UBO
CNRS
IFREMER
ORSTOM
BRGM

Groupement Scientifique

GEOLOGIE :

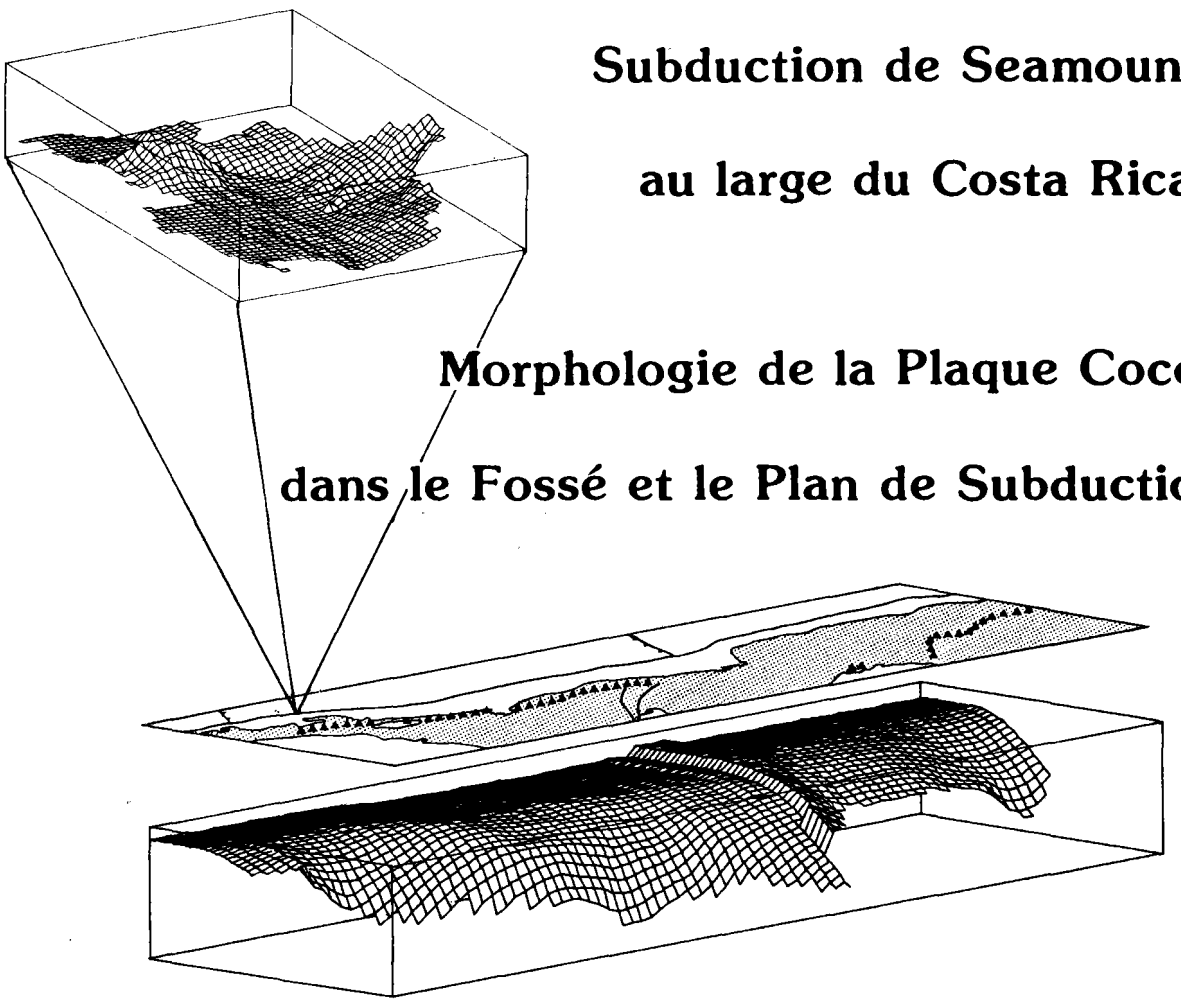
OCEANOLOGIE ET GEODYNAMIQUE

Jacqueline ROUMP

Le Fossé d'Amérique Centrale :

Subduction de Seamounts au large du Costa Rica ;

Morphologie de la Plaque Cocos dans le Fossé et le Plan de Subduction



THESE DE DOCTORAT DE 3ème CYCLE

BREST 1985

Volume 2

IFREMER

SI LES PROGRAMMES DE CETTE THESE VOUS INTERESSENT

Les programmes de cette thèse seront disponibles sur disquette, entièrement adaptés aux micro-ordinateurs compatibles IBM, aux imprimantes compatibles EPSON/IBM et au plotter HP 7475A, avec le code source et sous forme compilée, à partir d'Avril-Mai 1988.

Le prix de vente, non encore fixé, devrait avoisiné les 200 F. Si seul un programme vous intéresse, un prix individuel sera établi.

La demande est à adresser à :

Jacqueline ROUMP
191 rue Commandant Drogou
29200 BREST

IF YOU ARE INTERESTED BY THE COMPUTER PROGRAMS OF THIS THESIS

The programs of this thesis will be available on disk, completly adapted to IBM compatible micro-computers, to EPSON/IBM compatible printers and to HP 7475A plotter, with source code and compiled version, in April-May 1988.

The price should be close to 200 F. If if you are interested by only one of the programs, price will be lower.

Send the order to :

Jacqueline ROUMP
191 rue Commandant Drogou
29200 BREST
FRANCE

T 239

PROGRAMMES

SOMMAIRE

C - Copie d'écran d'un exemple d'exécution du programme `PLATES'
D - Exemple de sortie sur imprimante et sur disque

3 - CONVERSIONS D'ECHELLES FRACTIONNAIRES EN POUCES-PAR-DEGRE DE LONGITUDE ET INVERSEMENT

A - Listing du programme `SCALE'
B - Copie d'écran d'un premier exemple d'exécution du programme `SCALE'
C - Sortie sur traceur du résultat de cette première exécution
D - Copie d'écran d'un second exemple d'exécution du programme `SCALE'
E - Sortie sur traceur du résultat de cette seconde exécution

4 - PROFIL DU FOND DU FOSSE D'AMERIQUE CENTRALE

4.1 - CREATION DES FICHIERS DE DONNEES DE POINTS LE LONG DU FOSSE

A - Exemple de fichier de données original écrit en BASIC
B - Listing du programme `RANPROFL' de conversion du fichier de données écrit en BASIC en deux fichiers de données à accés aléatoire
C - Copie d'écran d'un exemple d'exécution du programme `RANPROFL'

4.2 - CALCUL DE LA POSITION DES POINTS LE LONG DU PROFIL

A - Listing du programme `PROFILE'
B - Copie d'écran d'un exemple d'exécution du programme `PROFILE'
C - Exemple de sortie sur imprimante

4.3 - DESSIN DU PROFIL

A - Listing du programme `MATPROFILE'
B - Copie d'écran de l'exécution du programme `MATPROFILE'
C - Sortie sur traceur réduite de 71 %

5 - RECHERCHE DE CENTRES DE CERCLES SUR LA SPHERE

5.1 - RECHERCHE EN UTILISANT LA DISTANCE

A - Exemple de fichier de données
B - Listing du programme `DISSMCRC'
C - Copie d'écran d'un exemple d'exécution du programme `DISSMCRC'
D - Exemple de sortie sur imprimante

5.2 - RECHERCHE EN UTILISANT LA DIRECTION DES STRUCTURES

A - Exemple de fichier de données
B - Listing du programme `DIRSMCRC'
C - Copie d'écran d'un exemple d'exécution du programme `DIRSMCRC'
D - Exemple de sortie sur imprimante

6 - RELATION PROFONDEUR-DU-FOSSE - RAYON-DU-CERCLE

6.1 - CALCUL DE LA PROFONDEUR MOYENNE DE SEGMENTS DE PROFILS

A - Listing du programme `MEANDEPT'
B - Copie d'écran d'un exemple d'exécution du programme `MEANDEPT'
C - Exemple de sortie sur imprimante

6.2 - DESSIN DU GRAPHIQUE MONTRANT LA RELATION PROFONDEUR-DU-FOSSE - RAYON-DU-CERCLE

A - Listing du programme `RADEPTH'
B - Copie d'écran de l'exécution du programme `RADEPTH'
C - Sortie sur traceur réduite de 67 %

7 - POSITIONS DE POINTS EN FONCTION DE LA DISTANCE ET DE LA DIRECTION, SUR LA SPHERE, DISTANCE LE LONG DE DROITES, DE POINTS PROJETES

7.1 - CALCUL DE LA POSITION DE POINTS SUR LA SPHERE EN FONCTION DE LA DISTANCE ET DE LA DIRECTION

A - Exemple de fichier de données
B - Listing du programme `POINTS'
C - Copie d'écran d'un exemple d'exécution du programme `POINTS'
D - Exemple de sortie sur imprimante

7.2 - CALCUL DE LA DISTANCE LE LONG DE DROITES DE POINTS PROJETES

A - Listing du programme `PROPOINT'
B - Copie d'écran d'un exemple d'exécution du programme `PROPOINT'
C - Exemple de sortie sur imprimante

8 - REPRESENTATION GRAPHIQUE DES RESULTATS SUR DIAGRAMMES X-Y

8.1 - DESSIN DE POINTS SUR UN DIAGRAMME X-Y

A - Exemple de fichier de données
B - Listing du programme `POINT PLOTTING'
C - Copie d'écran d'un exemple d'exécution du programme `POINT PLOTTING'
D - Exemple de sortie sur traceur réduite de 63 %

8.2 - DESSIN DE LIGNES SUR UN DIAGRAMME X-Y

A - Exemple de fichier de données
B - Listing du programme `LINEDRAW.BAS'
C - Copie d'écran d'un exemple d'exécution du programme `LINEDRAW.BAS'
D - Exemple de sortie sur traceur réduite de 71%

9 - REPRESENTATION GRAPHIQUE DES RESULTATS SUR CARTES

9.1 - DESSIN DE POINTS SUR UNE CARTE EN PROJECTION DE MERCATOR

A - Exemple de fichier de données
B - Listing du programme 'MERCATOR'
C - Copie d'écran d'un exemple d'exécution du programme 'MERCATOR'
D - Exemple de sortie sur traceur réduite de 55 %

9.2 - DESSIN DE VECTEURS SUR UNE CARTE EN PROJECTION DE MERCATOR

A - Exemple de fichier de données
B - Listing du programme 'MERVECTR.BAS'
C - Copie d'écran d'un exemple d'exécution du programme 'MERVECTR.BAS'
D - Exemple de sortie sur traceur réduite de 71%

10 - SEISMES : CARTE DES MAGNITUDES ET COUPES

10.1 - ENTREE DES DONNEES DE SEISMICITE

A - Listing du programme 'SEISEDIT'
B - Copie d'écran d'exemples d'exécution du programme 'SEISEDIT' :
  B1 - création du fichier de données
  B2 - correction du fichier de données

10.2 - TRANSFORMATION DU FICHIER DE DONNEES A ACCES SEQUENTIEL EN UN FICHIER DE DONNEES A ACCES ALEATOIRE

A - Listing du programme 'ERSQEDIT'
B - Copie d'écran d'un exemple d'exécution du programme 'ERSQEDIT'

10.3 - CALCUL DE LA MAGNITUDE AUX NOEUDS D'UNE GRILLE GEOGRAPHIQUE

A - Listing du programme 'SEISMES'
B - Copie d'écran d'un exemple d'exécution du programme 'SEISMES'
C - Exemple de sortie sur imprimante

10.4 - TRI DES SEISMES DANS UN POLYGONE POUR LES COUPES DE SEISMES

A - Listing du programme 'SEISSORT'
B - Copie d'écran d'un exemple d'exécution du programme 'SEISSORT'
C - Exemple de sortie sur imprimante

10.5 - DESSIN DES COUPES DE SEISMES

A - Exemple de fichier de données
B - Listing du programme 'CROSSECTION'
C - Copie d'écran d'un exemple d'exécution du programme 'CROSSECTION'
D - Exemple de sortie sur traceur réduite de 63 %

TABLE OF CONTENTS

3 - CONVERSION FROM FRACTIONAL SCALES TO INCHES-PER-DEGREE-OF-LONGITUDE AND INVERSELY

A - Listing of program 'SCALE'
B - Screen copy of a first sample execution of program 'SCALE'
C - Plotter output of the first sample execution
D - Screen copy of a second sample execution of program 'SCALE'
E - Plotter output of the second sample execution

4 - MIDDLE AMERICA TRENCH AXIS PROFILE

4.1 - CREATION OF DATA FILES FOR POINTS ALONG THE TRENCH AXIS

A - Original sample data file written in Basic
B - Listing of program 'RANPROFL' to convert a data file written in Basic into two random access data files
C - Screen copy of a sample execution of program 'RANPROFL'

4.2 - COMPUTATION OF THE POINTS POSITIONS ALONG THE PROFILE

A - Listing of program 'PROFILE'
B - Screen copy of a sample execution of program 'PROFILE'
C - Sample printer output

4.3 - PROFILE DRAWING

A - Listing of program 'MATPROFILE'
B - Screen copy of the execution of program 'MATPROFILE'
C - Plotter output reduced to 71%

5 - SEARCH FOR CIRCLE CENTERS ON A SPHERE

5.1 - SEARCH USING THE DISTANCE

A - Sample data file
B - Listing of program 'DISSMCRC'
C - Screen copy of a sample execution of program 'DISSMCRC'
D - Sample printer output

5.2 - SEARCH USING STRUCTURES DIRECTIONS

A - Sample data file
B - Listing of program 'DIRSMCRC'
C - Screen copy of a sample execution of program 'DIRSMCRC'
D - Sample printer output

6 - RELATIONSHIP BETWEEN DEPTH-OF-THE-TRENCH AND RADIUS-OF-THE-CIRCLE

6.1 - COMPUTATION OF THE MEAN DEPTH OF PROFILE SEGMENTS

A - Listing of program 'MEANDEPT'
B - Screen copy of a sample execution of program 'MEANDEPT'
C - Sample printer output

D - Sample plotter output reduced to 71%

10 - EARTHQUAKES : MAGNITUDE MAPS AND CROSS-SECTIONS

10.1 - INPUT OF EARTHQUAKES DATA

A - Listing of program 'SEISEDIT'
B - Screen copy of sample executions of program 'SEISEDIT' :
B1 - creation of the data file
B2 - correction of the data file

10.2 - TRANSFORMATION OF THE SEQUENTIAL ACCESS DATA FILE INTO A RANDOM ACCESS DATA FILE

A - Listing of program 'ERSQEDIT'
B - Screen copy of a sample execution of program 'ERSQEDIT'

10.3 - COMPUTATION OF MAGNITUDE AT KNOTS OF A GEOGRAPHICAL GRID

A - Listing of program 'SEISMES'
B - Screen copy of a sample execution of program 'SEISMES'
C - Sample printer output

10.4 - SORT OF EARTHQUAKES INTO A POLYGON FOR EARTHQUAKES CROSS-SECTIONS

A - Listing of program 'SEISSORT'
B - Screen copy of a sample execution of program 'SEISSORT'
C - Sample printer output

10.5 - DRAWING OF EARTHQUAKES CROSS-SECTIONS

A - Sample data file
B - Listing of program 'CROSSECTION'
C - Screen copy of a sample execution of program 'CROSSECTION'
D - Sample plotter output reduced to 63%

AVANT-PROPOS

— Ce supplément informatique réunit les différents programmes créés pour réaliser les divers calculs et de nombreuses figures de la thèse. —

Le listing de chaque programme est accompagné d'un exemple de fichier de données lorsqu'il existe. La copie d'écran est exécutée avec le fichier de données servant d'exemple, et les sorties sur imprimante ou sur traceur correspondent à l'exécution du programme. Dans certains cas, les exécutions des programmes sont étroitement liées entre elles, les résultats de l'un servant à l'exécution de l'autre (première, quatrième, sixième, septième et dixième parties).

Les sous-programmes de dessin sur traceur Benson 1332 à partir du Tektronix 4054 ont été recopiés du manuel 'Logiciel Basic Tektronix Série 405X (Séries 02, 12, 32 et 42)' écrit par Benson (Juin 1981)

Dans la partie 'exécution' d'un programme :
une étoile signifie qu'il y a effacement de l'écran ;
les caractères soulignés sont des réponses données par l'opérateur ;
les lettres collées soulignées dans la marge sont des lettres tapées au clavier mais qui n'apparaissent pas à l'écran.

Les notes qui suivent concernent les différents ordinateurs et imprimante utilisés avec leurs particularités. Le point commun à ces outils est le code ASCII utilisé qui ne comprend que 127 caractères.

NOTES SUR LE BASIC DU ZENITH 100 :

Le BASIC du Z100 est un BASIC 'Microsoft' qui ressemble beaucoup aux BASIC utilisés sur le micro-ordinateur IBM PC. Les modifications nécessaires pour transposer les programmes du Zenith 100 sur l'IBM PC ne concernent que le graphisme :

SCREEN a,b avec a = 0 ou 1 (0 = mode texte, 1 = mode graphique)
et b = 0 ou 1 (0 = vidéo normal, 1 = vidéo inverse)

L'écran du Zenith 100 a une hauteur de 225 lignes et une largeur de 640 points en mode graphique. En mode texte il possède 80 colonnes et 25 lignes.

NOTES SUR LE BASIC DU TEKTRONIX 4054 :

Le BASIC du TEKTRONIX 4054 est légèrement différent du BASIC du Z100 :

une ligne ne peut comporter qu'une seule instruction ;
les noms des variables numériques sont limités à une lettre plus éventuellement un chiffre de 0 à 9, ceux des chaines de caractères à une seule lettre suivie de $;
le mot REM ne peut être remplacé par ' ;
il est impossible d'indenter les lignes ;
les fonctions arc-cosinus et arc-sinus existent ;
le mixage d'un autre programme se fait par :
FIND n (n = numéro de fichier sur la bande magnétique)
APPEND nn (nn = numéro de ligne où se placera le fichier)
il est nécessaire de remettre les variables à 0 au début des programmes par INIT ;
pour effacer l'écran il faut utiliser l'instruction PAGE ;
il est possible de définir la taille des caractères par CHARSIZE n (n de 1 à 4 : 1 correspond à la taille la plus petite) ;
le nombre PI est disponible et il est possible de calculer les valeurs des fonctions trigonométriques directement en degrés après avoir écrit l'instruction SET DEGREES ;
l'instruction READ lit directement toutes les données d'un tableau à partir de lignes de DATA : ligne par ligne et dans chaque ligne colonne par colonne ;
les instructions GOTO i OF n1,n2... et GOSUB i OF n1,n2... remplacent ON i GOTO n1,n2... et ON i GOSUB n1,n2... du BASIC `Microsoft', où i est une valeur entière positive et n1 le numéro de ligne de branchement pour i=1, n2 pour i=2, etc. ;
le chainage de 2 chaines de caractères se fait avec & ;
l'instruction SEG(a$,n,n1) extrait une sous-chaine de n1 caractères de la chaine A$ à partir de la lettre n ;
l'instruction STR(i) convertit le chiffre i en une chaine de caractères.

Certaines `lourdeurs' sont dues au langage :
l'instruction INPUT ne peut comporter de texte : il faut donc généralement utiliser 2 lignes (une avec un PRINT pour poser la question et une autre avec INPUT pour recevoir la réponse) ;
...

L'unité pour le traceur BENSON est le centimètre.

NOTES SUR L'IMPRIMANTE `APPLE DOT MATRIX' :

L'imprimante Apple est une imprimante d'une largeur de 80 colonnes en écriture Pica. Les différentes options de l'imprimante sont contrôlées par la réception de chaines de `caractères' bien définies, avec l'instruction LPRINT en BASIC on a :

```
LPRINT CHR$(27)"Lnnn" : marge gauche de nnn caractères
LPRINT CHR$(27)"n" : 9 caractères par pouce
LPRINT CHR$(27)"N" : 10 caractères par pouce
```

LPRINT CHR$(27)"E" : 12 caractères par pouce
LPRINT CHR$(27)"q" : 15 caractères par pouce
LPRINT CHR$(27)"Q" : 17 caractères par pouce
LPRINT CHR$(27)"Tnn" : interligne de nn/144 pouce
LPRINT CHR$(27)"A" : retour à l'interligne 1/6 pouce
LPRINT CHR$(27)"!" : sélectionne l'écriture en caractères gras
LPRINT CHR$(27)CHR$(34) : stoppe l'écriture en caractères gras
LPRINT CHR$(14) : sélectionne l'impression en caractères dilatés
LPRINT CHR$(15) : stoppe l'écriture en caractères dilatés
LPRINT CHR$(27)"X" : commence le soulignage
LPRINT CHR$(27)"Y" : stoppe le soulignage
LPRINT CHR$(27)"Gnnnn"A$: imprime les nnnn caractères graphiques contenus dans la chaine de caractères A$ et codés sous forme binaire.

