.00187
-82.50000	17.50000	0.99850
-77.50000	17.50000	1.00170
-72.50000	17.50000	1.01060
-67.50000	17.50000	1.01862
-62.50000	17.50000	1.01822
-92.50000	13.50000	1.02509
-87.50000	12.50000	1.01395
-82.50000	12.50000	0.99997
-77.50000	12.50000	0.99880
-72.50000	12.50000	1.00937
-67.50000	12.50000	1.01330
-62.50000	12.50000	0.97311
-87.50000	9.00000	1.02522
-82.50000	8.00000	0.99509
-77.50000	8.00000	0.99018

NORMAL END OF RUN.