```
      PROGRAM TRANS
C PROGRAM FOR THE TRANSFORMATION OF FILES OF COORDINATES
C TO OR FROM GEOGRAPHICAL COORDINATES FROM/TO A PLANE SYSTEM.
C PROGRAMMED BY C.C.TSCHERNING, 1990.03.01.
C LATEST UPDATE: 1997.07.16 BY CCT.
      IMPLICIT REAL *8(A-H,O-Z)
C LU1    LOGICAL UNIT TO READ COORDINATES FROM.
C NPTS   MAXIMAL NUMBER OF DATA POINTS.
C
      PARAMETER (LU1=10,NPTS=200000)
      COMMON /ELL/ ELLIP(15,2),XDBLON(NPTS),YDBLAT(NPTS),TDAT(12),
     *TEMP(NPTS),SA(120),ELLIPTX(16),PROJTX(9)
      LOGICAL TEST,GRID,IMGRID,FORWAR,REWERS,LSTORE,LSHIFT,LKEYB,
     *LSTOP,NOSTNO
      COMMON /EUCL/X,Y,Z,XY,XY2,DISTO,DIST2
      COMMON /CONST/PI,PI4,RADDEG,DEGRAD,D0,D1,D2
      COMMON /ITRANC/DX,DY,DZ,EPS3,EPS2,EPS1,S1,AX,E2
      DIMENSION ISTNX(NPTS)
      CHARACTER*60 PROJTX
      CHARACTER*25 ELLIPTX
      CHARACTER*75 FIGTXT
      CHARACTER*72 INAME,PNAME
C
       LSTOP=.FALSE.
       TEST=.FALSE.
       PI4=DATAN(1.0D0)
       PI=4.0D0*PI4
       DEGRAD=PI/180.0D0
       RADDEG=D1/DEGRAD
C
       WRITE(*,140)
 140   FORMAT(' MAP TRANSFORMATION PROGRAM, VERS. AUG. 23, 1994',/
     *' COPYRIGHT GEOPHYSICAL INSTITUTE, UNIV. COPENHAGEN, 1991, 94'
     *//' THE PROGRAM TRANSFORM FROM GEOGRAPHICAL COORDINATES TO ',/
     *' PLANE OR 3-D CARTESIAN COORDINATES OR REVERSE',/
     *' INPUT IDENTIFYING TEXT (MAX. 72 CHAR). ')
       READ(*,72)FIGTXT
   72  FORMAT(A72)
       WRITE(*,72)FIGTXT
       WRITE(*,*)' FORWARD TRANSFORMATION (GEO TO XY(Z)) ? (T/F) '
       READ(*,*)FORWAR
       REWERS=.NOT.FORWAR
C
       WRITE(*,111)
  111  FORMAT(' SELECT SPHEROID:'/
     *' 1: SPHERE, 2: CLARKE 1866,'/
     *' 3: HAYFORD 1909 (INTERNATIONAL), 4: GRS 1980',/
     *' 5: CLARK1880, 6: BESSEL 1841, 7: KRASOVSKY 1940, 8: WGS 1972'/
     *' 9: AUSTRALIAN 1965, 10: AIRY1849, 11: EVEREST 1830'/
     *' 12: HOUGH 1956,: 13: FISHER 1960, 14&15: SPHERE, 16: INPUT '/)
       READ(*,*)ISPHER
       IF (ISPHER.LT.17) THEN
       WRITE(*,*) ' ELLIPSOID : ',ELLIPTX(ISPHER),ISPHER
       SEMIA=ELLIP(ISPHER,1)
       SEMIB=ELLIP(ISPHER,2)
       ELSE
       WRITE(*,*)' INPUT SEMI MAJOR AND MINOR AXIS IN M '
       READ(*,*)SEMIA,SEMIB
       END IF
       FLA1=(SEMIA-SEMIB)/SEMIA
       IF (FLA1.GT.D0)FLA=D1/FLA1
       WRITE(*,100)FLA
  100  FORMAT(' 1/FLATTENING =',F12.7)
C
       WRITE(*,112)
  112  FORMAT(' SELECT PROJECTION:'/
     * ' 1: U T M, 2: TRANSVERSE MERCATOR, 3: MERCATOR'/
     * ' 4: LAMBERT CONFORMAL CONIC (2 STD. PARAL.)'/
     * ' 5: POLAR STEREOGRAPHIC (AZIMUTHAL-),'/
```



```
     * ' 6: SYSTEM 34 JYLLAND, 7: SYSTEM 34 SJAELLAND,'/
     * ' 8: SYSTEM 45 BORNHOLM, 9: CARTESIAN (X,Y,Z) '/)
       READ(*,*)IPROJ
       WRITE(*,*)'                    ;',PROJTX(IPROJ)
       IF (IPROJ.GT.5.AND.IPROJ.LT.9.AND.ISPHER.NE.3) WRITE(*,*)
     * ' WARNING *** HAYFORD ELLIPSOID MUST BE USED WITH S34 '
C
C DEFINE PROJECTION SCALE FACTOR.