(CHR$(27) est le code pour ESCAPE)

FOREWORD

This 'program' supplement presents the different programs created to make various calculations and many figures of the thesis.

For each program its listing and a sample of data file if needed are shown. The screen copy of the program execution is made with the sample data file, and the printer or plotter outputs represent the program execution outputs. In some cases several programs executions are linked : results of a first one is used by a second one (parts 1, 4, 6, 7 and 10).

The drawing subroutines for Benson plotter 1332 from Tektronix 4054 computer have been copied from 'Logiciel Basic Tektronix Série 405X (Séries 02, 12, 32 et 42)' written by Benson (June 1981).

In the 'execution' part of a program :
- an asterisk means that the screen is cleared ;
- underlined characters are answers given by the operator ;
- underlined pressed letters in the margin are letters typed on the keyboard but not echoed on the screen.

Following notes are concerned with the various computers and printer used for this thesis and their peculiarities. The common property of these instruments is the ASCII table which has only 127 'characters'.

NOTES ON ZENITH 100 BASIC :

Zenith 100 BASIC is a 'Microsoft' BASIC which is very close to the version used on IBM micro-computers. To translate these programs for IBM-PC it is only necessary to change a few graphic orders :

SCREEN a,b : a = 0 or 1 (0 = text mode, 1 = graphic mode)
b = 0 or 1 (0 = normal video, 1 = reverse video)

The Zenith screen has a 225 lines height and a 640 points width in graphic mode. In text mode it has 80 columns and 25 lines.

NOTES ON TEKTRONIX 4054 BASIC :

Tektronix 4054 BASIC is slightly different from Z100 BASIC :

a line may have only one instruction ;
the numerical variable names are limited to one letter plus if wanted one number from 0 to 9, and character variable names have only one letter followed by $;
the word REM can not be replaced by ' ;
indentation of lines is impossible ;
arc-cosine and arc-sine functions exist ;
to merge a program, type :
FIND n (n = file number on the magnetic tape)
APPEND nn (nn = line number where to place the file)
it is necessary to initialize the variables at the beginning of a program with INIT ;
the PAGE statement clears the screen ;
it is possible to define the character size on the screen by using CHARSIZE n (n = 1 to 4, with 1 meaning the smallest size) ;
PI number is present and it is possible to compute all trigonometric functions in degrees by writing SET DEGREES at the beginning of the program ;
the READ statement allows to read directly all the data of a table from DATA lines, line by line and inside each line column by column ;
GOTO i OF n1,n2... and GOSUB i OF n1,n2... replace `Microsoft' ON i GOTO n1,n2... and ON i GOSUB n1,n2..., where i is an integer variable varying from 1 to ... and n1 the line number where the program must go when i=1 (n2 for i=2, etc.) ;
to link 2 character strings one must use & ;
the SEG(a$,n,n1) statement extracts a n1 characters long substring from string a$ beginning at the nth letter ;
STR(i) converts the number i into a character string.

Some `ponderousnesses' come from the language :
the INPUT statement cannot have text, so one must usually use 2 lines (one with PRINT to ask the question, and one with INPUT to receive the answer) ;
...

The unit for the BENSON plotter is the centimeter.

NOTES ON THE `APPLE DOT MATRIX' PRINTER :

The Apple printer has 80 columns when using Pica pitch. The printer various options are controled by special `character' strings used with LPRINT command in BASIC :

```
LPRINT CHR$(27)"Lnnn" : set left margin to nnn characters
LPRINT CHR$(27)"n" : 9 characters per inch
LPRINT CHR$(27)"N" : 10 characters per inch
LPRINT CHR$(27)"E" : 12 characters per inch
LPRINT CHR$(27)"q" : 15 characters per inch
LPRINT CHR$(27)"Q" : 17 characters per inch
LPRINT CHR$(27)"Tnn" : distance between lines to be nn/144 inch
LPRINT CHR$(27)"A" : line feed pitch of 6 lines per inch
LPRINT CHR$(27)"!" : start boldface printing
LPRINT CHR$(27)CHR$(34) : end boldface printing
```

```
LPRINT CHR$(14) : start headline mode
LPRINT CHR$(15) : end headline mode
LPRINT CHR$(27)"X" : start underlining text
LPRINT CHR$(27)"Y" : end underlining text
LPRINT CHR$(27)"Gnnnn"A$  :  print the nnnn graphic characters of the
     binary coded string A$.

(CHR$(27) means ESCAPE)
```

CORRECTION
DES DONNEES MAGNETIQUES

1.1 - INTRODUCTION DES POINTS DE CORRECTION DE LA NAVIGATION SEABEAM

A - Listing du programme 'CBEAMNAV'

B - Copie d'écran d'un exemple d'exécution du programme 'CBEAMNAV'

C - Exemple de sortie sur imprimante

1.2 - INTRODUCTION DE TOUS LES POINTS DE MESURE DU MAGNETISME

A - Listing du programme 'ORIGNAV'

B - Copie d'écran d'un exemple d'exécution du programme 'ORIGNAV'

C - Exemple de sortie sur imprimante

1.3 - POSITION CORRIGEE ET CHAMP MAGNETIQUE DES POINTS DE MESURE

A - Listing du programme 'NAVMAGCR'

B - Copie d'écran d'un exemple d'exécution du programme 'NAVMAGCR'

C - Exemple de sortie sur imprimante

1.4 - INTRODUCTION DES POINTS D'ANOMALIE MAGNETIQUE CORRIGEE

A - Listing du programme 'ANOCOREC'

B - Copie d'écran d'un exemple d'exécution du programme 'ANOCOREC'

1.5 - CALCUL DE L'ANOMALIE MAGNETIQUE DE L'ENSEMBLE DES POINTS DE MESURE

A - Listing du programme 'MAGCOREC'

B - Copie d'écran d'un exemple d'exécution du programme 'MAGCOREC'

C - Exemple de sortie sur imprimante

1.1

A

```
*-----------------------------------------------------------------------*
*                                                                       *
*                          PROGRAM 'CBEAMNAV'                           *
*                                                                       *
*                                        Jacqueline ROUMP, Dec. 1984    *
*-----------------------------------------------------------------------*

      PROGRAM CBEAMNAV

*
* This program has been written to put the navigation correction
* parameters determined from Scripps Seabeam maps into a sequential
* data file and after the input of these data to translate them into
* a format easier for computational use.

* The original data are : year (YY), month (MM), day (DD), time (input
* in HHMM format), and differences of latitude and longitude in inches
* (dependent of scale) and in X.XX format ; these data will be put into
* a data file named SBNAVDM.DAT.

* You are allowed to stop introducing the data, and to resume the data
* input during another session. Before the translation of this data you
* will be able to correct them using the editor.

* As the translated data will be used to correct the position of other
* points of the survey, and to compute the real magnetic anomaly, the
* second data file (SBNAVDD.DAT) will have decimal year (year+day+time),
* and differences of latitude and longtude in degrees.

*
*  PREPARATION
*  -----------
*
      INTEGER N,I,YEAR,MONTH,DAY,TIME,ANSWER,MONLEN(11),NDAY,N1,PAGE
      REAL NSINCH,EWINCH,SCALE,MLAT,HOUR,HR,DDEC,ML,NSDEC,EWDEC
      REAL*8 MERCP,YDEC,PI,MLATRD
      CHARACTER R,TITLE*40,TIT(40),ESC,SO,SI,FF

      EQUIVALENCE (TITLE,TIT)

      DATA MONLEN /31,28,31,30,31,30,2*31,30,31,30/

      OPEN (1, FILE='SBNAVDM.DAT')
      OPEN (2, FILE='SBNAVDD.DAT', STATUS='NEW')
      OPEN (3, FILE='PRN')

*
*  FORMATS
*  -------
* 1 : to write and read the original data file
* 2 : to write and read the translated data file
* 3 : to write the original and translated data on the printer
*
   1  FORMAT (1X,3(I2,1X),I4,2(1X,F5.2))
   2  FORMAT (1X,F10.7,2(1X,F8.5))
   3  FORMAT (1X,I4,4X,I2,5X,I2,4X,I2,2X,I4,2(2X,F5.2),8X,F10.7,2(2X,F8.
     +5))
*
*  PRESENTATION
*  ------------
*
      WRITE (*,'(A//)') '                              PROGRAM CBEAMNAV'
      WRITE (*,'(A)') ' This program has been written to put the navigat
     +ion correction'
      WRITE (*,'(A)') ' parameters determined from Scripps Seabeam maps
     +into a sequential'
      WRITE (*,'(A)') ' data file and after the input of these data to t
     +ranslate them into'
      WRITE (*,'(A/)') ' a format easier for computational use.'
      WRITE (*,'(A)') ' The original data are : year (YY), month (MM), d
     +ay (DD), time (input'
      WRITE (*,'(A)') ' in HHMM format), and differences of latitude and
     + longitude in inches'
      WRITE (*,'(A)') ' (dependent of scale) and in X.XX format ; these
     +data will be put into'
      WRITE (*,'(A/)') ' a data file named SBNAVDM.DAT.'
      WRITE (*,'(A)') ' You are allowed to stop introducing the data, an
     +d to resume the data'
      WRITE (*,'(A)') ' input during another session. Before the transla
     +tion of this data'
      WRITE (*,'(A/)') ' you will be able to correct them using the edit
     +or.'
      WRITE (*,'(A)') ' As the translated data will be used to correct t
     +he position of other'
      WRITE (*,'(A)') ' points of the survey, and to compute the real ma
     +gnetic anomaly, the'
      WRITE (*,'(A)') ' second data file (SBNAVDD.DAT) will have decimal
     + year (year+day+time),'
      WRITE (*,'(A//)') ' and differences of latitude and longitude in d
     +egrees.'

*
*  PART CHOICE : INPUT OF ORIGINAL DATA OR TRANSLATION
*  ---------------------------------------------------
*
   4  WRITE (*,'(A\)') ' Do you want to input original data (1) or to tr
     +anslate them (2) ? '
      READ (*,'(I1)') ANSWER
          IF ((ANSWER .NE. 1) .AND. (ANSWER .NE. 2)) GOTO 4
          IF (ANSWER .EQ. 2) GOTO 8

*
*  ORIGINAL DATA INPUT PART
*  ------------------------
*
      WRITE (*,'(/A/)') ' Now you will input your data, for the last one
     +, type 0 for the YEAR'
      WRITE (*,'(A\)') ' What is the number of the first point to be inp
     +ut : '
      READ (*,'(I5)') N
          IF (N .EQ. 1) GOTO 6
*
* Reading the beginning of the file
*
```

```
      DO 5 J=1,N-1
   5     READ (1,1) YEAR,MONTH,DAY,TIME,NSINCH,EWINCH
*
* Writing the data
*
   6  WRITE (*,'(/A,I5)') '  POINT ', N
      WRITE (*,'(A\)') ' Year (YY) ? '
      READ (*,'(I2)') YEAR
          IF (YEAR .EQ. 0) GOTO 7
      WRITE (*,'(A\)') ' Month (MM) ? '
      READ (*,'(I2)') MONTH
      WRITE (*,'(A\)') ' Day (DD) ? '
      READ (*,'(I2)') DAY
      WRITE (*,'(A\)') ' Time (HHMM) ? '
      READ (*,'(I4)') TIME
      WRITE (*,'(A\)') ' NS difference (X.XX) ? '
      READ (*,'(F5.2)') NSINCH
      WRITE (*,'(A\)') ' EW difference (X.XX) ? '
      READ (*,'(F5.2)') EWINCH
      WRITE (1,1) YEAR, MONTH, DAY, TIME, NSINCH, EWINCH
      N=N+1
      GOTO 6

   7  WRITE (*,'(A\)') ' Is it the last point of the whole file (Y/N) ?
     + '
      READ (*,'(A)') R
          IF ((R .NE. 'Y') .AND. (R .NE. 'y')) GOTO 20
      ENDFILE 1

      WRITE (*,'(A\)') ' Do you want to translate the original data file
     + immediatly (Y/N) ? '
      READ (*,'(A)') R
          IF ((R .NE. 'Y') .AND. (R .NE. 'y')) GOTO 20
      REWIND 1

*
*  TRANSLATION PART
*  ----------------
*  MERCP is the WGS72 ellipsoid parameter

   8  WRITE (*,'(/A\)') ' Scale (in/deg) of the map used to measure the
     +differences (XXX.XXX) ? '
      READ (*,'(F7.3)') SCALE
      WRITE (*,'(A\)') ' Mean latitude of the survey (DD.MMm) '
      READ (*,'(F7.3)') MLAT
      WRITE (*,'(A)') ' Title of the study (up to 40 characters) ? '
      READ (*,'(A)') TITLE

      MERCP=0.99333056822
      PI=3.141592653589887
      ML=FLOAT(IFIX(MLAT))
      MLATRD=(ML+(MLAT-ML)*100/60)*PI/180.
   9  READ (1,1,END=11) YEAR,MONTH,DAY,TIME,NSINCH,EWINCH
*
* Computing the decimal year
*
      NDAY=0
      HOUR=FLOAT(TIME)/100
      HR=FLOAT(IFIX(HOUR))
      DDEC=(HR+(HOUR-HR)*100/60)/24
      DO 10 I=1,MONTH-1
  10      NDAY=MONLEN(I)+NDAY
      YDEC=YEAR + (NDAY+(DAY-1)+DDEC)/365
*
* Computing the differences of latitudes and longitudes in degrees
*
      NSDEC=NSINCH*SNGL(DCOS(DATAN(MERCP*DTAN(MLATRD))))/SCALE
      EWDEC=EWINCH/SCALE
*
* Writing the data into the file
*
      WRITE (2,2) YDEC, NSDEC, EWDEC
      GOTO 9
*
* Preparing the printing part
*
  11  ENDFILE 2
      REWIND 1
      REWIND 2
*
* Preparing the title
*
      DO 12  I = 40, 1, -1
           IF (TIT(I) .NE. ' ') GOTO 13
  12  CONTINUE
*
*  PRINTING PART
*  -------------
*
  13  WRITE (*,'(//A\)') ' PRINTER READY (Y/N) ? '
      READ (*,'(A)') R
          IF ((R .NE. 'Y') .AND. (R .NE. 'y')) GOTO 20
*
* Control characters for the printer
*
      ESC=CHAR(27)
      SO=CHAR(14)
      SI=CHAR(15)
      FF=CHAR(12)
*
* Printing the title and comments
*
      WRITE (3,'(1X,7A)') ESC,'E',ESC,'L007',ESC,'!',SO
      N1=INT((40-I)/2)+1
      DO 14  N = 1, N1
  14     WRITE (3,'(A\)') ' '
      WRITE (3,'(2A\)') ESC,'X'
      DO 15  N = 1, I
  15     WRITE (3,'(A\)') TIT(N)
      WRITE (3,'(2A)') ESC,'Y'
      WRITE (3,'(1X,3A////)') SI,ESC,'N'
      WRITE (3,'(1X,A)') 'SEABEAM NAVIGATION CORRECTION DATA'
      WRITE (3,'(1X,4A)') ESC,'*',ESC,'E'
      WRITE (3,'(1X,A,F7.3,A)') 'Scale of the original map : ',SCALE,' i
     +nch/deg'
      WRITE (3,'(1X,A,F7.3,A//)') 'Mean latitude of the map :  ',MLAT,
```

```
     +' deg.mm'
*
* Printing the column headings and data
*
      I=1
      PAGE=1
  16  WRITE (3,'(1X,A)') 'POINT  YEAR  MONTH  DAY  TIME  NSDIF  EWDIF
     +        TIME       NS.DIF    EW.DIF  '
      WRITE (3,'(1X,A/)') ' no.    yy     mm    dd  hhmm   inch   inch
     +       dec.year     degree    degree '

  17  READ (1,1,END=19) YEAR,MONTH,DAY,TIME,NSINCH,EWINCH
      READ (2,2) YDEC,NSDEC,EWDEC
      WRITE (3,3) I,YEAR,MONTH,DAY,TIME,NSINCH,EWINCH,YDEC,NSDEC,EWDEC
      I=I+1
          IF ((PAGE .EQ. 1) .AND. (I .LT. 49)) GOTO 17
          IF ((PAGE .EQ. 1) .AND. (I .EQ. 49)) GOTO 18
          IF (I .NE. (49+(PAGE-1)*61)) GOTO 17
*
*  Page changing
*
  18  PAGE=PAGE+1
      WRITE (3,'(1X,A)') FF
      GOTO 16

*
*  END OF THE PROGRAM
*  ------------------
*
  19  WRITE (3,'(/)')
  20  CLOSE (1)
      CLOSE (2)
      END
```

B

★

PROGRAM CBEAMNAV

This program has been written to put the navigation correction parameters determined from Scripps Seabeam maps into a sequential data file and after the input of these data to translate them into a format easier for computational use.

The original data are : year (YY), month (MM), day (DD), time (input in HHMM format), and differences of latitude and longitude in inches (dependent of scale) and in X.XX format ; these data will be put into a data file named SBNAVDM.DAT.

You are allowed to stop introducing the data, and to resume the data input during another session. Before the translation of this data you will be able to correct them using the editor.

As the translated data will be used to correct the position of other points of the survey, and to compute the real magnetic anomaly, the second data file (SBNAVDD.DAT) will have decimal year (year+day+time), and differences of latitude and longitude in degrees.

Do you want to input original data (1) or to translate them (2) ? 1

Now you will input your data, for the last one, type 0 for the YEAR

What is the number of the first point to be input : 1

```
 POINT     1
Year (YY) ? 82
Month (MM) ? 4
Day (DD) ? 7
Time (HHMM) ? 800
NS difference (X.XX) ? .00
EW difference (X.XX) ? .00

 POINT     2
Year (YY) ? 82
Month (MM) ? 4
Day (DD) ? 7
Time (HHMM) ? 956
NS difference (X.XX) ? .33
EW difference (X.XX) ? .21
```

```
 POINT     3
Year (YY) ? 82
Month (MM) ? 4
Day (DD) ? 7
Time (HHMM) ? 1102
NS difference (X.XX) ? .02
EW difference (X.XX) ? .17

 POINT     4
Year (YY) ? 82
Month (MM) ? 4
Day (DD) ? 7
Time (HHMM) ? 1148
NS difference (X.XX) ? .23
EW difference (X.XX) ? .15

 POINT     5
Year (YY) ? 82
Month (MM) ? 4
Day (DD) ? 7
Time (HHMM) ? 1235
NS difference (X.XX) ? .40
EW difference (X.XX) ? -.38

 POINT     6
Year (YY) ? 82
Month (MM) ? 4
Day (DD) ? 7
Time (HHMM) ? 1323
NS difference (X.XX) ? .19
EW difference (X.XX) ? -.25

 POINT     7
Year (YY) ? 82
Month (MM) ? 4
Day (DD) ? 7
Time (HHMM) ? 1337
NS difference (X.XX) ? .13
EW difference (X.XX) ? -.18
```

.
.
.
.
.

```
 POINT   160
Year (YY) ? 82
Month (MM) ? 4
Day (DD) ? 9
Time (HHMM) ? 920
NS difference (X.XX) ? .23
EW difference (X.XX) ? -.05

 POINT   161
Year (YY) ? 82
Month (MM) ? 4
Day (DD) ? 9
Time (HHMM) ? 953
NS difference (X.XX) ? .00
EW difference (X.XX) ? .00

 POINT   162
Year (YY) ? 82
Month (MM) ? 4
Day (DD) ? 9
Time (HHMM) ? 954
NS difference (X.XX) ? -.51
EW difference (X.XX) ? -.26

 POINT   163
Year (YY) ? 82
Month (MM) ? 4
Day (DD) ? 9
Time (HHMM) ? 1010
NS difference (X.XX) ? -.60
EW difference (X.XX) ? -.24

 POINT   164
Year (YY) ? 82
Month (MM) ? 4
Day (DD) ? 9
Time (HHMM) ? 1055
NS difference (X.XX) ? -.67
EW difference (X.XX) ? -.24

 POINT   165
Year (YY) ? 82
Month (MM) ? 4
Day (DD) ? 9
Time (HHMM) ? 1147
NS difference (X.XX) ? -.50
EW difference (X.XX) ? .39
```

```
 POINT   166
Year (YY) ? 82
Month (MM) ? 4
Day (DD) ? 9
Time (HHMM) ? 1148
NS difference (X.XX) ? -.10
EW difference (X.XX) ? -.26

 POINT   167
Year (YY) ? 82
Month (MM) ? 4
Day (DD) ? 9
Time (HHMM) ? 1226
NS difference (X.XX) ? .38
EW difference (X.XX) ? -.10

 POINT   168
Year (YY) ? 82
Month (MM) ? 4
Day (DD) ? 9
Time (HHMM) ? 1227
NS difference (X.XX) ? .11
EW difference (X.XX) ? -.25

 POINT   169
Year (YY) ? 82
Month (MM) ? 4
Day (DD) ? 9
Time (HHMM) ? 1404
NS difference (X.XX) ? .00
EW difference (X.XX) ? .00

 POINT   170
Year (YY) ? 0
Is it the last point of the whole file (Y/N) ? Y
Do you want to translate the original data file immediatly (Y/N) ? Y

Scale (in/deg) of the map used to measure the differences (XXX.XXX) ? 80
Mean latitude of the survey (DD.MMm) 9.10
Title of the study (up to 40 characters) ?
COSTA RICA I SURVEY

PRINTER READY (Y/N) ? Y
```