       IF (IPROJ.EQ.1.OR.IPROJ.GT.5) CESFA=0.9996D0
       IF (IPROJ.EQ.3) CESFA=D1
       IF (IPROJ.EQ.1.OR.IPROJ.EQ.3.OR.IPROJ.GT.5) GO TO 610
       WRITE(*,*)' INPUT CENTRAL SCALE FACTOR '
       READ(*,*)CESFA
  610  DM=D1-CESFA
C
       IF (IPROJ.EQ.1.OR.IPROJ.GT.5) GO TO 770
       WRITE(*,*)' INPUT LONGITUDE OF CENTRAL MERIDIAN IN DEG. '
       READ(*,*)CMER
       CMERR=CMER*DEGRAD
C
       IF (IPROJ.GE.5) GOTO 774
       IF (IPROJ.EQ.2) THEN
       WRITE(6,*)' INPUT ABSCISSA CONSTANT IN M. '
       READ(5,*)XCONST
       END IF
       WRITE(*,*)' INPUT LATITUDE OF BASE PARALLEL IN DEG. '
       READ(*,*)BPAR
       IF (ABS(BPAR).GT.0.0D-10.AND.IPROJ.EQ.3)
     * WRITE(*,*)' Y=0 AT BASE PARALLEL FOR KMS ROUTINES '
       BPARR=BPAR*DEGRAD
       IF (IPROJ.NE.4) GO TO 774
       WRITE(*,*)' INPUT LATITUDE OF 1. & 2. STD. PARALLEL IN DEG. '
       READ(*,*)SPAR1,SPAR2
       WRITE(*,*)SPAR1,SPAR2
       SPARR1=SPAR1*DEGRAD
       SPARR2=SPAR2*DEGRAD
       GO TO 774
C
 770   IF (IPROJ.GE.5) GO TO 774
       WRITE(*,*)' INPUT UTM ZONE '
       READ(*,*)IZONE
C FOR S34 PROJECTIONS, WE TRANSFORM FROM GEOGRAHICAL TO UTM ZONE
C 32 AND 33, AND THEN TO S34.
 774   IF (IPROJ.EQ.6.OR.IPROJ.EQ.7) IZONE=32
       IF (IPROJ.GT.7) IZONE=33
C
       GO TO (792,792,793,794,796,792,792,792,780)IPROJ
 792   EASTPAR=D0
       IF (IPROJ.EQ.2) GO TO 797
       CMERR=(-183+IZONE*6)*DEGRAD
       EASTPAR=500000.0D0
 797   CALL UTMCON(SEMIA,1/FLA,0,DM,EASTPAR,CMERR)
       GO TO 796
C
 793   CALL MRCCON(SEMIA,1/FLA,BPARR,CMERR,0.0)
       GO TO 796
C
 794   CALL  LMBCON(SEMIA,1/FLA,SPARR1,SPARR2,1,CMERR,DM,0.0,BPAR)
       GO TO 796
 780   E2=FLA1*(2.0D0-FLA1)
       AX=SEMIA
       HP=0.0D0
       WRITE(*,*)' IS DATUM SHIFT NECESSARY ? (T/F)'
       READ(*,*)LSHIFT
       IF (LSHIFT) THEN
       WRITE(*,*)' INPUT DX,DY,DZ (M)'
       READ(*,*)DX,DY,DZ
       WRITE(*,*)' INPUT SCALE CHANGE -1.0'
       READ(*,*)S1
```

```
       S1=D1+S1
       WRITE(*,*)
     * ' INPUT ROTATIONS IN ARCSEC AROUND X,Y,Z AXES '
       READ(*,*)EPS3,EPS2,EPS1
       SECRAD=DEGRAD/3600.0D0
       EPS3=EPS3*SECRAD
       EPS2=EPS2*SECRAD
       EPS1=EPS1*SECRAD
       END IF
C
C FILE TO BE TRANSFORMED MUST HAVE RECORDS ON THE FORM:
C (1) NO. LATITUDE, LONGITUDE, DATA(1),..,DATA(NDUSE)
C (2) NO. Y, X, DATA(1),....,DATA(NDUSE), WITH Z=DATA(1),
C (3) NO. X, Y, Z, DATA(2), ....,DATA(NDUSE), WITH Z=DATA(1),
C (4) GRID FORM, WITH COORDINATES GIVEN IMPLICITLY THROUGH A
C     GRID LABEL. IN THIS CASE, NDE:= 0.