C

COSTA RICA I SURVEY

SEABEAM NAVIGATION CORRECTION DATA

Scale of the original map : 80.000 inch/deg
Mean latitude of the map : 9.100 deg.mm

POINT no.	YEAR yy	MONTH mm	DAY dd	TIME hhmm	NSDIF inch	EWDIF inch	TIME dec.year	NS.DIF degree	EW.DIF degree
1	82	4	7	800	.00	.00	82.2639269	.00000	.00000
2	82	4	7	956	.33	.21	82.2641476	.00407	.00262
3	82	4	7	1102	.02	.17	82.2642732	.00025	.00213
4	82	4	7	1148	.23	.15	82.2643607	.00284	.00188
5	82	4	7	1235	.40	-.38	82.2644502	.00494	-.00475
6	82	4	7	1323	.19	-.25	82.2645415	.00235	-.00313
7	82	4	7	1337	.13	-.18	82.2645681	.00160	-.00225
8	82	4	7	1346	.10	-.17	82.2645852	.00123	-.00213
9	82	4	7	1408	.02	.08	82.2646271	.00025	.00100
10	82	4	7	1415	.01	.12	82.2646404	.00012	.00150
11	82	4	7	1427	-.03	.19	82.2646632	-.00037	.00237
12	82	4	7	1450	-.13	.63	82.2647070	-.00160	.00788
13	82	4	7	1459	-.10	.69	82.2647241	-.00123	.00862
14	82	4	7	1545	.02	.24	82.2648116	.00025	.00300
15	82	4	7	1611	.07	.29	82.2648611	.00086	.00362
16	82	4	7	1645	.21	-.21	82.2649258	.00259	-.00262
17	82	4	7	1700	.12	-.29	82.2649543	.00148	-.00362
18	82	4	7	1731	.05	.05	82.2650133	.00062	.00062
19	82	4	7	1755	-.28	.15	82.2650590	-.00346	.00188
20	82	4	7	1820	-.25	.06	82.2651065	-.00309	.00075
21	82	4	7	1835	-.22	-.02	82.2651351	-.00272	-.00025
22	82	4	7	1901	-.15	-.13	82.2651846	-.00185	-.00162
23	82	4	7	1916	-.12	-.15	82.2652131	-.00148	-.00188
24	82	4	7	1929	.48	-.31	82.2652378	.00592	-.00387
25	82	4	7	2010	.26	-.09	82.2653158	.00321	-.00113
26	82	4	7	2030	.16	.03	82.2653539	.00197	.00038
27	82	4	7	2050	.05	.15	82.2653919	.00062	.00188
28	82	4	7	2115	.02	.41	82.2654395	.00025	.00513
29	82	4	7	2135	.06	.88	82.2654775	.00074	.01100
30	82	4	7	2155	.06	.86	82.2655156	.00074	.01075
31	82	4	7	2207	.05	.86	82.2655384	.00062	.01075
32	82	4	7	2215	.09	.82	82.2655537	.00111	.01025
33	82	4	7	2237	.12	.82	82.2655955	.00148	.01025
34	82	4	7	2247	.00	.69	82.2656145	.00000	.00862
35	82	4	7	2300	.03	.61	82.2656393	.00037	.00763
36	82	4	7	2325	-.26	.74	82.2656868	-.00321	.00925
37	82	4	7	2349	-.11	.62	82.2657325	-.00136	.00775
38	82	4	8	11	-.22	.55	82.2657744	-.00272	.00687
39	82	4	8	21	-.29	.52	82.2657934	-.00358	.00650
40	82	4	8	40	-.42	.38	82.2658295	-.00518	.00475
41	82	4	8	56	-.48	.35	82.2658600	-.00592	.00438
42	82	4	8	110	-.23	-.15	82.2658866	-.00284	-.00188
43	82	4	8	147	-.13	-.80	82.2659570	-.00160	-.01000
44	82	4	8	220	-.03	-.89	82.2660198	-.00037	-.01113
45	82	4	8	255	.14	-.78	82.2660864	.00173	-.00975
46	82	4	8	309	.35	.06	82.2661130	.00432	.00075
47	82	4	8	355	.27	.23	82.2662005	.00333	.00288
48	82	4	8	404	.23	.27	82.2662177	.00284	.00338

POINT no.	YEAR yy	MONTH mm	DAY dd	TIME hhmm	NSDIF inch	EWDIF inch	TIME dec.year	NS.DIF degree	EW.DIF degree
49	82	4	8	410	-.01	.16	82.2662291	-.00012	.00200
50	82	4	8	420	-.02	.18	82.2662481	-.00025	.00225
51	82	4	8	516	-.07	.27	82.2663546	-.00086	.00338
52	82	4	8	525	.12	.50	82.2663718	.00148	.00625
53	82	4	8	606	-.07	.78	82.2664498	-.00086	.00975
54	82	4	8	641	.33	.34	82.2665164	.00407	.00425
55	82	4	8	704	.00	.10	82.2665601	.00000	.00125
56	82	4	8	710	.00	.00	82.2665715	.00000	.00000
57	82	4	8	731	.02	-.18	82.2666115	.00025	-.00225
58	82	4	8	809	.24	-.01	82.2666838	.00296	-.00012
59	82	4	8	814	.17	-.10	82.2666933	.00210	-.00125
60	82	4	8	910	-.19	-.16	82.2667998	-.00235	-.00200
61	82	4	8	920	-.27	.43	82.2668189	-.00333	.00538
62	82	4	8	944	-.10	.42	82.2668645	-.00123	.00525
63	82	4	8	950	-.02	.35	82.2668760	-.00025	.00438
64	82	4	8	1014	-.07	-.11	82.2669216	-.00086	-.00137
65	82	4	8	1022	-.08	-.06	82.2669368	-.00099	-.00075
66	82	4	8	1045	-.15	.19	82.2669806	-.00185	.00237
67	82	4	8	1101	-.11	.14	82.2670110	-.00136	.00175
68	82	4	8	1121	-.09	.05	82.2670491	-.00111	.00062
69	82	4	8	1122	.00	.00	82.2670510	.00000	.00000
70	82	4	8	1127	.19	.14	82.2670605	.00235	.00175
71	82	4	8	1140	.12	.20	82.2670852	.00148	.00250
72	82	4	8	1150	.05	.23	82.2671043	.00062	.00288
73	82	4	8	1201	-.06	.17	82.2671252	-.00074	.00213
74	82	4	8	1220	-.28	.36	82.2671613	-.00346	.00450
75	82	4	8	1245	.10	.36	82.2672089	.00123	.00450
76	82	4	8	1255	.19	.11	82.2672279	.00235	.00137
77	82	4	8	1300	-.30	.32	82.2672374	-.00370	.00400
78	82	4	8	1320	-.31	.41	82.2672755	-.00383	.00513
79	82	4	8	1335	-.13	.20	82.2673040	-.00160	.00250
80	82	4	8	1351	-.06	.08	82.2673345	-.00074	.00100
81	82	4	8	1408	.00	.00	82.2673668	.00000	.00000
82	82	4	8	1443	.19	-.37	82.2674334	.00235	-.00463
83	82	4	8	1450	.26	-.13	82.2674467	.00321	-.00162
84	82	4	8	1521	.09	.07	82.2675057	.00111	.00088
85	82	4	8	1536	-.07	.14	82.2675342	-.00086	.00175
86	82	4	8	1545	-.13	.20	82.2675514	-.00160	.00250
87	82	4	8	1605	-.08	.36	82.2675894	-.00099	.00450
88	82	4	8	1615	-.09	.48	82.2676084	-.00111	.00600
89	82	4	8	1625	-.11	.62	82.2676275	-.00136	.00775
90	82	4	8	1640	.03	.53	82.2676560	.00037	.00662
91	82	4	8	1650	.09	.44	82.2676750	.00111	.00550
92	82	4	8	1656	.00	.63	82.2676865	.00000	.00788
93	82	4	8	1701	-.21	.51	82.2676960	-.00259	.00637
94	82	4	8	1702	.03	.37	82.2676979	.00037	.00463
95	82	4	8	1721	-.08	.43	82.2677340	-.00099	.00538
96	82	4	8	1736	-.14	.47	82.2677626	-.00173	.00587
97	82	4	8	1750	-.05	.29	82.2677892	-.00062	.00362
98	82	4	8	1810	-.01	.18	82.2678272	-.00012	.00225
99	82	4	8	1830	.00	.00	82.2678653	.00000	.00000
100	82	4	8	1840	.23	-.11	82.2678843	.00284	-.00137
101	82	4	8	1848	.26	-.35	82.2678995	.00321	-.00438
102	82	4	8	1857	-.25	.19	82.2679167	-.00309	.00237
103	82	4	8	1940	-.02	.12	82.2679985	-.00025	.00150
104	82	4	8	1955	-.12	.27	82.2680270	-.00148	.00338
105	82	4	8	2005	-.07	.22	82.2680460	-.00086	.00275
106	82	4	8	2020	-.02	.16	82.2680746	-.00025	.00200
107	82	4	8	2050	.15	.32	82.2681317	.00185	.00400
108	82	4	8	2102	.03	.31	82.2681545	.00037	.00387
109	82	4	8	2103	.02	.16	82.2681564	.00025	.00200
110	82	4	8	2115	.02	.14	82.2681792	.00025	.00175
111	82	4	8	2135	.02	.16	82.2682173	.00025	.00200
112	82	4	8	2156	-.04	.15	82.2682572	-.00049	.00188
113	82	4	8	2221	-.25	.28	82.2683048	-.00309	.00350
114	82	4	8	2239	-.34	.43	82.2683390	-.00420	.00538
115	82	4	8	2244	-.34	.48	82.2683486	-.00420	.00600
116	82	4	8	2255	-.38	.53	82.2683695	-.00469	.00662
117	82	4	8	2316	-.64	.42	82.2684094	-.00790	.00525
118	82	4	8	2326	-.69	.46	82.2684285	-.00852	.00575
119	82	4	8	2341	-.63	.44	82.2684570	-.00778	.00550
120	82	4	9	0	-.24	.21	82.2684932	-.00296	.00262
121	82	4	9	3	-.18	.02	82.2684989	-.00222	.00025
122	82	4	9	11	-.27	-.04	82.2685141	-.00333	-.00050
123	82	4	9	12	-.18	-.33	82.2685160	-.00222	-.00412
124	82	4	9	41	-.37	-.05	82.2685712	-.00457	-.00062
125	82	4	9	49	-.04	-.01	82.2685864	-.00049	-.00012
126	82	4	9	116	-.07	.04	82.2686377	-.00086	.00050
127	82	4	9	155	.37	-.13	82.2687119	.00457	-.00162
128	82	4	9	231	.32	-.16	82.2687804	.00395	-.00200
129	82	4	9	245	.40	-.16	82.2688071	.00494	-.00200
130	82	4	9	300	.41	-.14	82.2688356	.00506	-.00175
131	82	4	9	310	.41	-.14	82.2688546	.00506	-.00175
132	82	4	9	326	.53	-.16	82.2688851	.00654	-.00200
133	82	4	9	332	.55	-.14	82.2688965	.00679	-.00175
134	82	4	9	333	.57	.15	82.2688984	.00704	.00188
135	82	4	9	345	.29	.32	82.2689212	.00358	.00400
136	82	4	9	355	.18	.44	82.2689403	.00222	.00550
137	82	4	9	420	.00	.00	82.2689878	.00000	.00000
138	82	4	9	432	-.05	-.21	82.2690107	-.00062	-.00262
139	82	4	9	433	.33	-.39	82.2690126	.00407	-.00487
140	82	4	9	441	.39	-.45	82.2690278	.00481	-.00562
141	82	4	9	451	.47	-.42	82.2690468	.00580	-.00525
142	82	4	9	500	.58	-.37	82.2690639	.00716	-.00463
143	82	4	9	510	-.07	-.43	82.2690830	-.00086	-.00538
144	82	4	9	516	-.08	-.39	82.2690944	-.00099	-.00487
145	82	4	9	526	-.07	-.29	82.2691134	-.00086	-.00362
146	82	4	9	533	.02	-.22	82.2691267	.00025	-.00275
147	82	4	9	556	.02	-.05	82.2691705	.00025	-.00062
148	82	4	9	626	.00	-.08	82.2692275	.00000	-.00100
149	82	4	9	646	.01	.16	82.2692656	.00012	.00200
150	82	4	9	651	.12	.33	82.2692751	.00148	.00412
151	82	4	9	705	.23	.27	82.2693018	.00284	.00338
152	82	4	9	716	.24	.29	82.2693227	.00296	.00362
153	82	4	9	731	.12	.29	82.2693512	.00148	.00362
154	82	4	9	745	.09	.34	82.2693779	.00111	.00425
155	82	4	9	747	.07	.07	82.2693817	.00086	.00088
156	82	4	9	802	.00	.00	82.2694102	.00000	.00000
157	82	4	9	839	.07	-.10	82.2694806	.00086	-.00125
158	82	4	9	840	.26	-.08	82.2694825	.00321	-.00100
159	82	4	9	900	.18	-.18	82.2695205	.00222	-.00225
160	82	4	9	920	.23	-.05	82.2695586	.00284	-.00062
161	82	4	9	953	.00	.00	82.2696214	.00000	.00000
162	82	4	9	954	-.51	-.26	82.2696233	-.00629	-.00325
163	82	4	9	1010	-.60	-.24	82.2696537	-.00741	-.00300
164	82	4	9	1055	-.67	-.24	82.2697393	-.00827	-.00300
165	82	4	9	1147	-.50	.39	82.2698383	-.00617	.00487
166	82	4	9	1148	-.10	-.26	82.2698402	-.00123	-.00325
167	82	4	9	1226	.38	-.10	82.2699125	.00469	-.00125
168	82	4	9	1227	.11	-.25	82.2699144	.00136	-.00313
169	82	4	9	1404	.00	.00	82.2700989	.00000	.00000

1.2
A

```
*-----------------------------------------------------------------------*
*                                                                       *
*                          PROGRAM 'ORIGNAV'                            *
*                                                                       *
*                                        Jacqueline ROUMP, Dec. 1984    *
*-----------------------------------------------------------------------*

      PROGRAM ORIGNAV


* This program has been written to put the navigation of Scripps
* Seabeam maps and the recorded magnetic field into a sequential data
* file and after the input of these data to translate them into a
* easier for computational use.

* The original data are : year (YY), month (MM), day (DD), time (input
* in HHMM format), latitude and longitude in degrees.minutes, and
* magnetic field (in nT) ; these data will be put into a data file
* named ORNAVDM.DAT.

* You are allowed to stop introducing the data, and to resume the data
* input during another session. Before the translation of this data you
* will be able to correct them using the editor.

* As the translated data will be corrected using the position of points
* points of the survey from Seabeam maps, and will be used to compute
* the real magnetic anomaly, the translated file (ORNAVDD.DAT) will have
* decimal year (year+day+time), and latitudes and longitudes in degrees.

*
*  PREPARATION
*  -----------
*
      INTEGER N,I,YEAR,MONTH,DAY,TIME,ANSWER,MONLEN(11),NDAY,N1,PAGE,MAG
      REAL LATDM,LONGDM,HOUR,HR,DDEC,LATDD,LONGDD
      REAL*8 YDEC
      CHARACTER R,TITLE*40,TIT(40),ESC,SO,SI,FF

      EQUIVALENCE (TITLE,TIT)

      DATA MONLEN /31,28,31,30,31,30,2*31,30,31,30/

      OPEN (1, FILE='ORNAVDM.DAT')
      OPEN (2, FILE='ORNAVDD.DAT', STATUS='NEW')
      OPEN (3, FILE='PRN')

*
*  FORMATS
*  -------
* 1 : to write and read the original data file
* 2 : to write and read the translated data file
* 3 : to write the original and translated data on the printer
*
   1  FORMAT (1X,3(I2,1X),I4,1X,F7.3,1X,F8.3,1X,I5)
   2  FORMAT (1X,F10.7,1X,F9.5,1X,F10.5,1X,I5)
   3  FORMAT (1X,I4,4X,I2,5X,I2,4X,I2,2X,I4,2X,F7.3,2X,F8.3,8X,F10.7,2X,
     +F9.5,2X,F10.5,2X,I5)
*
*  PRESENTATION
*  ------------
*
      WRITE (*,'(A//)') '                              PROGRAM ORIGNAV'
      WRITE (*,'(A)') ' This program has been written to put the navigat
     +ion of Scripps'
      WRITE (*,'(A)') ' Seabeam maps and the recorded magnetic field int
     +o a sequential data'
      WRITE (*,'(A)') ' file and after the input of these data to transl
     +ate them into a'
      WRITE (*,'(A)') ' format easier for computational use.'
      WRITE (*,'(A)') ' The original data are : year (YY), month (MM), d
     +ay (DD), time (input'
      WRITE (*,'(A)') ' in HHMM format), latitude and longitude in degre
     +es.minutes, and'
      WRITE (*,'(A)') ' magnetic field (in nT) ; these data will be put
     +into a data file'
      WRITE (*,'(A/)')' named ORNAVDM.DAT.'
      WRITE (*,'(A)') ' You are allowed to stop introducing the data, an
     +d to resume the data'
      WRITE (*,'(A)') ' input during another session. Before the transla
     +tion of this data'
      WRITE (*,'(A/)') ' you will be able to correct them using the edit
     +or.'
      WRITE (*,'(A)') ' As the translated data will be corrected using t
     +he position of other'
      WRITE (*,'(A)') ' points of the survey from Seabeam maps, and will
     + be used to compute'
      WRITE (*,'(A)')' the real magnetic anomaly, the translated file (
     +ORNAVDD.DAT) will have'
      WRITE (*,'(A//)') ' decimal year (year+day+time) and latitudes and
     + longitudes in degrees.'

*
*  PART CHOICE : INPUT OF ORIGINAL DATA OR TRANSLATION
*  ---------------------------------------------------
*
   4  WRITE (*,'(A\)') ' Do you want to input original data (1) or to tr
     +anslate them (2) ? '
      READ (*,'(I1)') ANSWER
          IF ((ANSWER .NE. 1) .AND. (ANSWER .NE. 2)) GOTO 4
          IF (ANSWER .EQ. 2) GOTO 8

*
*  ORIGINAL DATA INPUT PART
*  ------------------------
*
      WRITE (*,'(/A/)') ' Now you will input your data, for the last one
     +, type 0 for the YEAR'
      WRITE (*,'(A\)') ' What is the number of the first point to be inp
     +ut : '
      READ (*,'(I5)') N
          IF (N .EQ. 1) GOTO 6
*
* Reading the beginning of the file
*
```

```
      DO 5 J=1,N-1
   5     READ (1,1) YEAR,MONTH,DAY,TIME,LATDM,LONGDM,MAG
*
* Writing the data
*
   6  WRITE (*,'(/A,I5)') '  POINT ', N
      WRITE (*,'(A\)') ' Year (YY) ? '
      READ (*,'(I2)') YEAR
          IF (YEAR .EQ. 0) GOTO 7
      WRITE (*,'(A\)') ' Month (MM) ? '
      READ (*,'(I2)') MONTH
      WRITE (*,'(A\)') ' Day (DD) ? '
      READ (*,'(I2)') DAY
      WRITE (*,'(A\)') ' Time (HHMM) ? '
      READ (*,'(I4)') TIME
      WRITE (*,'(A\)') ' Latitude (DD.MMm) ? '
      READ (*,'(F7.3)') LATDM
      WRITE (*,'(A\)') ' Longitude (DDD.MMm) ? '
      READ (*,'(F8.3)') LONGDM
      WRITE (*,'(A\)') ' Magnetic field in nT (XXXXX) ? '
      READ (*,'(I5)') MAG
      WRITE (1,1) YEAR, MONTH, DAY, TIME, LATDM, LONGDM, MAG
      N=N+1
      GOTO 6

   7  WRITE (*,'(A\)') ' Is it the last point of the whole file (Y/N) ?
     + '
      READ (*,'(A)') R
          IF ((R .NE. 'Y') .AND. (R .NE. 'y')) GOTO 20
      ENDFILE 1

      WRITE (*,'(A\)') ' Do you want to translate the original data file
     + immediatly (Y/N) ? '
      READ (*,'(A)') R
          IF ((R .NE. 'Y') .AND. (R .NE. 'y')) GOTO 20
      REWIND 1

*
*  TRANSLATION PART
*  ----------------
*
   8  WRITE (*,'(A)') ' Title of the study (up to 40 characters) ? '
      READ (*,'(A)') TITLE

   9  READ (1,1,END=11) YEAR,MONTH,DAY,TIME,LATDM,LONGDM,MAG
*
* Computing the decimal year
*
      NDAY=0
      HOUR=FLOAT(TIME)/100
      HR=FLOAT(IFIX(HOUR))
      DDEC=(HR+(HOUR-HR)*100/60)/24
      DO 10 I=1,MONTH-1
  10      NDAY=MONLEN(I)+NDAY
      YDEC=YEAR+(NDAY+(DAY-1)+DDEC)/365
*
* Computing the latitudes and longitudes in decimal degrees
*
      LATDD=IFIX(LATDM)+(LATDM-IFIX(LATDM))*100/60
      LONGDD=IFIX(LONGDM)+(LONGDM-IFIX(LONGDM))*100/60
*
* Writing the data into the file
*
      WRITE (2,2) YDEC, LATDD, LONGDD, MAG
      GOTO 9
*
* Preparing the printing part
*
  11  ENDFILE 2
      REWIND 1
      REWIND 2
*
* Preparing the title
*
      DO 12  I = 40, 1, -1
           IF (TIT(I) .NE. ' ') GOTO 13
  12  CONTINUE
*
*  PRINTING PART
*  -------------
*
  13  WRITE (*,'(//A\)') ' PRINTER READY (Y/N) ? '
      READ (*,'(A)') R
          IF ((R .NE. 'Y') .AND. (R .NE. 'y')) GOTO 20
*
* Control characters for the printer
*
      ESC=CHAR(27)
      SO=CHAR(14)
      SI=CHAR(15)
      FF=CHAR(12)
*
* Printing the title and comments
*
      WRITE (3,'(1X,9A)') ESC,'q',ESC,'L015',ESC,'E',ESC,'!',SO
      N1=INT((40-I)/2)+1
      DO 14  N = 1, N1
  14     WRITE (3,'(A\)') ' '
      WRITE (3,'(2A\)') ESC,'X'
      DO 15  N = 1, I
  15     WRITE (3,'(A\)') TIT(N)
      WRITE (3,'(2A)') ESC,'Y'
      WRITE (3,'(1X,3A////)') SI,ESC,'N'
      WRITE (3,'(1X,A/)') 'ORIGINAL NAVIGATION DATA'
      WRITE (3,'(1X,4A)') ESC,'*',ESC,'q'
*
* Printing the column headings and data
*
      I=1
      PAGE=1
  16  WRITE (3,'(1X,A)') 'POINT  YEAR  MONTH  DAY  TIME     LAT       LON
     +G            TIME          LAT         LONG      MAG'
      WRITE (3,'(1X,A/)') ' no.     yy      mm     dd  hhmm    dd.mmm    ddd.
     +mmm          dec.year      dec.deg     dec.deg     nT'

  17  READ (1,1,END=19) YEAR,MONTH,DAY,TIME,LATDM,LONGDM,MAG
```

```
      READ (2,2) YDEC,LATDD,LONGDD,MAG
      WRITE (3,3) I,YEAR,MONTH,DAY,TIME,LATDM,LONGDM,YDEC,LATDD,LONGDD,M
     +AG
      I=I+1
          IF ((PAGE .EQ. 1) .AND. (I .LT. 52)) GOTO 17
          IF ((PAGE .EQ. 1) .AND. (I .EQ. 52)) GOTO 18
          IF (I .NE. (52+(PAGE-1)*61)) GOTO 17
*
*  Page changing
*
  18  PAGE=PAGE+1
      WRITE (3,'(1X,A)') FF
      GOTO 16

*
*  END OF THE PROGRAM
*  ------------------
*
  19  WRITE (3,'(/)')
  20  CLOSE (1)
      CLOSE (2)
      END
```

B
★

PROGRAM ORIGNAV

This program has been written to put the navigation of Scripps Seabeam maps and the recorded magnetic field into a sequential data file and after the input of these data to translate them into a format easier for computational use.

The original data are : year (YY), month (MM), day (DD), time (input in HHMM format), latitude and longitude in degrees.minutes, and magnetic field (in nT) ; these data will be put into a data file named ORNAVDN.DAT.

You are allowed to stop introducing the data, and to resume the data input during another session. Before the translation of this data you will be able to correct them using the editor.

As the translated data will be corrected using the position of other points of the survey from Seabeam maps, and will be used to compute the real magnetic anomaly, the translated file (ORNAVDD.DAT) will have decimal year (year+day+time) and latitudes and longitudes in degrees.

Do you want to input original data (1) or to translate them (2) ? 1

Now you will input your data, for the last one, type 0 for the YEAR

What is the number of the first point to be input : 1

```
 POINT     1
Year (YY) ? 82
Month (MM) ? 4
Day (DD) ? 7
Time (HHMM) ? 815
Latitude (DD.MMm) ? 9.226
Longitude (DDD.MMm) ? -85.004
Magnetic field in nT (XXXXX) ? 37200

 POINT     2
Year (YY) ? 82
Month (MM) ? 4
Day (DD) ? 7
Time (HHMM) ? 820
Latitude (DD.MMm) ? 9.220
Longitude (DDD.MMm) ? -85.008
Magnetic field in nT (XXXXX) ? 37200

 POINT     3
Year (YY) ? 82
Month (MM) ? 4
Day (DD) ? 7
Time (HHMM) ? 825
Latitude (DD.MMm) ? 9.213
Longitude (DDD.MMm) ? -85.013
Magnetic field in nT (XXXXX) ? 37194
```

```
 POINT     4
Year (YY) ? 82
Month (MM) ? 4
Day (DD) ? 7
Time (HHMM) ? 830
Latitude (DD.MMm) ? 9.207
Longitude (DDD.MMm) ? -85.017
Magnetic field in nT (XXXXX) ? 37183

 POINT     5
Year (YY) ? 82
Month (MM) ? 4
Day (DD) ? 7
Time (HHMM) ? 835
Latitude (DD.MMm) ? 9.2
Longitude (DDD.MMm) ? -85.022
Magnetic field in nT (XXXXX) ? 37169

 POINT     6
Year (YY) ? 82
Month (MM) ? 4
Day (DD) ? 7
Time (HHMM) ? 840
Latitude (DD.MMm) ? 9.193
Longitude (DDD.MMm) ? -85.026
Magnetic field in nT (XXXXX) ? 37154

 POINT     7
Year (YY) ? 82
Month (MM) ? 4
Day (DD) ? 7
Time (HHMM) ? 845
Latitude (DD.MMm) ? 9.187
Longitude (DDD.MMm) ? -85.031
Magnetic field in nT (XXXXX) ? 37145

 POINT     8
Year (YY) ? 82
Month (MM) ? 4
Day (DD) ? 7
Time (HHMM) ? 851
Latitude (DD.MMm) ? 3.179
Longitude (DDD.MMm) ? -85.036
Magnetic field in nT (XXXXX) ? 37135

 POINT     9
Year (YY) ? 82
Month (MM) ? 4
Day (DD) ? 7
Time (HHMM) ? 855
Latitude (DD.MMm) ? 9.174
Longitude (DDD.MMm) ? -85.04
Magnetic field in nT (XXXXX) ? 37124
```

.
.
.
.
.

```
 POINT   620
Year (YY) ? 82
Month (MM) ? 4
Day (DD) ? 9
Time (HHMM) ? 1250
Latitude (DD.MMm) ? 9.177
Longitude (DDD.MMm) ? -85.482
Magnetic field in nT (XXXXX) ? 37021

 POINT   621
Year (YY) ? 82
Month (MM) ? 4
Day (DD) ? 9
Time (HHMM) ? 1255
Latitude (DD.MMm) ? 9.185
Longitude (DDD.MMm) ? -85.486
Magnetic field in nT (XXXXX) ? 37064

 POINT   622
Year (YY) ? 82
Month (MM) ? 4
Day (DD) ? 9
Time (HHMM) ? 1300
Latitude (DD.MMm) ? 9.193
Longitude (DDD.MMm) ? -85.489
Magnetic field in nT (XXXXX) ? 37102

 POINT   623
Year (YY) ? 82
Month (MM) ? 4
Day (DD) ? 9
Time (HHMM) ? 1305
Latitude (DD.MMm) ? 9.2
Longitude (DDD.MMm) ? -85.491
Magnetic field in nT (XXXXX) ? 37126

 POINT   624
Year (YY) ? 82
Month (MM) ? 4
Day (DD) ? 9
Time (HHMM) ? 1310
Latitude (DD.MMm) ? 9.208
Longitude (DDD.MMm) ? -85.495
Magnetic field in nT (XXXXX) ? 37140
```

```
 POINT   625
Year (YY) ? 82
Month (MM) ? 4
Day (DD) ? 9
Time (HHMM) ? 1315
Latitude (DD.MMm) ? 9.216
Longitude (DDD.MMm) ? -85.498
Magnetic field in nT (XXXXX) ? 37144

 POINT   626
Year (YY) ? 82
Month (MM) ? 4
Day (DD) ? 9
Time (HHMM) ? 1321
Latitude (DD.MMm) ? 9.225
Longitude (DDD.MMm) ? -85.501
Magnetic field in nT (XXXXX) ? 37137

 POINT   627
Year (YY) ? 82
Month (MM) ? 4
Day (DD) ? 9
Time (HHMM) ? 1325
Latitude (DD.MMm) ? 9.231
Longitude (DDD.MMm) ? -85.502
Magnetic field in nT (XXXXX) ? 37135

 POINT   628
Year (YY) ? 82
Month (MM) ? 4
Day (DD) ? 9
Time (HHMM) ? 1330
Latitude (DD.MMm) ? 9.239
Longitude (DDD.MMm) ? -85.504
Magnetic field in nT (XXXXX) ? 37149

 POINT   629
Year (YY) ? 0
Is it the last point of the whole file (Y/N) ? Y
Do you want to translate the original data file immediatly (Y/N) ? Y
Title of the study (up to 40 characters) ?
COSTA RICA I SURVEY

PRINTER READY (Y/N) ? Y
```

C

COSTA RICA I SURVEY

ORIGINAL NAVIGATION DATA

POINT no.	YEAR yy	MONTH mm	DAY dd	TIME hhmm	LAT dd.mmm	LONG ddd.mmm	TIME dec.year	LAT dec.deg	LONG dec.deg	MAG nT
1	82	4	7	815	9.226	-85.004	82.2639555	9.37667	-85.00666	37200
2	82	4	7	820	9.220	-85.008	82.2639650	9.36667	-85.01334	37200
3	82	4	7	825	9.213	-85.013	82.2639745	9.35500	-85.02167	37194
4	82	4	7	830	9.207	-85.017	82.2639840	9.34500	-85.02833	37183
5	82	4	7	835	9.200	-85.022	82.2639935	9.33333	-85.03667	37169
6	82	4	7	840	9.193	-85.026	82.2640030	9.32167	-85.04333	37154
7	82	4	7	845	9.187	-85.031	82.2640126	9.31167	-85.05167	37145
8	82	4	7	851	9.179	-85.036	82.2640240	9.29833	-85.06001	37135
9	82	4	7	855	9.174	-85.040	82.2640316	9.29000	-85.06667	37124
10	82	4	7	900	9.167	-85.044	82.2640411	9.27833	-85.07333	37106
11	82	4	7	906	9.160	-85.049	82.2640525	9.26667	-85.08167	37083
12	82	4	7	911	9.154	-85.054	82.2640620	9.25667	-85.09000	37064
13	82	4	7	916	9.147	-85.058	82.2640715	9.24500	-85.09666	37042
14	82	4	7	920	9.142	-85.062	82.2640791	9.23667	-85.10332	37025
15	82	4	7	926	9.135	-85.067	82.2640906	9.22500	-85.11167	37019
16	82	4	7	930	9.129	-85.071	82.2640982	9.21500	-85.11833	37024
17	82	4	7	935	9.124	-85.075	82.2641077	9.20667	-85.12499	37034
18	82	4	7	940	9.119	-85.078	82.2641172	9.19833	-85.13000	37040
19	82	4	7	945	9.113	-85.082	82.2641267	9.18833	-85.13667	37049
20	82	4	7	950	9.106	-85.087	82.2641362	9.17667	-85.14500	37063
21	82	4	7	955	9.099	-85.091	82.2641457	9.16500	-85.15167	37078
22	82	4	7	1000	9.092	-85.096	82.2641553	9.15333	-85.16000	37094
23	82	4	7	1005	9.086	-85.100	82.2641648	9.14333	-85.16666	37103
24	82	4	7	1010	9.079	-85.104	82.2641743	9.13167	-85.17332	37099
25	82	4	7	1015	9.072	-85.109	82.2641838	9.12000	-85.18167	37081
26	82	4	7	1020	9.066	-85.113	82.2641933	9.11000	-85.18833	37051
27	82	4	7	1025	9.060	-85.117	82.2642028	9.10000	-85.19499	37018
28	82	4	7	1030	9.053	-85.121	82.2642123	9.08833	-85.20167	36990
29	82	4	7	1035	9.047	-85.126	82.2642218	9.07833	-85.21000	36964
30	82	4	7	1040	9.040	-85.130	82.2642314	9.06667	-85.21666	36940
31	82	4	7	1045	9.034	-85.134	82.2642409	9.05667	-85.22334	36918
32	82	4	7	1050	9.027	-85.138	82.2642504	9.04500	-85.23000	36898
33	82	4	7	1055	9.021	-85.143	82.2642599	9.03500	-85.23833	36878
34	82	4	7	1100	9.015	-85.147	82.2642694	9.02500	-85.24500	36860
35	82	4	7	1105	9.008	-85.151	82.2642789	9.01333	-85.25167	36849
36	82	4	7	1110	9.002	-85.156	82.2642884	9.00333	-85.25999	36845
37	82	4	7	1115	8.595	-85.160	82.2642979	8.99167	-85.26667	36846
38	82	4	7	1120	8.588	-85.165	82.2643075	8.98000	-85.27500	36843
39	82	4	7	1125	8.582	-85.169	82.2643170	8.97000	-85.28166	36838
40	82	4	7	1130	8.575	-85.174	82.2643265	8.95833	-85.29001	36831
41	82	4	7	1135	8.568	-85.178	82.2643360	8.94667	-85.29667	36824
42	82	4	7	1140	8.562	-85.183	82.2643455	8.93667	-85.30500	36816
43	82	4	7	1145	8.555	-85.187	82.2643550	8.92500	-85.31166	36808
44	82	4	7	1150	8.549	-85.192	82.2643645	8.91500	-85.32000	36805
45	82	4	7	1155	8.543	-85.196	82.2643740	8.90500	-85.32667	36806
46	82	4	7	1250	8.545	-85.207	82.2644787	8.90833	-85.34500	36785
47	82	4	7	1255	8.551	-85.202	82.2644882	8.91833	-85.33667	36778
48	82	4	7	1300	8.557	-85.198	82.2644977	8.92833	-85.32999	36779
49	82	4	7	1305	8.563	-85.193	82.2645072	8.93833	-85.32167	36787
50	82	4	7	1310	8.569	-85.189	82.2645167	8.94833	-85.31500	36799
51	82	4	7	1315	8.575	-85.184	82.2645263	8.95833	-85.30666	36810

POINT no.	YEAR yy	MONTH mm	DAY dd	TIME hhmm	LAT dd.mmm	LONG ddd.mmm	TIME dec.year	LAT dec.deg	LONG dec.deg	MAG nT
52	82	4	7	1321	8.582	-85.179	82.2645377	8.97000	-85.29833	36822
53	82	4	7	1326	8.588	-85.174	82.2645472	8.98000	-85.29001	36831
54	82	4	7	1330	8.593	-85.171	82.2645548	8.98833	-85.28500	36839
55	82	4	7	1336	9.000	-85.166	82.2645662	9.00000	-85.27666	36846
56	82	4	7	1341	9.007	-85.162	82.2645757	9.01167	-85.27000	36853
57	82	4	7	1346	9.014	-85.159	82.2645852	9.02333	-85.26499	36863
58	82	4	7	1351	9.021	-85.156	82.2645947	9.03500	-85.25999	36879
59	82	4	7	1355	9.027	-85.153	82.2646024	9.04500	-85.25500	36902
60	82	4	7	1401	9.035	-85.149	82.2646138	9.05833	-85.24834	36925
61	82	4	7	1405	9.041	-85.147	82.2646214	9.06833	-85.24500	36951
62	82	4	7	1410	9.047	-85.144	82.2646309	9.07833	-85.24000	36979
63	82	4	7	1415	9.054	-85.141	82.2646404	9.09000	-85.23500	37003
64	82	4	7	1420	9.060	-85.138	82.2646499	9.10000	-85.23000	37026
65	82	4	7	1425	9.067	-85.135	82.2646594	9.11167	-85.22501	37054
66	82	4	7	1430	9.073	-85.131	82.2646689	9.12167	-85.21833	37088
67	82	4	7	1435	9.080	-85.128	82.2646785	9.13333	-85.21333	37121
68	82	4	7	1440	9.087	-85.125	82.2646880	9.14500	-85.20834	37148
69	82	4	7	1445	9.093	-85.122	82.2646975	9.15500	-85.20334	37158
70	82	4	7	1450	9.100	-85.119	82.2647070	9.16667	-85.19834	37147
71	82	4	7	1455	9.105	-85.124	82.2647165	9.17500	-85.20667	37135
72	82	4	7	1500	9.107	-85.131	82.2647260	9.17833	-85.21833	37133
73	82	4	7	1505	9.100	-85.134	82.2647355	9.16667	-85.22334	37162
74	82	4	7	1510	9.093	-85.138	82.2647451	9.15500	-85.23000	37172
75	82	4	7	1515	9.086	-85.142	82.2647546	9.14333	-85.23666	37163
76	82	4	7	1520	9.079	-85.145	82.2647641	9.13167	-85.24166	37146
77	82	4	7	1525	9.072	-85.149	82.2647736	9.12000	-85.24834	37131
78	82	4	7	1530	9.065	-85.153	82.2647831	9.10833	-85.25500	37116
79	82	4	7	1535	9.058	-85.157	82.2647926	9.09667	-85.26167	37098
80	82	4	7	1540	9.051	-85.162	82.2648021	9.08500	-85.27000	37070
81	82	4	7	1545	9.044	-85.167	82.2648116	9.07333	-85.27834	37040
82	82	4	7	1550	9.038	-85.171	82.2648212	9.06333	-85.28500	37002
83	82	4	7	1555	9.031	-85.175	82.2648307	9.05167	-85.29167	36975
84	82	4	7	1600	9.024	-85.180	82.2648402	9.04000	-85.30000	36955
85	82	4	7	1605	9.018	-85.184	82.2648497	9.03000	-85.30666	36938
86	82	4	7	1610	9.011	-85.188	82.2648592	9.01833	-85.31334	36921
87	82	4	7	1615	9.005	-85.192	82.2648687	9.00833	-85.32000	36901
88	82	4	7	1620	8.599	-85.197	82.2648782	8.99833	-85.32833	36881
89	82	4	7	1625	8.592	-85.201	82.2648877	8.98667	-85.33499	36862
90	82	4	7	1630	8.585	-85.206	82.2648973	8.97500	-85.34334	36846
91	82	4	7	1635	8.579	-85.210	82.2649068	8.96500	-85.35000	36830
92	82	4	7	1640	8.572	-85.215	82.2649163	8.95333	-85.35833	36815
93	82	4	7	1645	8.565	-85.219	82.2649258	8.94167	-85.36501	36804
94	82	4	7	1650	8.568	-85.225	82.2649353	8.94667	-85.37500	36782
95	82	4	7	1655	8.572	-85.232	82.2649448	8.95333	-85.38667	36766
96	82	4	7	1700	8.576	-85.238	82.2649543	8.96000	-85.39667	36758
97	82	4	7	1705	8.579	-85.233	82.2649639	8.96500	-85.38834	36759
98	82	4	7	1710	8.584	-85.228	82.2649734	8.97333	-85.38000	36779
99	82	4	7	1715	8.590	-85.224	82.2649829	8.98333	-85.37333	36800
100	82	4	7	1720	8.596	-85.219	82.2649924	8.99333	-85.36501	36825