C
 796   WRITE(*,703)
 703   FORMAT('  AN INPUT RECORD MUST BE ON THE FORM: ',
     * /'  (1) NO. LATITUDE, LONGITUDE,DATA(1),..,DATA(N)'/
     * '  (2) NO. Y, X,DATA(1), DATA(2),..,DATA(N)',/,
     * '  (3) NO. X, Y, Z, DATA(2),...,DATA(N) (CARTESIAN)',/
     * '  (4) GRID FORM, WITH COORDINATES GIVEN IMPLICITLY ',/,
     * '  THROUGH A GRID LABEL.    IF TRANSFORMATION ',/
     *'  FROM (1) TO (3) DATA(1) MUST BE THE HEIGHT'//
     * '  INPUT NUMBER OF DATA ELEMENTS, AND ELEMENT NUMBER USED',/
     *'  (IF NO STATION NUMBER AVAILABLE IN FILE USE NEGATIVE ELEM',
     *'ENT NUMBER)',/
     *'  (ALL ELEMENTS WILL BE REPRODUCED, IF OUTPUT TO FILE) ')
       READ(*,*)NDE,NDUSE
       NOSTNO=NDUSE.LT.0
       NDUSE=ABS(NDUSE)
       IF (REWERS.AND.IPROJ.EQ.9.AND.NDE.LT.1) WRITE(6,*)
     *' WARNING : MISSING Z VALUE '
      WRITE(*,*)
     *' ANGLES IN 1: DD MM SS.S, 2: DD MM.M 3: DD.D, 4: CC.C ? '
       READ(*,*)IANG
       IF (FORWAR) THEN
       IANGR=3
       ELSE
       IANGR=IANG
       IANG=3
       END IF
C
       WRITE(*,*)' ARE DATA GRIDDED ? (T/F) '
       READ(*,*)GRID
       IMGRID=GRID.AND.NDE.EQ.0
C
       WRITE(*,*)' ARE INPUT DATA FROM THE KEY-BOARD (T/F) ?'
       READ(*,*)LKEYB
       IF (LKEYB) THEN
       IFILE=5
       ELSE
       IFILE=12
       WRITE(*,*)' INPUT NAME OF FILE WITH DATA TO BE TRANSFORMED '
       READ(*,72)INAME
       OPEN(12,FILE=INAME,STATUS='OLD')
       END IF
C
  798 WRITE(*,*)' TRANSFORMED DATA TO BE STORED ON FILE ? (T/F) '
      READ(*,*)LSTORE
      IF (.NOT.LSTORE) GO TO 790
      WRITE(*,*)' INPUT NAME OF FILE TO HOLD TRANSFORMED DATA '
       READ(*,72)PNAME
       OPEN(14,FILE=PNAME,STATUS='UNKNOWN')
C
C INPUT DATA TO BE TRANSFORMED FROM UNIT 12.
C
 790   TMIN=1.0E8
```



```
       TMAX=-TMIN
       I=0
       IF (.NOT.GRID) GO TO 998
       WRITE(*,141)
 141   FORMAT(' GRID LABEL IS MIN, MAX NORTHGOING COORD,'/
     * ' MIN MAX EASTGOING COORD, N-SPACING, E-SPACING')
       IF (LKEYB) WRITE(*,*)' INPUT GRID LABEL (DD.D OR M) '
       READ(IFILE,*)YDBL0,YDBL1,XDBL0,XDBL1,YSTEP,XSTEP
       NDEAST=(XDBL1-XDBL0)/XSTEP+1.001
       NDSOU =(YDBL1-YDBL0)/YSTEP+1.001
       I=NDEAST*NDSOU
       NPTI=I
       IF (I.GT.NPTS)
     * WRITE(*,*)' ** ERROR ** GRID TOO LARGE '
       IF (LKEYB.AND.NDE.GT.0) WRITE(*,*)' INPUT ',I,' GRID VALUES '
       IF (NDE.GT.0) READ(IFILE,*)(TEMP(K),K=1,I)
       K=0
       DO 907 N=1,NDSOU
```