POINT no.	YEAR yy	MONTH mm	DAY dd	TIME hhmm	LAT dd.mmm	LONG ddd.mmm	TIME dec.year	LAT dec.deg	LONG dec.deg	MAG nT
540	82	4	9	610	9.114	-85.279	82.2691971	9.19000	-85.46500	36871
541	82	4	9	615	9.107	-85.279	82.2692066	9.17833	-85.46500	36908
542	82	4	9	620	9.099	-85.279	82.2692161	9.16500	-85.46500	36959
543	82	4	9	625	9.091	-85.278	82.2692256	9.15167	-85.46333	37012
544	82	4	9	630	9.083	-85.278	82.2692352	9.13833	-85.46333	37055
545	82	4	9	635	9.075	-85.277	82.2692447	9.12500	-85.46167	37082
546	82	4	9	640	9.067	-85.276	82.2692542	9.11167	-85.46000	37100
547	82	4	9	645	9.059	-85.275	82.2692637	9.09833	-85.45834	37091
548	82	4	9	651	9.050	-85.275	82.2692751	9.08333	-85.45834	37043
549	82	4	9	655	9.043	-85.274	82.2692827	9.07167	-85.45667	37013
550	82	4	9	700	9.035	-85.274	82.2692922	9.05833	-85.45667	36998
551	82	4	9	705	9.027	-85.273	82.2693018	9.04500	-85.45500	36949
552	82	4	9	710	9.026	-85.280	82.2693113	9.04333	-85.46667	36946
553	82	4	9	715	9.026	-85.287	82.2693208	9.04333	-85.47834	36988
554	82	4	9	720	9.027	-85.294	82.2693303	9.04500	-85.49000	37046
555	82	4	9	725	9.027	-85.301	82.2693398	9.04500	-85.50167	37099
556	82	4	9	730	9.027	-85.308	82.2693493	9.04500	-85.51333	37121
557	82	4	9	735	9.027	-85.315	82.2693588	9.04500	-85.52500	37123
558	82	4	9	740	9.027	-85.323	82.2693683	9.04500	-85.53833	37123
559	82	4	9	745	9.028	-85.330	82.2693779	9.04667	-85.55000	37125
560	82	4	9	750	9.033	-85.333	82.2693874	9.05500	-85.55500	37133
561	82	4	9	755	9.040	-85.332	82.2693969	9.06667	-85.55334	37136
562	82	4	9	800	9.048	-85.331	82.2694064	9.08000	-85.55167	37130
563	82	4	9	805	9.055	-85.330	82.2694159	9.09167	-85.55000	37111
564	82	4	9	810	9.063	-85.329	82.2694254	9.10500	-85.54834	37072
565	82	4	9	815	9.071	-85.327	82.2694349	9.11833	-85.54501	37022
566	82	4	9	820	9.078	-85.326	82.2694444	9.13000	-85.54333	36962
567	82	4	9	825	9.086	-85.325	82.2694540	9.14333	-85.54166	36912
568	82	4	9	830	9.094	-85.324	82.2694635	9.15667	-85.53999	36865
569	82	4	9	835	9.101	-85.322	82.2694730	9.16833	-85.53667	36832
570	82	4	9	840	9.108	-85.322	82.2694825	9.18000	-85.53667	36816
571	82	4	9	845	9.103	-85.327	82.2694920	9.17167	-85.54501	36817
572	82	4	9	850	9.097	-85.332	82.2695015	9.16167	-85.55334	36818
573	82	4	9	855	9.092	-85.337	82.2695110	9.15333	-85.56166	36822
574	82	4	9	901	9.085	-85.342	82.2695225	9.14167	-85.57001	36833
575	82	4	9	906	9.079	-85.347	82.2695320	9.13167	-85.57833	36847
576	82	4	9	910	9.075	-85.351	82.2695396	9.12500	-85.58500	36864
577	82	4	9	915	9.069	-85.356	82.2695491	9.11500	-85.59334	36884
578	82	4	9	920	9.063	-85.361	82.2695586	9.10500	-85.60167	36904
579	82	4	9	925	9.059	-85.364	82.2695681	9.09833	-85.60667	36923
580	82	4	9	930	9.054	-85.368	82.2695776	9.09000	-85.61333	36941
581	82	4	9	935	9.048	-85.373	82.2695871	9.08000	-85.62167	36957
582	82	4	9	940	9.043	-85.377	82.2695967	9.07167	-85.62833	36975
583	82	4	9	945	9.037	-85.382	82.2696062	9.06167	-85.63667	36992
584	82	4	9	950	9.032	-85.386	82.2696157	9.05333	-85.64333	37010
585	82	4	9	955	9.029	-85.390	82.2696252	9.04833	-85.65000	37027
586	82	4	9	1000	9.036	-85.392	82.2696347	9.06000	-85.65333	37010
587	82	4	9	1005	9.043	-85.394	82.2696442	9.07167	-85.65666	36918
588	82	4	9	1010	9.050	-85.396	82.2696537	9.08333	-85.66000	36832
589	82	4	9	1015	9.057	-85.398	82.2696632	9.09500	-85.66334	36768
590	82	4	9	1020	9.064	-85.400	82.2696728	9.10667	-85.66667	36727
591	82	4	9	1025	9.071	-85.402	82.2696823	9.11833	-85.67000	36699
592	82	4	9	1030	9.078	-85.404	82.2696918	9.13000	-85.67333	36686
593	82	4	9	1035	9.085	-85.406	82.2697013	9.14167	-85.67667	36696
594	82	4	9	1040	9.092	-85.408	82.2697108	9.15333	-85.67999	36726
595	82	4	9	1045	9.099	-85.410	82.2697203	9.16500	-85.68334	36762
596	82	4	9	1050	9.105	-85.412	82.2697298	9.17500	-85.68667	36798
597	82	4	9	1055	9.113	-85.414	82.2697393	9.18833	-85.69000	36834
598	82	4	9	1100	9.119	-85.416	82.2697489	9.19833	-85.69334	36866
599	82	4	9	1105	9.126	-85.418	82.2697584	9.21000	-85.69666	36897
600	82	4	9	1110	9.133	-85.420	82.2697679	9.22167	-85.70000	36928

POINT no.	YEAR yy	MONTH mm	DAY dd	TIME hhmm	LAT dd.mmm	LONG ddd.mmm	TIME dec.year	LAT dec.deg	LONG dec.deg	MAG nT
601	82	4	9	1115	9.140	-85.422	82.2697774	9.23333	-85.70333	36955
602	82	4	9	1120	9.147	-85.424	82.2697869	9.24500	-85.70667	36976
603	82	4	9	1125	9.154	-85.426	82.2697964	9.25667	-85.71001	36994
604	82	4	9	1130	9.161	-85.428	82.2698059	9.26833	-85.71333	37010
605	82	4	9	1135	9.168	-85.430	82.2698154	9.28000	-85.71667	37021
606	82	4	9	1140	9.176	-85.433	82.2698250	9.29333	-85.72166	37029
607	82	4	9	1145	9.183	-85.436	82.2698345	9.30500	-85.72666	37030
608	82	4	9	1150	9.186	-85.440	82.2698440	9.31000	-85.73334	37030
609	82	4	9	1155	9.179	-85.444	82.2698535	9.29833	-85.74000	37027
610	82	4	9	1200	9.172	-85.448	82.2698630	9.28667	-85.74667	37023
611	82	4	9	1205	9.166	-85.452	82.2698725	9.27667	-85.75334	37018
612	82	4	9	1210	9.159	-85.457	82.2698820	9.26500	-85.76167	37021
613	82	4	9	1215	9.152	-85.461	82.2698916	9.25333	-85.76833	37026
614	82	4	9	1220	9.146	-85.465	82.2699011	9.24333	-85.77499	37029
615	82	4	9	1225	9.143	-85.468	82.2699106	9.23833	-85.78001	37029
616	82	4	9	1230	9.146	-85.470	82.2699201	9.24333	-85.78333	37020
617	82	4	9	1235	9.153	-85.473	82.2699296	9.25500	-85.78833	37009
618	82	4	9	1240	9.161	-85.476	82.2699391	9.26833	-85.79333	36993
619	82	4	9	1245	9.169	-85.479	82.2699486	9.28167	-85.79832	36993
620	82	4	9	1250	9.177	-85.482	82.2699581	9.29500	-85.80334	37021
621	82	4	9	1255	9.185	-85.486	82.2699677	9.30833	-85.81000	37064
622	82	4	9	1300	9.193	-85.489	82.2699772	9.32167	-85.81499	37102
623	82	4	9	1305	9.200	-85.491	82.2699867	9.33333	-85.81833	37126
624	82	4	9	1310	9.208	-85.495	82.2699962	9.34667	-85.82500	37140
625	82	4	9	1315	9.216	-85.498	82.2700057	9.36000	-85.83000	37144
626	82	4	9	1321	9.225	-85.501	82.2700171	9.37500	-85.83500	37137
627	82	4	9	1325	9.231	-85.502	82.2700247	9.38500	-85.83666	37135
628	82	4	9	1330	9.239	-85.504	82.2700342	9.39833	-85.84000	37149

1.3

A

```
*-----------------------------------------------------------------------*
*                                                                       *
*                          PROGRAM 'NAVMAGCR'                           *
*                                                                       *
*                                        Jacqueline ROUMP, Jan. 1985    *
*-----------------------------------------------------------------------*

      PROGRAM NAVMAGCR

*  This program has been written to compute the position of any point of
*  the ship track of a Seabeam map using the corrections needed for
*  Scripps Seabeam post-processing. This corrected navigation is used to
*  compute the magnetic anomaly of each point from the original data.
*  The synthetic magnetic field is for the period 1975-1985.
*
*  This program used three files : SBNAVDD.DAT created by the program
*  CBEAMNAV, ORNAVDM.DAT and ORNAVDD.DAT created by the program
*  ORIGNAV. The program will also create a file named NAVMAGCR.DAT.
*
*  PREPARATION
*  -----------
*
      INTEGER ORMAG,PHT,ANOM,PAGE,YEAR,MONTH,DAY,TIME,RMAG
      REAL ORLAT,ORLONG,PI,CRLAT,CRLONG
      REAL*8 SBYER1,SBYER2,ORYEAR,DTIME,ORSBDY,SBDYER
      CHARACTER TIT(40),BEEP,RS,ESC,SO,SI,FF,TITLE*40

      EQUIVALENCE (TITLE,TIT)
      COMMON PI,A2,B2


      OPEN (1, FILE='SBNAVDD.DAT')
      OPEN (2, FILE='ORNAVDM.DAT')
      OPEN (3, FILE='ORNAVDD.DAT')
      OPEN (4, FILE='NAVMAGCR.DAT', STATUS='NEW')
      OPEN (5, FILE='PRN')

*
*  FORMATS
*  -------
*  1 : to read the data from the decimal degrees Seabeam data file
*  2 : to read the data from the original deg.min nav. & mag. data file
*  3 : to read the data from the original decim. deg. nav. & mag. data file
*  4 : to write the corrected navigation and magnetic anomaly in a file
*  5 : to write the corrected navigation and magnetic anomaly on the printer

   1  FORMAT (1X,F10.7,2(1X,F8.5))
   2  FORMAT (1X,3(I2,1X),I4,1X,F7.3,1X,F8.3,1X,I5)
   3  FORMAT (1X,F10.7,1X,F9.5,1X,F10.5,1X,I5)
   4  FORMAT (1X,3(I2,1X),I4,1X,F9.5,1X,F10.5,3(1X,I5))
   5  FORMAT (1X,I4,4X,I2,5X,I2,4X,I2,2X,I4,2X,F7.3,2X,F8.3,3(4X,I5))

*
*  PRESENTATION
*  ------------
*
      WRITE (*,'(A)') ' This program has been written to compute the pos
     +ition of any point of'
      WRITE (*,'(A)') ' the ship track of a Seabeam map using the correc
     +tions needed for'
      WRITE (*,'(A)') ' Scripps Seabeam post-processing. This corrected
     +navigation is used to'
      WRITE (*,'(A)') ' compute the magnetic anomaly of each point from
     +the original data.'
      WRITE (*,'(A/)') ' The synthetic magnetic field is for the period
     +1975-1985.'
      WRITE (*,'(A)') ' This program used three files : SBNAVDD.DAT crea
     +ted by the program'
      WRITE (*,'(A)') ' CBEAMNAV, ORNAVDM.DAT and ORNAVDD.DAT created by
     + the program '
      WRITE (*,'(A/)') ' ORIGNAV. The program will also create a file na
     +med NAVMAGCR.DAT.'

*
*  QUESTIONS
*  ---------
*
      WRITE (*,'(A)') ' Title of the table (up to 40 characters) : '
      READ (*,'(A)') TITLE
      WRITE (*,'(//)')

*
*  COMPUTATION PART
*  ----------------
*
      PI=3.14159265
      A2=6378.135**2
      B2=6356.75**2
      READ (1,1) SBYER1,SBNS1,SBEW1
      READ (1,1) SBYER2,SBNS2,SBEW2
   6  READ (3,3,END=8) ORYEAR,ORLAT,ORLONG,ORMAG
      READ (2,2) YEAR,MONTH,DAY,TIME,RLAT,RLONG,RMAG
   7    IF (ORYEAR .GT. SBYER2) THEN
           NX=NX+1
           SBYER1=SBYER2
           SBNS1=SBNS2
           SBEW1=SBEW2
           READ (1,1) SBYER2,SBNS2,SBEW2
           GOTO 7
        ELSE
           DTIME=ORYEAR+1900.-1980.
           ORSBDY=ORYEAR-SBYER1
           SBDYER=SBYER2-SBYER1
           CRLAT=ORLAT+SBNS1+(SBNS2-SBNS1)*(ORSBDY/SBDYER)
           CRLONG=ORLONG+SBEW1+(SBEW2-SBEW1)*(ORSBDY/SBDYER)
           CALL IGRF(DTIME,CRLAT,CRLONG,PHT)
           ANOM=ORMAG-PHT
           WRITE (4,4) YEAR,MONTH,DAY,TIME,CRLAT,CRLONG,ORMAG,PHT,ANOM
        ENDIF
      GOTO 6

   8  CLOSE (1)
      CLOSE (2)
      CLOSE (3)
      ENDFILE 4
```

```
      REWIND 4

*
*  PRINTING PART
*  -------------
*
   9  BEEP=CHAR(7)
      WRITE (*,'(1X,A)') BEEP
      WRITE (*,'(A\)') ' Printer Ready (Y/N) ? '
      READ (*,'(A)') RS
          IF ((RS .NE. 'Y') .AND. (RS .NE. 'y')) GOTO 9
*
*  Control characters for the printer
*
      ESC=CHAR(27)
      SO=CHAR(14)
      SI=CHAR(15)
      FF=CHAR(12)
*
*  Preparing the title, printing it and the comments
*
      DO 10 I=40,1,-1
         IF (TIT(I) .NE. ' ') GOTO 11
  10  CONTINUE
  11  WRITE (5,'(1X,7A)') ESC,'E',ESC,'L010',ESC,'!',SO
      NA=INT((40-I)/2)+1
      DO 12 NC=1,NA
  12     WRITE (5,'(A\)') ' '
      WRITE (5,'(2A\)') ESC,'X'
      DO 13 NC=1,I
  13     WRITE (5,'(A\)') TIT(NC)
      WRITE (5,'(2A)') ESC,'Y'
      WRITE (5,'(1X,3A////)') SI,ESC,'N'
      WRITE (5,'(1X,A/)') 'CORRECTED NAVIGATION AND MAGNETIC ANOMALY'
      WRITE (5,'(1X,4A)') ESC,'"',ESC,'E'
*
*  Printing the column headings and the data
*
      I=1
      PAGE=1
  14  WRITE (5,'(1X,A)') 'POINT  YEAR  MONTH  DAY  TIME     LAT      LON
     +G    MESURED  COMPUTD  ANOMALY'
      WRITE (5,'(1X,A/)') ' no.    yy     mm    dd  hhmm   dd.mmm   ddd.
     +mmm      nT       nT       nT'
  15  READ (4,4,END=17) YEAR,MONTH,DAY,TIME,CRLAT,CRLONG,ORMAG,PHT,ANOM
      WRITE (5,5) I,YEAR,MONTH,DAY,TIME,CRLAT,CRLONG,ORMAG,PHT,ANOM
      I=I+1
          IF ((PAGE .EQ. 1) .AND. (I .LT. 48)) GOTO 15
          IF ((PAGE .EQ. 1) .AND. (I .EQ. 48)) GOTO 16
          IF (I .NE. (48+(PAGE-1)*57)) GOTO 15
*
*  Page changing
*
  16  PAGE=PAGE+1
      WRITE (5,'(1X,A)') FF
      GOTO 14

*
*  END OF THE PROGRAM
*  ------------------
*
  17  WRITE (5,'(/)')
      CLOSE (4)

      END


      SUBROUTINE IGRF(DTIME,CRLAT,CRLONG,PHT)
*
*   IGRF 1980 model magnetic field
*
*  This subroutine comes from a program computed the synthetic magnetic
*  anomaly along an oceanic profile. This program has been originally
*  written by D.P. McKenzie at sea, on Argo (July 1968) (revised in
*  1969), then a version for IBM 1130 (Fortran IV) has been compiled for
*  use at COB including a symmetrical plot (May 1971). The last version
*  has been written for a Zenith 100 computer in Fortran 77 (Dec. 1984).
*
      INTEGER AGH(80),PHT
      REAL DGH(80),CL(8),SL(8),P(44),Q(44)
      REAL*8 DTIME,DPONE,DPTWO,DPTHREE,X,Y,Z
      COMMON PI,A2,B2
*
*  DATA TABLES
*   AGH : spherical harmonic coefficients of the geomagnetic field (nT)
*   DGH : derivatives of coefficients with time
*         (the values for AGH and DGH are from IAGA Division 1, Working
*          Group 1, 1981)
*

      DATA AGH /-29988,-1957,5606,-1997,3028,-2129,1662,-199,1279,-2181,
     +-335,1251,271,833,-252,938,783,212,398,-257,-419,53,199,-298,-219,
     +357,46,261,149,-74,-150,-162,-78,-48,92,49,65,-15,42,93,-192,71,4,
     +-43,14,-2,-108,17,70,-59,-83,2,-28,20,-5,-13,16,1,18,11,-23,-2,-10
     +,20,7,7,1,-18,-11,4,-7,-22,4,9,3,16,7,-13,-1,-15/

      DATA DGH /22.4,11.3,-15.9,-18.3,3.2,-12.7,7.,-25.2,0.,-6.5,-.2,-.7
     +,2.7,1.,-7.9,-1.4,-1.4,4.6,-8.2,1.6,-1.8,2.9,-5.,.4,1.5,.4,1.8,-.8
     +,-.4,-3.3,0.,.2,1.3,1.4,2.1,.4,0.,-.5,3.4,-1.4,.8,0.,.8,-1.6,.3,.5
     +,-.1,0.,-1.,-.8,-.4,.4,.4,.5,.2,1.6,1.4,.1,-.5,.1,-.1,0.,1.1,.8,-.
     +2,-.1,-.3,-.7,.3,0.,-.8,-.8,-.2,.2,.7,.2,-.3,-1.1,1.2,.8/

      COLAT=90.-CRLAT
      ONE=COLAT*PI/180
      SLAT=COS(ONE)
      CLAT=SIN(ONE)
      ONE=CRLONG*PI/180
      CL(1)=COS(ONE)
      SL(1)=SIN(ONE)
      X=0.
      Y=0.
      Z=0.
      L=1
      M=1
      N=0
      ONE=A2*CLAT*CLAT
```

```
      TWO=B2*SLAT*SLAT
      THREE=ONE+TWO
      FOUR=SQRT(THREE)
      R=SQRT((A2*ONE+B2*TWO)/THREE)
      CD=FOUR/R
      SD=(A2-B2)/FOUR*SLAT*CLAT/R
      ONE=SLAT
      SLAT=SLAT*CD-CLAT*SD
      CLAT=CLAT*CD+ONE*SD
      RATIO=6371.2/R
      P(1)=2*SLAT
      P(2)=2*CLAT
      P(3)=3.*3./2*SLAT*SLAT-3./2
      P(4)=SQRT(3.**3)*CLAT*SLAT
      Q(1)=-CLAT
      Q(2)=SLAT
      Q(3)=-3*CLAT*SLAT
      Q(4)=SQRT(3.)*(SLAT*SLAT-CLAT*CLAT)
      DO 1 K=1,44
            IF (N .LT. M) THEN
               M=0
               N=N+1
               RR=RATIO**(N+2)
               FA=N
            ENDIF
         FM=M
            IF (K .GE. 5) THEN
                 IF (M .EQ. N) THEN
                    ONE=SQRT(1-.5/FM)
                    J=K-N-1
                    P(K)=(1+1./FM)*ONE*CLAT*P(J)
                    Q(K)=ONE*(CLAT*Q(J)+SLAT/FM*P(J))
                    SL(M)=SL(M-1)*CL(1)+CL(M-1)*SL(1)
                    CL(M)=CL(M-1)*CL(1)-SL(M-1)*SL(1)
                 ELSE
                    ONE=SQRT(FA*FA-FM*FM)
                    TWO=SQRT((FA-1)**2-FM*FM)/ONE
                    THREE=(2*FA-1)/ONE
                    I=K-N
                    J=K-2*N+1
                    P(K)=(FA+1)*(THREE*SLAT/FA*P(I)-TWO/(FA-1)*P(J))
                    Q(K)=THREE*(SLAT*Q(I)-CLAT/FA*P(I))-TWO*Q(J)
                 ENDIF
            ENDIF
         DPONE=(AGH(L)+DGH(L)*DTIME)*RR
            IF (M .EQ. 0) THEN
               X=X+DPONE*Q(K)
               Z=Z-DPONE*P(K)
               L=L+1
            ELSE
               DPTWO=(AGH(L+1)+DGH(L+1)*DTIME)*RR
               DPTHREE=DPONE*CL(M)+DPTWO*SL(M)
               X=X+DPTHREE*Q(K)
               Z=Z-DPTHREE*P(K)
                   IF (CLAT .GT. 0.) THEN
                      Y=Y+(DPONE*SL(M)-DPTWO*CL(M))*FM*P(K)/((FA+1)*CLA
     +T)
                   ELSE
                      Y=Y+(DPONE*SL(M)-DPTWO*CL(M))*Q(K)*SLAT
                   ENDIF
                L=L+2
            ENDIF
         M=M+1
    1 CONTINUE
      DPONE=X
      X=X*CD+Z*SD
      Z=Z*CD-DPONE*SD
      PHT=IDINT(DSQRT(X*X+Y*Y+Z*Z))

      END
```

B

★

This program has been written to compute the position of any point of the ship track of a Seabeam map using the corrections needed for Scripps Seabeam post-processing. This corrected navigation is used to compute the magnetic anomaly of each point from the original data. The synthetic magnetic field is for the period 1975-1985.

This program used three files : SBNAVDD.DAT created by the program CBEAMNAV, ORNAVDM.DAT and ORNAVDD.DAT created by the program ORIGNAV. The program will also create a file named NAVMAGCR.DAT.

Title of the table (up to 40 characters) :
COSTA RICA I SURVEY

Printer Ready (Y/N) ? Y

C

COSTA RICA I SURVEY

CORRECTED NAVIGATION AND MAGNETIC ANOMALY

POINT no.	YEAR yy	MONTH mm	DAY dd	TIME hhmm	LAT dd.mmm	LONG ddd.mmm	MESURED nT	COMPUTD nT	ANOMALY nT
1	82	4	7	815	9.377	-85.006	37200	37124	76
2	82	4	7	820	9.367	-85.013	37200	37118	82
3	82	4	7	825	9.356	-85.021	37194	37111	83
4	82	4	7	830	9.346	-85.028	37183	37105	78
5	82	4	7	835	9.335	-85.036	37169	37098	71
6	82	4	7	840	9.323	-85.042	37154	37091	63
7	82	4	7	845	9.313	-85.051	37145	37085	60
8	82	4	7	851	9.300	-85.059	37135	37077	58
9	82	4	7	855	9.292	-85.065	37124	37072	52
10	82	4	7	900	9.280	-85.072	37106	37065	41
11	82	4	7	906	9.269	-85.080	37083	37058	25
12	82	4	7	911	9.259	-85.088	37064	37053	11
13	82	4	7	916	9.248	-85.095	37042	37046	-4
14	82	4	7	920	9.239	-85.102	37025	37041	-16
15	82	4	7	926	9.228	-85.110	37019	37034	-15
16	82	4	7	930	9.218	-85.116	37024	37028	-4
17	82	4	7	935	9.210	-85.123	37034	37023	11
18	82	4	7	940	9.202	-85.128	37040	37018	22
19	82	4	7	945	9.192	-85.134	37049	37012	37
20	82	4	7	950	9.181	-85.143	37063	37005	58
21	82	4	7	955	9.169	-85.149	37078	36998	80
22	82	4	7	1000	9.157	-85.157	37094	36991	103
23	82	4	7	1005	9.147	-85.164	37103	36985	118
24	82	4	7	1010	9.135	-85.171	37099	36977	122
25	82	4	7	1015	9.123	-85.179	37081	36970	111
26	82	4	7	1020	9.113	-85.186	37051	36964	87
27	82	4	7	1025	9.102	-85.193	37018	36958	60
28	82	4	7	1030	9.090	-85.199	36990	36950	40
29	82	4	7	1035	9.080	-85.208	36964	36944	20
30	82	4	7	1040	9.068	-85.214	36940	36937	3
31	82	4	7	1045	9.058	-85.221	36918	36931	-13
32	82	4	7	1050	9.046	-85.228	36898	36924	-26
33	82	4	7	1055	9.036	-85.236	36878	36917	-39
34	82	4	7	1100	9.025	-85.243	36860	36911	-51
35	82	4	7	1105	9.014	-85.250	36849	36904	-55
36	82	4	7	1110	9.004	-85.258	36845	36898	-53
37	82	4	7	1115	8.993	-85.265	36846	36891	-45
38	82	4	7	1120	8.981	-85.273	36843	36884	-41
39	82	4	7	1125	8.972	-85.280	36838	36879	-41
40	82	4	7	1130	8.960	-85.288	36831	36872	-41
41	82	4	7	1135	8.949	-85.295	36824	36865	-41
42	82	4	7	1140	8.939	-85.303	36816	36859	-43
43	82	4	7	1145	8.928	-85.310	36808	36852	-44
44	82	4	7	1150	8.918	-85.318	36805	36846	-41
45	82	4	7	1155	8.908	-85.326	36806	36840	-34
46	82	4	7	1250	8.912	-85.349	36785	36843	-58
47	82	4	7	1255	8.922	-85.341	36778	36849	-71

POINT no.	YEAR yy	MONTH mm	DAY dd	TIME hhmm	LAT dd.mmm	LONG ddd.mmm	MESURED nT	COMPUTD nT	ANOMALY nT
48	82	4	7	1300	8.932	-85.334	36779	36855	-76
49	82	4	7	1305	8.942	-85.325	36787	36861	-74
50	82	4	7	1310	8.951	-85.319	36799	36866	-67
51	82	4	7	1315	8.961	-85.310	36810	36872	-62
52	82	4	7	1321	8.972	-85.302	36822	36879	-57
53	82	4	7	1326	8.982	-85.293	36831	36885	-54
54	82	4	7	1330	8.990	-85.288	36839	36890	-51
55	82	4	7	1336	9.002	-85.279	36846	36897	-51
56	82	4	7	1341	9.013	-85.272	36853	36904	-51
57	82	4	7	1346	9.025	-85.267	36863	36911	-48
58	82	4	7	1351	9.036	-85.261	36879	36918	-39
59	82	4	7	1355	9.046	-85.256	36902	36923	-21
60	82	4	7	1401	9.059	-85.248	36925	36931	-6
61	82	4	7	1405	9.069	-85.244	36951	36937	14
62	82	4	7	1410	9.079	-85.239	36979	36943	36
63	82	4	7	1415	9.090	-85.233	37003	36950	53
64	82	4	7	1420	9.100	-85.228	37026	36956	70
65	82	4	7	1425	9.111	-85.223	37054	36963	91
66	82	4	7	1430	9.121	-85.215	37088	36969	119
67	82	4	7	1435	9.133	-85.209	37121	36976	145
68	82	4	7	1440	9.144	-85.203	37148	36983	165
69	82	4	7	1445	9.154	-85.197	37158	36989	169
70	82	4	7	1450	9.165	-85.190	37147	36996	151
71	82	4	7	1455	9.174	-85.198	37135	37001	134
72	82	4	7	1500	9.177	-85.210	37133	37003	130
73	82	4	7	1505	9.166	-85.215	37162	36996	166
74	82	4	7	1510	9.154	-85.223	37172	36989	183
75	82	4	7	1515	9.143	-85.230	37163	36982	181
76	82	4	7	1520	9.131	-85.236	37146	36975	171
77	82	4	7	1525	9.120	-85.243	37131	36968	163
78	82	4	7	1530	9.108	-85.250	37116	36961	155
79	82	4	7	1535	9.097	-85.257	37098	36954	144
80	82	4	7	1540	9.085	-85.266	37070	36947	123
81	82	4	7	1545	9.074	-85.275	37040	36940	100
82	82	4	7	1550	9.064	-85.282	37002	36934	68
83	82	4	7	1555	9.052	-85.288	36975	36927	48
84	82	4	7	1600	9.041	-85.297	36955	36920	35
85	82	4	7	1605	9.031	-85.303	36938	36914	24
86	82	4	7	1610	9.019	-85.310	36921	36907	14
87	82	4	7	1615	9.009	-85.317	36901	36901	0
88	82	4	7	1620	9.000	-85.326	36881	36896	-15
89	82	4	7	1625	8.988	-85.334	36862	36889	-27
90	82	4	7	1630	8.977	-85.343	36846	36882	-36
91	82	4	7	1635	8.967	-85.351	36830	36876	-46
92	82	4	7	1640	8.956	-85.360	36815	36869	-54
93	82	4	7	1645	8.944	-85.368	36804	36862	-58
94	82	4	7	1650	8.949	-85.378	36782	36865	-83
95	82	4	7	1655	8.955	-85.390	36766	36869	-103
96	82	4	7	1700	8.961	-85.400	36750	36873	-123
97	82	4	7	1705	8.966	-85.391	36759	36875	-116
98	82	4	7	1710	8.975	-85.382	36779	36880	-101
99	82	4	7	1715	8.984	-85.375	36800	36886	-86
100	82	4	7	1720	8.994	-85.366	36825	36892	-67

POINT no.	YEAR yy	MONTH mm	DAY dd	TIME hhmm	LAT dd.mmm	LONG ddd.mmm	MESURED nT	COMPUTD nT	ANOMALY nT
561	82	4	9	755	9.067	-85.553	37136	36936	200
562	82	4	9	800	9.080	-85.552	37130	36944	186
563	82	4	9	805	9.092	-85.550	37111	36951	160
564	82	4	9	810	9.105	-85.549	37072	36959	113
565	82	4	9	815	9.119	-85.545	37022	36967	55
566	82	4	9	820	9.130	-85.544	36962	36974	-12
567	82	4	9	825	9.144	-85.542	36912	36982	-70
568	82	4	9	830	9.157	-85.541	36865	36991	-126
569	82	4	9	835	9.169	-85.538	36832	36998	-166
570	82	4	9	840	9.183	-85.538	36816	37006	-190
571	82	4	9	845	9.175	-85.546	36817	37001	-184
572	82	4	9	850	9.164	-85.555	36818	36995	-177
573	82	4	9	855	9.156	-85.564	36822	36990	-168
574	82	4	9	901	9.144	-85.572	36833	36982	-149
575	82	4	9	906	9.134	-85.580	36847	36976	-129
576	82	4	9	910	9.128	-85.586	36864	36972	-108
577	82	4	9	915	9.118	-85.594	36884	36966	-82
578	82	4	9	920	9.108	-85.602	36904	36960	-56
579	82	4	9	925	9.101	-85.607	36923	36956	-33
580	82	4	9	930	9.092	-85.614	36941	36951	-10
581	82	4	9	935	9.082	-85.622	36957	36945	12
582	82	4	9	940	9.073	-85.629	36975	36939	36
583	82	4	9	945	9.062	-85.637	36992	36933	59
584	82	4	9	950	9.054	-85.643	37010	36928	82
585	82	4	9	955	9.042	-85.653	37027	36921	106
586	82	4	9	1000	9.053	-85.656	37010	36927	83
587	82	4	9	1005	9.065	-85.660	36918	36934	-16
588	82	4	9	1010	9.076	-85.663	36832	36941	-109
589	82	4	9	1015	9.087	-85.666	36768	36948	-180
590	82	4	9	1020	9.099	-85.670	36727	36955	-228
591	82	4	9	1025	9.111	-85.673	36699	36962	-263
592	82	4	9	1030	9.122	-85.676	36686	36969	-283
593	82	4	9	1035	9.134	-85.680	36696	36976	-280
594	82	4	9	1040	9.145	-85.683	36726	36983	-257
595	82	4	9	1045	9.157	-85.686	36762	36990	-228
596	82	4	9	1050	9.167	-85.690	36798	36996	-198
597	82	4	9	1055	9.180	-85.693	36834	37004	-170
598	82	4	9	1100	9.190	-85.696	36866	37010	-144
599	82	4	9	1105	9.202	-85.698	36897	37017	-120
600	82	4	9	1110	9.214	-85.701	36928	37025	-97
601	82	4	9	1115	9.226	-85.703	36955	37032	-77
602	82	4	9	1120	9.238	-85.706	36976	37039	-63
603	82	4	9	1125	9.250	-85.708	36994	37046	-52
604	82	4	9	1130	9.261	-85.711	37010	37053	-43
605	82	4	9	1135	9.273	-85.714	37021	37060	-39
606	82	4	9	1140	9.287	-85.718	37029	37069	-40
607	82	4	9	1145	9.299	-85.722	37030	37076	-46
608	82	4	9	1150	9.309	-85.736	37030	37082	-52
609	82	4	9	1155	9.298	-85.743	37027	37075	-48
610	82	4	9	1200	9.287	-85.749	37023	37069	-46
611	82	4	9	1205	9.278	-85.756	37018	37063	-45
612	82	4	9	1210	9.267	-85.764	37021	37057	-36
613	82	4	9	1215	9.256	-85.770	37026	37050	-24
614	82	4	9	1220	9.247	-85.777	37029	37044	-15
615	82	4	9	1225	9.243	-85.781	37029	37042	-13
616	82	4	9	1230	9.245	-85.786	37020	37043	-23
617	82	4	9	1235	9.256	-85.791	37009	37050	-41

POINT no.	YEAR yy	MONTH mm	DAY dd	TIME hhmm	LAT dd.mmm	LONG ddd.mmm	MESURED nT	COMPUTD nT	ANOMALY nT
618	82	4	9	1240	9.270	-85.796	36993	37058	-65
619	82	4	9	1245	9.283	-85.801	36993	37066	-73
620	82	4	9	1250	9.296	-85.806	37021	37074	-53
621	82	4	9	1255	9.309	-85.812	37064	37082	-18
622	82	4	9	1300	9.323	-85.817	37102	37090	12
623	82	4	9	1305	9.334	-85.820	37126	37097	29
624	82	4	9	1310	9.347	-85.827	37140	37105	35
625	82	4	9	1315	9.361	-85.832	37144	37113	31
626	82	4	9	1321	9.376	-85.836	37137	37122	15
627	82	4	9	1325	9.386	-85.838	37135	37128	7
628	82	4	9	1330	9.399	-85.841	37149	37136	13

1.4 A

```
*-----------------------------------------------------------------------*
*                                                                       *
*                          PROGRAM 'ANOCOREC'                           *
*                                                                       *
*                                    Jacqueline ROUMP, Jan. 1985        *
*-----------------------------------------------------------------------*

      PROGRAM ANOCOREC

*  This program has been written to put the corrected magnetic anomalies
*  of crossing points of the magnetic map drawn from data processed by
*  program NAVMAGCR, into a sequential data file named ANOCOREC.DAT. The
*  differences at crossings were up to 60 nT on the Costa Rica I Survey,
*  and may be attributed to diurnal variation. The created data file will
*  be used by program MAGCOREC to change all the values of the survey.
*  Principle of the correction method :
*  choose a first base hour and from this first point write the
*  difference of the corresponding crossing point ;
*  choose a second point, close to one of the other two, estimate the
*  difference (this difference is small during the night), and write the
*  difference for the corresponding crossing point ;
*  continue the same as long as possible, sometimes the value at one point
*  point must be averaged between two others ;
*  once this is done for every point it is useful to plot the differences
*  along a time axis, and if you see that some estimations and averagings
*  have been over- or under-estimated do the work all over again.
*  The last step is to change the base to have a mean difference along the
*  whole axis close to zero.
*  The data file may be corrected using a text editor.

*
*  PREPARATION
*  -----------
*
      INTEGER YEAR,MONTH,DAY,TIME,ANOCOR

      OPEN (1, FILE='ANOCOREC.DAT', STATUS='NEW')

*
*  FORMATS
*  -------
*
*  1 : to write the corrected data into a file

    1 FORMAT (1X,3(I2,1X),I4,1X,I5)

*
*  PRESENTATION
*  ------------
*
      WRITE (*,'(A)') ' This program has been written to put the correct
     +ed magnetic anomalies'
      WRITE (*,'(A)') ' of crossing points of the magnetic map drawn fro
     +m data processed by'
      WRITE (*,'(A)') ' program NAVMAGCR, into a sequential data file na
     +med ANOCOREC.DAT. The'
      WRITE (*,'(A)') ' differences at crossings were up to 60 nT on the
     + Costa Rica I Survey,'
      WRITE (*,'(A)') ' and may be attributed to diurnal variation. The
     +created data file will'
      WRITE (*,'(A)') ' be used by program MAGCOREC to change all the va
     +lues of the survey.'
      WRITE (*,'(A)') ' Principle of the correction method :'
      WRITE (*,'(A)') ' choose a first base hour and from this first poi
     +nt write the'
      WRITE (*,'(A)') ' difference of the corresponding crossing point ;
     +'
      WRITE (*,'(A)') ' choose a second point, close to one of the other
     + two, estimate the'
      WRITE (*,'(A)') ' difference (this difference is small during the
     +night), and write the'
      WRITE (*,'(A)') ' difference for the corresponding crossing point
     +;'
      WRITE (*,'(A)') ' continue the same as long as possible, sometimes
     + the value at one point'
      WRITE (*,'(A)') ' point must be averaged between two others ;'
      WRITE (*,'(A)') ' once this is done for every point it is useful t
     +o plot the differences'
      WRITE (*,'(A)') ' along a time axis, and if you see that some esti
     +mations and averagings'
      WRITE (*,'(A)') ' have been over- or under-estimated do the work a
     +ll over again.'
      WRITE (*,'(A)') ' The last step is to change the base to have a me
     +an difference along the'
      WRITE (*,'(A)') ' whole axis close to zero.'
      WRITE (*,'(A)') ' The data file may be corrected using a text edit
     +or.'

*
*  QUESTIONS
*  ---------
*
      WRITE (*,'(A)') ' Now you will input your data, for the last one,
     +type 0 for the YEAR'

      N=1
    2 WRITE (*,'(/A,I4)') '  POINT ',N
      WRITE (*,'(A\)') ' Year (YY) ? '
      READ (*,'(I2)') YEAR
          IF (YEAR .EQ. 0) GOTO 3
      WRITE (*,'(A\)') ' Month (MM) ? '
      READ (*,'(I2)') MONTH
      WRITE (*,'(A\)') ' Day (DD) ? '
      READ (*,'(I2)') DAY
      WRITE (*,'(A\)') ' Time (HHMM) ? '
      READ (*,'(I4)') TIME
      WRITE (*,'(A\)') ' Magnetic correction (CCCCC) ? '
      READ (*,'(I5)') ANOCOR
      WRITE (1,1) YEAR, MONTH, DAY, TIME, ANOCOR
      N=N+1
      GOTO 2

*
*  END OF THE PROGRAM
*  ------------------
```

```
*
   3  ENDFILE 1
      CLOSE (1)
      END
```

B
★

This program has been written to put the corrected magnetic anomalies of crossing points of the magnetic map drawn from data processed by program NAVMAGCR, into a sequential data file named ANOCOREC.DAT. The differences at crossings were up to 60 nT on the Costa Rica I Survey, and may be attributed to diurnal variation. The created data file will be used by program MAGCOREC to change all the values of the survey.
Principle of the correction method :
choose a first base hour and from this first point write the difference of the corresponding crossing point ;
choose a second point, close to one of the other two, estimate the difference (this difference is small during the night), and write the difference for the corresponding crossing point ;
continue the same as long as possible, sometimes the value at one point point must be averaged between two others ;
once this is done for every point it is useful to plot the differences along a time axis, and if you see that some estimations and averagings have been over- or under-estimated do the work all over again.
The last step is to change the base to have a mean difference along the whole axis close to zero.
The data file may be corrected using a text editor.
Now you will input your data, for the last one, type 0 for the YEAR

```
 POINT    1
Year (YY) ? 82
Month (MM) ? 4
Day (DD) ? 7
Time (HHMM) ? 1117
Magnetic correction (CCCCC) ? 0

 POINT    2
Year (YY) ? 82
Month (MM) ? 4
Day (DD) ? 7
Time (HHMM) ? 1120
Magnetic correction (CCCCC) ? 0

 POINT    3
Year (YY) ? 82
Month (MM) ? 4
Day (DD) ? 7
Time (HHMM) ? 1331
Magnetic correction (CCCCC) ? 6

 POINT    4
Year (YY) ? 82
Month (MM) ? 4
Day (DD) ? 7
Time (HHMM) ? 1341
Magnetic correction (CCCCC) ? 6
```

```
 POINT    5
Year (YY) ? 82
Month (MM) ? 4
Day (DD) ? 7
Time (HHMM) ? 1448
Magnetic correction (CCCCC) ? -14

 POINT    6
Year (YY) ? 82
Month (MM) ? 4
Day (DD) ? 7
Time (HHMM) ? 1612
Magnetic correction (CCCCC) ? -36

 POINT    7
Year (YY) ? 82
Month (MM) ? 4
Day (DD) ? 7
Time (HHMM) ? 1625
Magnetic correction (CCCCC) ? -46

 POINT    8
Year (YY) ? 82
Month (MM) ? 4
Day (DD) ? 7
Time (HHMM) ? 1740
Magnetic correction (CCCCC) ? -30

 POINT    9
Year (YY) ? 82
Month (MM) ? 4
Day (DD) ? 7
Time (HHMM) ? 1900
Magnetic correction (CCCCC) ? -15
```

.
.
.
.
.
.
.
.
.

```
 POINT   55
Year (YY) ? 82
Month (MM) ? 4
Day (DD) ? 9
Time (HHMM) ? 401
Magnetic correction (CCCCC) ? 12

 POINT   56
Year (YY) ? 82
Month (MM) ? 4
Day (DD) ? 9
Time (HHMM) ? 408
Magnetic correction (CCCCC) ? 8

 POINT   57
Year (YY) ? 82
Month (MM) ? 4
Day (DD) ? 9
Time (HHMM) ? 419
Magnetic correction (CCCCC) ? 6

 POINT   58
Year (YY) ? 82
Month (MM) ? 4
Day (DD) ? 9
Time (HHMM) ? 440
Magnetic correction (CCCCC) ? 4

 POINT   59
Year (YY) ? 82
Month (MM) ? 4
Day (DD) ? 9
Time (HHMM) ? 450
Magnetic correction (CCCCC) ? 2

 POINT   60
Year (YY) ? 82
Month (MM) ? 4
Day (DD) ? 9
Time (HHMM) ? 505
Magnetic correction (CCCCC) ? 0

 POINT   61
Year (YY) ? 82
Month (MM) ? 4
Day (DD) ? 9
Time (HHMM) ? 517
Magnetic correction (CCCCC) ? 5
```

```
 POINT   62
Year (YY) ? 82
Month (MM) ? 4
Day (DD) ? 9
Time (HHMM) ? 630
Magnetic correction (CCCCC) ? -10

 POINT   63
Year (YY) ? 82
Month (MM) ? 4
Day (DD) ? 9
Time (HHMM) ? 640
Magnetic correction (CCCCC) ? -22

 POINT   64
Year (YY) ? 82
Month (MM) ? 4
Day (DD) ? 9
Time (HHMM) ? 652
Magnetic correction (CCCCC) ? -20

 POINT   65
Year (YY) ? 82
Month (MM) ? 4
Day (DD) ? 9
Time (HHMM) ? 705
Magnetic correction (CCCCC) ? -15

 POINT   66
Year (YY) ? 82
Month (MM) ? 4
Day (DD) ? 9
Time (HHMM) ? 910
Magnetic correction (CCCCC) ? 38

 POINT   67
Year (YY) ? 0
```

1.5
A

```
*-----------------------------------------------------------------------*
*                                                                       *
*                          PROGRAM 'MAGCOREC'                           *
*                                                                       *
*                                        Jacqueline ROUMP, Jan. 1985    *
*-----------------------------------------------------------------------*

      PROGRAM MAGCOREC

*  This program has been written to correct as much as possible the
*  magnetic data after the NAVMAGCR program run. The correction appears
*  necessary because some crossings show as much as 60 nT difference.
*  These differences at crossings may be attributed to diurnal variation.
*
*  The correction on every point of the whole file NAVMAGCR.DAT is made
*  by simple interpolation between fixes written in a file named
*  ANOCOREC.DAT by ANOCOREC program, and put into a file named
*  MAGCOREC.DAT before being printed on the line printer. The file named
*  ORNAVDD.DAT is also used for the date in decimal year.
*  In fact the whole file will not be corrected : the corrected before
*  the first corrected point will be 0, and the correction will stop
*  after the last crossing point.

*
*  PREPARATION
*  -----------
*
      INTEGER CRYEAR,CRMON,CRDAY,CRTIME,ANOCR1,ANOCR2,YEAR,MONTH,DAY,TIM
     +E,ORMAG,PHT,ANOM,CRANOM,PAGE
      REAL CRLAT,CRLONG
      REAL*8 YDEC,YCRDC1,YCRDC2
      CHARACTER TIT(40),BEEP,RS,ESC,SO,SI,FF,TITLE*40

      EQUIVALENCE (TITLE,TIT)

      OPEN (1, FILE='ANOCOREC.DAT')
      OPEN (2, FILE='ORNAVDD.DAT')
      OPEN (3, FILE='NAVMAGCR.DAT')
      OPEN (4, FILE='MAGCOREC.DAT', STATUS='NEW')
      OPEN (5, FILE='PRN')

*
*  FORMATS
*  -------
*
*  1 : to read the crossing corrected magnetic data
*  2 : to read the original data time in decimal year
*  3 : to read the data to be corrected
*  4 : to write the corrected data into a file
*  5 : to write the corrected data on the printer

   1  FORMAT (1X,3(I2,1X),I4,1X,I5)
   2  FORMAT (1X,F10.7,1X,F9.5,1X,F8.3,1X,I5)
   3  FORMAT (1X,3(I2,1X),I4,1X,F9.5,1X,F10.5,3(1X,I5))
   4  FORMAT (1X,3(I2,1X),I4,1X,F9.5,1X,F10.5,2(1X,I5))
   5  FORMAT (1X,I4,4X,I2,5X,I2,4X,I2,2X,I4,2X,F7.3,2X,F8.3,2(4X,I5))

*
*  PRESENTATION
*  ------------
*
      WRITE (*,'(A)') ' This program has been written to correct as much
     + as possible the '
      WRITE (*,'(A)') ' magnetic data after the NAVMAGCR program run. Th
     +e correction appears'
      WRITE (*,'(A)') ' necessary because some crossings show as much as
     + 60 nT difference.'
      WRITE (*,'(A/)') ' These differences at crossings may be attribute
     +d to diurnal variation.'
      WRITE (*,'(A)') ' The correction on every point of the whole file
     +NAVMAGCR.DAT is made'
      WRITE (*,'(A)') ' by simple interpolation between fixes written in
     + a file named'
      WRITE (*,'(A)') ' ANOCOREC.DAT by ANOCOREC program, and put into a
     + file named'
      WRITE (*,'(A)') ' MAGCOREC.DAT before being printed on the line pr
     +inter. The file named'
      WRITE (*,'(A)') ' ORNAVDD.DAT is also used for the date in decimal
     + year.'
      WRITE (*,'(A)') ' In fact the whole file will not be corrected : t
     +he corrected before'
      WRITE (*,'(A)') ' the first corrected point will be 0, and the cor
     +rection will stop'
      WRITE (*,'(A/)') ' after the last crossing point.'

*
*  QUESTION
*  --------
*
      WRITE (*,'(A)') ' Title of the table (up to 40 characters) :'
      READ (*,'(A)') TITLE
      WRITE (*,'(//)')

*
*  COMPUTATION PART
*  ----------------
*
      READ (1,1) CRYEAR,CRMON,CRDAY,CRTIME,ANOCR1
      CALL YCMPT(CRYEAR,CRMON,CRDAY,CRTIME,YCRDC1)
      READ (1,1) CRYEAR,CRMON,CRDAY,CRTIME,ANOCR2
      CALL YCMPT(CRYEAR,CRMON,CRDAY,CRTIME,YCRDC2)

   6  READ (2,2) YDEC
      READ (3,3) YEAR,MONTH,DAY,TIME,CRLAT,CRLONG,ORMAG,PHT,ANOM
   7    IF (YDEC .LT. YCRDC1) THEN
           CORREC=0
        ELSE
           IF ((YDEC .GE. YCRDC1) .AND. (YDEC .LT. YCRDC2)) THEN
               CORREC=ANOCR1+(YDEC-YCRDC1)/(YCRDC2-YCRDC1)*(ANOCR2-ANOC
     +R1)
           ELSE
               YCRDC1=YCRDC2
               ANOCR1=ANOCR2
               READ (1,1,END=8) CRYEAR,CRMON,CRDAY,CRTIME,ANOCR2
               CALL YCMPT(CRYEAR,CRMON,CRDAY,CRTIME,YCRDC2)
```

```
            GOTO 7
          ENDIF
        ENDIF

      CRANOM=ANOM+IFIX(CORREC)
      WRITE (4,4) YEAR,MONTH,DAY,TIME,CRLAT,CRLONG,ANOM,CRANOM
      GOTO 6

   8  CLOSE (1)
      CLOSE (2)
      CLOSE (3)
      ENDFILE 4
      REWIND 4

*
*  PRINTING PART
*  -------------
*
   9  BEEP=CHAR(7)
      WRITE (*,'(1X,A)') BEEP
      WRITE (*,'(A\)') ' Printer Ready (Y/N) ? '
      READ (*,'(A)') RS
          IF ((RS .NE. 'Y') .AND. (RS .NE. 'y')) GOTO 9
*
*  Control characters for the printer
*
      ESC=CHAR(27)
      SO=CHAR(14)
      SI=CHAR(15)
      FF=CHAR(12)
*
*  Preparing the title, printing it and the comments
*
      DO 10 I=40,1,-1
           IF (TIT(I) .NE. ' ') GOTO 11
  10  CONTINUE
  11  WRITE (5,'(1X,7A)') ESC,'E',ESC,'L012',ESC,'!',SO
      NA=INT((40-1)/2)+1
      DO 12 NC=1,NA
  12     WRITE (5,'(A\)') ' '
      WRITE (5,'(2A\)') ESC,'X'
      DO 13 NC=1,I
  13     WRITE (5,'(A\)') TIT(NC)
      WRITE (5,'(2A)') ESC,'Y'
      WRITE (5,'(1X,3A////)') SI,ESC,'N'
      WRITE (5,'(1X,A/)') 'MAGNETIC ANOMALIES CORRECTED FOR DIURNAL VARI
     +ATIONS'
      WRITE (5,'(1X,4A)') ESC,'"',ESC,'E'
*
*  Printing the column headings and the data
*
      I=1
      PAGE=1
  14  WRITE (5,'(1X,A)') 'POINT  YEAR  MONTH  DAY  TIME     LAT      LON
     +G    ANOMALY  CORRECT'
      WRITE (5,'(1X,A/)') ' no.    yy     mm    dd  hhmm   dd.mmm   ddd.
     +mmm      nT       nT'
  15  READ (4,4,END=17) YEAR,MONTH,DAY,TIME,CRLAT,CRLONG,ANOM,CRANOM
      WRITE (5,5) I,YEAR,MONTH,DAY,TIME,CRLAT,CRLONG,ANOM,CRANOM
      I=I+1
          IF ((PAGE .EQ. 1) .AND. (I .LT. 48)) GOTO 15
          IF ((PAGE .EQ. 1) .AND. (I .EQ. 48)) GOTO 16
          IF (I .NE. (48+(PAGE-1)*57)) GOTO 15
*
*  Page changing
*
  16  PAGE=PAGE+1
      WRITE (5,'(1X,A)') FF
      GOTO 14

*
*  END OF THE PROGRAM
*  ------------------
*
  17  WRITE (5,'(/)')
      CLOSE (4)

      END


      SUBROUTINE YCMPT(YEAR,MONTH,DAY,TIME,YDEC)

*  Subroutine to compute the decimal year from year, month, day and time

      INTEGER YEAR,MONTH,DAY,TIME,MONLEN(11),NDAY
      REAL HOUR,HR,DDEC
      REAL*8 YDEC

      DATA MONLEN /31,28,31,30,31,30,2*31,30,31,30/

      NDAY=0
      HOUR=FLOAT(TIME)/100
      HR=FLOAT(IFIX(HOUR))
      DDEC=(HR+(HOUR-HR)*100/60)/24
      DO 1 I=1,MONTH-1
   1     NDAY=MONLEN(I)+NDAY
      YDEC=YEAR+(NDAY+(DAY-1)+DDEC)/365

      END
```

B

★

This program has been written to correct as much as possible the magnetic data after the NAVMAGCR program run. The correction appears necessary because some crossings show as much as 60 nT difference. These differences at crossings may be attributed to diurnal variation.

The correction on every point of the whole file NAVMAGCR.DAT is made by simple interpolation between fixes written in a file named ANOCOREC.DAT by ANOCOREC program, and put into a file named MAGCOREC.DAT before being printed on the line printer. The file named ORNAYDD.DAT is also used for the date in decimal year.
In fact the whole file will not be corrected : the corrected before the first corrected point will be 0, and the correction will stop after the last crossing point.

Title of the table (up to 40 characters) :
COSTA RICA I SURVEY

Printer Ready (Y/N) ? Y

C

COSTA RICA I SURVEY

MAGNETIC ANOMALIES CORRECTED FOR DIURNAL VARIATIONS

POINT no.	YEAR yy	MONTH mm	DAY dd	TIME hhmm	LAT dd.mmm	LONG ddd.mmm	ANOMALY nT	CORRECT nT
1	82	4	7	815	9.377	-85.006	76	76
2	82	4	7	820	9.367	-85.013	82	82
3	82	4	7	825	9.356	-85.021	83	83
4	82	4	7	830	9.346	-85.028	78	78
5	82	4	7	835	9.335	-85.036	71	71
6	82	4	7	840	9.323	-85.042	63	63
7	82	4	7	845	9.313	-85.051	60	60
8	82	4	7	851	9.300	-85.059	58	58
9	82	4	7	855	9.292	-85.065	52	52
10	82	4	7	900	9.280	-85.072	41	41
11	82	4	7	906	9.269	-85.080	25	25
12	82	4	7	911	9.259	-85.088	11	11
13	82	4	7	916	9.248	-85.095	-4	-4
14	82	4	7	920	9.239	-85.102	-16	-16
15	82	4	7	926	9.228	-85.110	-15	-15
16	82	4	7	930	9.218	-85.116	-4	-4
17	82	4	7	935	9.210	-85.123	11	11
18	82	4	7	940	9.202	-85.128	22	22
19	82	4	7	945	9.192	-85.134	37	37
20	82	4	7	950	9.181	-85.143	58	58
21	82	4	7	955	9.169	-85.149	80	80
22	82	4	7	1000	9.157	-85.157	103	103
23	82	4	7	1005	9.147	-85.164	118	118
24	82	4	7	1010	9.135	-85.171	122	122
25	82	4	7	1015	9.123	-85.179	111	111
26	82	4	7	1020	9.113	-85.186	87	87
27	82	4	7	1025	9.102	-85.193	60	60
28	82	4	7	1030	9.090	-85.199	40	40
29	82	4	7	1035	9.080	-85.208	20	20
30	82	4	7	1040	9.068	-85.214	3	3
31	82	4	7	1045	9.058	-85.221	-13	-13
32	82	4	7	1050	9.046	-85.228	-26	-26
33	82	4	7	1055	9.036	-85.236	-39	-39
34	82	4	7	1100	9.025	-85.243	-51	-51
35	82	4	7	1105	9.014	-85.250	-55	-55
36	82	4	7	1110	9.004	-85.258	-53	-53
37	82	4	7	1115	8.993	-85.265	-45	-45
38	82	4	7	1120	8.981	-85.273	-41	-41
39	82	4	7	1125	8.972	-85.280	-41	-41
40	82	4	7	1130	8.960	-85.288	-41	-41
41	82	4	7	1135	8.949	-85.295	-41	-41
42	82	4	7	1140	8.939	-85.303	-43	-43
43	82	4	7	1145	8.928	-85.310	-44	-43
44	82	4	7	1150	8.918	-85.318	-41	-40
45	82	4	7	1155	8.908	-85.326	-34	-33
46	82	4	7	1250	8.912	-85.349	-58	-54
47	82	4	7	1255	8.922	-85.341	-71	-67

POINT no.	YEAR yy	MONTH mm	DAY dd	TIME hhmm	LAT dd.mmm	LONG ddd.mmm	ANOMALY nT	CORRECT nT
48	82	4	7	1300	8.932	-85.334	-76	-72
49	82	4	7	1305	8.942	-85.325	-74	-70
50	82	4	7	1310	8.951	-85.319	-67	-62
51	82	4	7	1315	8.961	-85.310	-62	-57
52	82	4	7	1321	8.972	-85.302	-57	-52
53	82	4	7	1326	8.982	-85.293	-54	-49
54	82	4	7	1330	8.990	-85.288	-51	-46
55	82	4	7	1336	9.002	-85.279	-51	-45
56	82	4	7	1341	9.013	-85.272	-51	-45
57	82	4	7	1346	9.025	-85.267	-48	-44
58	82	4	7	1351	9.036	-85.261	-39	-36
59	82	4	7	1355	9.046	-85.256	-21	-20
60	82	4	7	1401	9.059	-85.248	-6	-6
61	82	4	7	1405	9.069	-85.244	14	13
62	82	4	7	1410	9.079	-85.239	36	34
63	82	4	7	1415	9.090	-85.233	53	49
64	82	4	7	1420	9.100	-85.228	70	65
65	82	4	7	1425	9.111	-85.223	91	84
66	82	4	7	1430	9.121	-85.215	119	111
67	82	4	7	1435	9.133	-85.209	145	135
68	82	4	7	1440	9.144	-85.203	165	154
69	82	4	7	1445	9.154	-85.197	169	156
70	82	4	7	1450	9.165	-85.190	151	137
71	82	4	7	1455	9.174	-85.198	134	119
72	82	4	7	1500	9.177	-85.210	130	113
73	82	4	7	1505	9.166	-85.215	166	148
74	82	4	7	1510	9.154	-85.223	183	164
75	82	4	7	1515	9.143	-85.230	181	160
76	82	4	7	1520	9.131	-85.236	171	149
77	82	4	7	1525	9.120	-85.243	163	140
78	82	4	7	1530	9.108	-85.250	155	131
79	82	4	7	1535	9.097	-85.257	144	118
80	82	4	7	1540	9.085	-85.266	123	96
81	82	4	7	1545	9.074	-85.275	100	72
82	82	4	7	1550	9.064	-85.282	68	38
83	82	4	7	1555	9.052	-85.288	48	17
84	82	4	7	1600	9.041	-85.297	35	3
85	82	4	7	1605	9.031	-85.303	24	-10
86	82	4	7	1610	9.019	-85.310	14	-21
87	82	4	7	1615	9.009	-85.317	0	-38
88	82	4	7	1620	9.000	-85.326	-15	-57
89	82	4	7	1625	8.988	-85.334	-27	-72
90	82	4	7	1630	8.977	-85.343	-36	-80
91	82	4	7	1635	8.967	-85.351	-46	-89
92	82	4	7	1640	8.956	-85.360	-54	-96
93	82	4	7	1645	8.944	-85.368	-58	-99
94	82	4	7	1650	8.949	-85.378	-83	-123
95	82	4	7	1655	8.955	-85.390	-103	-142
96	82	4	7	1700	8.961	-85.400	-123	-161
97	82	4	7	1705	8.966	-85.391	-116	-153
98	82	4	7	1710	8.975	-85.382	-101	-137
99	82	4	7	1715	8.984	-85.375	-86	-121
100	82	4	7	1720	8.994	-85.366	-67	-101

POINT no.	YEAR yy	MONTH mm	DAY dd	TIME hhmm	LAT dd.mmm	LONG ddd.mmm	ANOMALY nT	CORRECT nT
504	82	4	9	310	9.022	-85.460	-136	-106
505	82	4	9	315	9.022	-85.447	-136	-105
506	82	4	9	320	9.024	-85.434	-153	-122
507	82	4	9	325	9.025	-85.420	-166	-135
508	82	4	9	330	9.025	-85.405	-156	-127
509	82	4	9	335	9.033	-85.398	-129	-103
510	82	4	9	340	9.043	-85.400	-94	-71
511	82	4	9	345	9.054	-85.404	-56	-36
512	82	4	9	350	9.066	-85.407	-26	-9
513	82	4	9	356	9.080	-85.411	3	17
514	82	4	9	400	9.090	-85.414	12	24
515	82	4	9	405	9.101	-85.420	43	52
516	82	4	9	410	9.114	-85.424	70	77
517	82	4	9	415	9.127	-85.429	80	86
518	82	4	9	420	9.138	-85.435	76	81
519	82	4	9	425	9.150	-85.441	61	66
520	82	4	9	430	9.163	-85.447	16	20
521	82	4	9	436	9.171	-85.442	-34	-30
522	82	4	9	441	9.165	-85.431	27	30
523	82	4	9	446	9.159	-85.420	69	71
524	82	4	9	451	9.152	-85.409	81	82
525	82	4	9	455	9.146	-85.400	75	76
526	82	4	9	500	9.140	-85.388	53	53
527	82	4	9	505	9.145	-85.385	43	43
528	82	4	9	510	9.154	-85.389	66	68
529	82	4	9	515	9.166	-85.392	74	78
530	82	4	9	520	9.179	-85.396	59	63
531	82	4	9	525	9.192	-85.399	17	20
532	82	4	9	530	9.205	-85.401	-39	-37
533	82	4	9	535	9.215	-85.409	-94	-93
534	82	4	9	540	9.220	-85.420	-140	-140
535	82	4	9	545	9.222	-85.432	-166	-166
536	82	4	9	551	9.222	-85.448	-183	-184
537	82	4	9	555	9.222	-85.457	-194	-196
538	82	4	9	600	9.215	-85.466	-183	-186
539	82	4	9	605	9.204	-85.466	-168	-172
540	82	4	9	610	9.190	-85.466	-139	-144
541	82	4	9	615	9.178	-85.466	-95	-101
542	82	4	9	620	9.165	-85.466	-36	-43
543	82	4	9	625	9.152	-85.464	25	17
544	82	4	9	630	9.138	-85.464	76	66
545	82	4	9	635	9.125	-85.461	111	95
546	82	4	9	640	9.112	-85.459	137	116
547	82	4	9	645	9.098	-85.456	136	115
548	82	4	9	651	9.085	-85.454	96	76
549	82	4	9	655	9.074	-85.453	73	55
550	82	4	9	700	9.061	-85.453	66	50
551	82	4	9	705	9.048	-85.452	25	11
552	82	4	9	710	9.046	-85.463	23	11
553	82	4	9	715	9.046	-85.475	65	55
554	82	4	9	720	9.048	-85.486	122	114
555	82	4	9	725	9.047	-85.498	175	169
556	82	4	9	730	9.047	-85.510	197	193
557	82	4	9	735	9.046	-85.521	200	198
558	82	4	9	740	9.046	-85.534	200	200
559	82	4	9	745	9.048	-85.546	201	202
560	82	4	9	750	9.056	-85.554	204	208

POINT no.	YEAR yy	MONTH mm	DAY dd	TIME hhmm	LAT dd.mmm	LONG ddd.mmm	ANOMALY nT	CORRECT nT
561	82	4	9	755	9.067	-85.553	200	206
562	82	4	9	800	9.080	-85.552	186	194
563	82	4	9	805	9.092	-85.550	160	170
564	82	4	9	810	9.105	-85.549	113	125
565	82	4	9	815	9.119	-85.545	55	69
566	82	4	9	820	9.130	-85.544	-12	4
567	82	4	9	825	9.144	-85.542	-70	-52
568	82	4	9	830	9.157	-85.541	-126	-105
569	82	4	9	835	9.169	-85.538	-166	-143
570	82	4	9	840	9.183	-85.538	-190	-165
571	82	4	9	845	9.175	-85.546	-184	-157
572	82	4	9	850	9.164	-85.555	-177	-148
573	82	4	9	855	9.156	-85.564	-168	-137
574	82	4	9	901	9.144	-85.572	-149	-115
575	82	4	9	906	9.134	-85.580	-129	-93

CALCULS DES MOUVEMENTS DE PLAQUES D'APRES LE MODELE RM2 DE MINSTER ET JORDAN (1978)

A - Exemple de fichier de données

B - Listing du programme 'PLATES'

C - Copie d'écran d'un exemple d'exécution du programme 'PLATES'

D - Exemple de sortie sur imprimante et sur disque

2

A

FICHIER DE DONNEES : FMATPNT.DAT

```
30
  18.678  -104.963   18.373  -104.555   17.767  -103.733
  17.372  -102.778   16.965  -101.892   16.645  -101.195
  16.340  -100.167   16.038   -99.358   15.773   -98.655
  15.495   -97.793   15.293   -97.230   15.218   -95.813
  15.007   -95.333   14.702   -94.843   14.168   -93.995
  13.817   -93.317   13.483   -92.667   12.812   -91.240
  12.505   -90.542   12.167   -89.583   11.777   -88.647
  11.555   -88.167   10.833   -87.218    9.815   -86.333
   9.150   -85.500    8.655   -84.425    8.355   -83.868
   8.178   -83.377    7.833   -82.992    7.317   -82.833
```

B

```
/*****************************************************************************/
/*                                                                           */
/*                            PROGRAM `PLATES.C'                             */
/*                                                                           */
/*  Translation of `PLATES.FOR'                 Jacqueline ROUMP, Aug. 1987  */
/*****************************************************************************/


/*    This program computes the mean speed and direction of plate motions,  */
/* the lower and upper values (confidence intervals : 70% or 95%) of speed   */
/* (with the corresponding direction) and direction (with the correspond-    */
/* ing speed), using Model RM2 (Minster & Jordan, 1978).                     */
/*                                                                           */
/*    The data must be located in a separate file having at the beginning,    */
/* on the first line the number of points in %3d format (up to 300 points    */
/* are allowed). Then there are the latitude and longitude of each point     */
/* in decimal degrees (format : %6.3f %7.3f) : 3 points per line.            */


#include <stdio.h>
#include <math.h>
#include <dir.h>

#define MAX  300                      /* maximum number of data points */
#define PI   3.141592653589887        /* pi */
#define FPI  (PI/180)
#define ESC  0x1b                     /* escape */
#define BS   0x08                     /* backspace */
#define BEEP 0x07                     /* bell */
#define sign(x) ((x < 0) ? -1 : 1)    /* sign of x */

double fnacos (double x);             /* acos */
double fnasin (double x);             /* asin */
double fndis (double x, double y, double z, double w);
double fnisos (double x, double y);
double fndir (double x, double y, double z);
double fnside (double x, double y, double z);
double fnelip (double x, double y, double z);
double direc (double rlat, double rlong, double plat, double plong, double rdis);
void posit (double si, int confin, double tr[], double pos[]);
int disdir (int confin, double tr[], double tab1[][], double extr1[],
            int ival[], double val[], int ivar[], double plat, double plong,
            double tab2[][], double extr2[], double meandis);
void clearscreen (void);
void locate (int row, int column);


main ()
  {
    struct ffblk ffblk;         /* to read directory using dir.h */

    FILE *fp;                   /* file pointer to open files */
    char data[20], data1[20], data2[20],
         fnamin[80],       /* name of input file  */
         fnamou[80];       /* name of output file */
```

```
/*     constant values                                                  */
/*     ~~~~~~~~~~~~~~~                                                  */
char ci[2][4] = { "70%", "95%" };        /* confidence interval */
char spd[2][6] = { "cm/yr", "km/MY" };   /* speed unit */

/*     table of the plate couple names                                  */
/*     ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~                                  */
char name[27][30] = { "North America - Pacific","Cocos - Pacific",
     "Nazca - Pacific","Eurasia - Pacific","India - Pacific",
     "Antarctica - Pacific","Cocos - North America",
     "Africa - North America","Eurasia - North America",
     "North America - Caribbean","Cocos - Caribbean",
     "Nazca - Caribbean","Cocos - Nazca",
     "North America - South America","Caribbean - South America",
     "Nazca - South America","Africa - South America",
     "Antarctica - South America","India - Africa","Arabia - Africa",
     "Africa - Eurasia","India - Eurasia","Arabia - Eurasia",
     "India - Arabia","Nazca - Antarctica","Africa - Antarctica",
     "India - Antarctica" };

/* CONFIN = confidence interval choice number                           */
/* UNIT = speed unit choice number                                      */
/* J[D or S[MIN or MAX [1 or 2]]] : tables to put the line number (in a */
/*                table) of best value for point considered and also    */
/*                for preceding point (if any)                          */
/* POLE = table giving the pole to use for next computations 0 = normal */
/*        1 = opposite position                                         */

int i, j, k, inpdat, fil, nl, a, confin, unit, ans, ival[4], ivar[3],
    jdmin1[2], jdmax1[2], jsmaxa1[2], jsmaxb1[2],
    jdmin2[2], jdmax2[2], jsmaxa2[2], jsmaxb2[2],
    jdmin3, jdmax3, jsmaxa3, jsmaxb3,
    na, nb;
static int pole[MAX];
float r1;                /* depend of unit to present speed */

/*     table of the plate motion parameters :                           */
/*     ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~                            */
/*     rotation vector : 1 = lat. of the pole, 2 = long. of the pole,   */
/*         3 = speed of rotation in deg. per million years              */
/*     error ellipse : 4 = Azimuth of the major axis in degrees from    */
/*         North toward East, 5 = length of the major axis in degrees,  */
/*         6 = length of the other axis in degrees                      */

float t[27][6] = { {  48.77,  -73.91,  .852, 109,  1.3 , 1.08 },
                   {  38.72, -107.39, 2.208, 143,  1.00,  .63 },
                   {  56.64,  -87.88, 1.539,   9,  1.91,  .96 },
                   {  60.64,  -78.92,  .977, 102,  1.51, 1.02 },
                   { -60.71,  174.21, 1.246,  98,   .90,  .76 },
                   { -64.67,   99.77,  .964,  52,  1.11,  .75 },
                   {  29.80, -121.28, 1.489, 105,  1.84,  .99 },
                   {  80.43,   56.36,  .258,  86,  5.88, 1.51 },
                   {  65.85,  132.44,  .231, 166,  6.36, 1.39 },
                   { -33.83,  -70.48,  .219, 167,  9.42,  .97 },
                   {  23.60, -115.55, 1.543, 117,  2.24, 1.21 },
                   {  47.30,  -97.57,  .711, 161,  5.59, 2.67 },
                   {   5.63, -124.40,  .972,  89,  2.60, 1.4  },
                   {  25.57,  -53.82,  .167, 166,  7.22, 5.49 },
                   { -73.51, -119.16,  .202, 128, 16.84, 6.84 },
                   {  59.08,  -94.75,  .835, 175,  3.77, 1.9  },
                   {  66.56,  -37.29,  .356, 172,  2.85,  .98 },
                   { -87.69, -104.80,  .302,  84,  3.22, 1.26 },
                   {  17.27,   46.02,  .644, 133,  1.24,  .66 },
                   {  30.82,    6.43,  .260, 101, 10.02, 2.93 },
                   {  25.23,  -21.19,  .104, 179,  4.25,  .89 },
                   {  19.71,   38.46,  .698, 115,  2.72,  .9  },
                   {  29.82,   -1.64,  .357,  95,  8.33, 2.45 },
                   {   7.08,   63.86,  .469, 129,  2.51, 1.89 },
                   {  43.21,  -95.02,  .605, 179,  4.50, 2.39 },
                   {   9.46,  -41.70,  .149, 138,  4.93, 1.45 },
                   {  18.67,   32.74,  .673, 118,  1.39, 1.1  } };

/* L = data points latitudes and longitudes in deg                      */
/* LR = data points in radians and results in deg (except 3rd column)   */
/* TR & OPTR = rotation pole parameters in radians                       */
/* VA,VB,V1,V2..,V3.. = tables to put intermediates computations for    */
/*                      ellipse error                                   */
/* S or D [MIN or MAX [1 to 3]] = table having the intermediate and     */
/*                final min or max distance or direction angles (in     */
/*                ellipse and values (these angles are found in line    */
/*                number JD... or JS...                                 */

double ai, ak, a1, d1, d2, dmax1[2], dmax2[2], dmax3[2], dmin1[2];
double dmin2[2], dmin3[2], l[MAX][2], lr[MAX][15], maxspeed;
double optr[6], pos[3], smaxa1[2], smaxa2[2], smaxa3[2];
double smaxb1[2], smaxb2[2], smaxb3[2], temp, tr[6];
double va[4], va[36][3], vb[36][2], v1[36][6];
double v2dis1[21][6], v3dis1[21][6], v2dis2[21][6], v3dis2[21][6];
double v2dir1[21][6], v3dir1[21][6], v2dir2[21][6], v3dir2[21][6];


/*     PRESENTATION                                                     */
/*     ============                                                     */

clearscreen ();
printf ("\n                                PROGRAM PLATES \n\n ");
printf ("    This program computes the mean speed and direction of ");
printf ("plate motion, the \nlower and upper values (confidence ");
printf ("intervals : 70% or 95%) of speed (with the \ncorresponding ");
printf ("direction) and direction (with the corresponding speed), using ");
printf ("\nModel RM2 (Minster & Jordan, 1978). \n\n");
printf ("    The data must be located in a separate file having at the ");
printf ("beginning, on the\nfirst line the number of points in %%3d ");
printf ("format (up to 300 points are allowed). \nThen there are the ");
printf ("latitude and longitude of each point in decimal degrees \n");
printf ("(format : %%6.3f %%7.3f) : 3 points per line. \n");


/*     INPUT OF THE DATA INTO THE PROGRAM AND QUESTIONS                 */
/*     ================================================                 */


/*     Old data file or new one ?                                       */
/*     --------------------------                                       */
```

```
do { printf ("\n\n\n Do you want to use an old data file (1) \n");
     printf ("                or to create a new one (2) : ");
     scanf ("%d",&inpdat);
   } while ( (inpdat != 1) && (inpdat != 2) );

if (inpdat == 1)
  {
    /*     asking old data filename and verifying                    */
    /*     ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~ */

    do {
        printf ("\n Name of your old data file : ");
        scanf ("%s", fnamin);
        fil = findfirst (fnamin,&ffblk,0);
        if (fil != 0)
           printf (" File not found\n");
       } while (fil);

    /*     writing of text explaining what the program is doing      */
    /*     ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~ */

    clearscreen ();
    locate (20,1);
    printf ("   READING THE DATA \n");
    fp = fopen (fnamin,"r");

    /*     reading the numb of pts & then the lat & long of each point */
    /*     ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~ */

    fscanf (fp,"%3d", &nl);
    for (i=0; i<nl; i++)
      {
        fscanf (fp,"%s %s ", data1, data2);
        l[i][0] = atof (data1);
        l[i][1] = atof (data2);
      }
    fclose (fp);
  }

  else

  {
    /*     new data file                                             */
    /*     ~~~~~~~~~~~~~                                             */

    /*     input of file name and number of data points              */
    /*     ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~ */

    do { printf ("\n Name to give to your new data file : ");
         scanf ("%s", &fnamin);
         fil = findfirst (fnamin,&ffblk,0);
         if (fil == 0)
           {
             printf ("   File already exist\n");
             printf ("   Do you want to overwrite it (Y/N) : ");
             while ( (ans = getchar()) == '\n' )
               continue;
           }
       } while ( (fil == 0) && (ans != 'Y') && (ans != 'y') );

    do { printf ("\n Number of data points (up to %d ) : ", MAX);
         scanf ("%d", &nl);
       } while ( (nl > MAX) || (nl <= 0) );

    /*     input of the data point latitudes and longitudes          */
    /*     ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~ */

    printf ("\n");
    for (i=0; i<nl; i++)
      {
        printf (" POINT %2d   latitude : ", i+1);
        scanf ("%s", data);
        l[i][0] = atof (data);
        printf ("           longitude : ");
        scanf ("%s", data);
        l[i][1] = atof (data);
      }

    /*     writing the data file to disk                             */
    /*     ~~~~~~~~~~~~~~~~~~~~~~~~~~~~                              */

    do { printf ("\n DISK READY (Y/N) : ");
         while ( (ans = getchar()) == '\n')
               continue;
       } while ( (ans != 'Y') && (ans != 'y') );

    fp = fopen (fnamin, "w");
    fprintf (fp,"%3d \n", nl);
    i = 0;
    while (i < nl)
      {
        fprintf (fp,"  %6.2f  %7.2f", l[i][0], l[i][1]);
        i++;
        if ( ((i % 3) == 0) || i == nl)
           fprintf (fp,"\n");
      }
    fclose (fp);
  }


/*     Choice of the plate couple by its number                      */
/*     ----------------------------------------                      */

clearscreen ();
printf ("      LIST OF THE PLATE COUPLES :\n\n");
i = 0;
while (i < 27)
   {
     printf (" %2d : %-29s  ", i+1, &name[i][0]);
     i++;
     if ( ((i % 2) == 0) || i == 27)
     printf ("\n");
   }

do { printf ("\n Give the number of the plate couple you have chosen : ");
```

```
    scanf ("%d", &a);
  } while ( (a < 1) || (a > 27) );


/*     Transformation of the plate couple param. from deg. into rad.     */
/*     ------------------------------------------------------------     */

for (i=0; i<6; i++)
    tr[i] = t[a-1][i] * FPI;


/*     Filling the opposite pole parameter table                         */
/*     -----------------------------------------                         */

optr[0] = -tr[0];
optr[1] = tr[1] - PI;
if (optr[1] < -PI)
   optr[1] += 2*PI;
for (i=2; i<6; i++)
    optr[i] = tr[i];


/*     Choice of the confidence interval for the error ellipse           */
/*     computation using Minster & Jordan values                         */
/*     ---------------------------------------------------------         */

do { printf ("\n What confidence interval do you want : 70%% (1) or ");
     printf ("95%% (2) : ");
     scanf ("%d", &confin);
   } while ( (confin != 1) && (confin != 2) );


/*     Choice of the unit to print the results                           */
/*     ---------------------------------------                           */

do { printf ("\n Do you want the results in cm/yr (1) or km/my (2) : ");
     scanf ("%d", &unit);
   } while ( (unit != 1) && (unit != 2) );

if (unit == 2)
   r1 = 1.;
else
   r1 = .1;


/*     Asking the name of the output file                                */
/*     ----------------------------------                                */

do { printf ("\n Name of the file to put data + results : ");
     scanf ("%s", &fnamou);
     fil = findfirst (fnamou,&ffblk,0);
     if (fil == 0)
       {
         printf ("   File already exist\n");
         printf ("   Do you want to overwrite it (Y/N) : ");
         while ( (ans = getchar()) == '\n' )
           continue;
       }
   } while ( (fil == 0) && (ans != 'Y') && (ans != 'y') );


/*     Translation of the lat. & long. of the data points from degrees   */
/*     into radians                                                      */
/*     ------------------------------------------------------------      */

for (i=0; i<nl; i++)
  {
    lr[i][0] = l[i][0] * FPI;
    lr[i][1] = l[i][1] * FPI;
  }


/*     COMPUTATION PART 1 : BEST FITTING POLE VALUES                     */
/*     =============================================                     */


/*     Writing of text explaining what the program is doing while        */
/*     replacing the preceding text                                      */
/*     ------------------------------------------------------------      */

clearscreen ();
locate (20,1);
printf ("COMPUTING THE MAIN RESULTS \n");


/*     Distance from the pole of rotation to each point in radians       */
/*     ----------------------------------------------------------        */

for (i=0; i<nl; i++)
  {
    lr[i][2] = fndis (tr[0],tr[1],lr[i][0],lr[i][1]);
    if (lr[i][2] > PI/2.)
      {
        lr[i][2] = fndis (optr[0],optr[1],lr[i][0],lr[i][1]);
        pole[i] = 1;
      }
  }

lr[0][3] = 0.0;
lr[0][4] = 0.0;
if (nl != 1)
  {
    /*     distance from each point to the first one in Km               */
    /*     ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~               */

    for (i=1; i<nl; i++)
        lr[i][3] = fndis (lr[0][0],lr[0][1],lr[i][0],lr[i][1])/FPI*10000/90;

    /*     distance from each point to the preceding one in Km           */
    /*     ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~            */

    for (i=1; i<nl; i++)
        lr[i][4]=fndis(lr[i-1][0],lr[i-1][1],lr[i][0],lr[i][1])/FPI*10000/90;
  }
```

```
/*     Speed in cm/yr or km/MY using the angle of rotation in deg/MY    */
/*     ------------------------------------------------------------     */

for (i=0; i<nl; i++)
    lr[i][5] = fnisos (lr[i][2],tr[2]) / FPI * 10000/90 * r1;


/*     Direction in degrees (from North toward East) of the rotation    */
/*     at each point, which is normal to the line pole of rotation-     */
/*     point on the plates limit                                        */
/*     ------------------------------------------------------------     */

for (i=0; i<nl; i++)
  {
    if (pole[i] == 0)
       lr[i][6] = direc (tr[0],tr[1],lr[i][0],lr[i][1],lr[i][2]);
    else
       lr[i][6] = direc (optr[0],optr[1],lr[i][0],lr[i][1],lr[i][2]);

    /*     to have the mean direction in degrees                        */
    /*     ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~                        */

    lr[i][6] = lr[i][6] / FPI;
  }



/*     COMPUTATION PART 2 : ERROR ELLIPSE                               */
/*     ==================================                               */


/*     Writing of text explaining what the program is doing, replacing  */
/*     the preceding text and giving the number of the point being      */
/*     computed                                                         */
/*     ------------------------------------------------------------     */

clearscreen ();
locate (20,1);
printf ("ERROR ELLIPSE COMPUTATIONS : Point No.   ");


/*     Position of the 36 points, 10 deg apart on the error ellipses    */
/*     ------------------------------------------------------------     */

for (k=0; k<=350; k=k+10)
  {
    j = k / 10;
    ai = k * FPI;
    posit (ai, confin, tr, pos);
    va[j][0] = pos[0];
    va[j][1] = pos[1];
    va[j][2] = pos[2];
    if (j < 18)
      {
        vb[j+18][0] = -pos[0];
        vb[j+18][1] = optr[1] + (tr[1] - pos[1]);
      }
    else
      {
        vb[j-18][0] = -pos[0];
        vb[j-18][1] = optr[1] + (tr[1] - pos[1]);
      }
  }


/*     Maximum speed of plate couple (90 deg from rotation pole)        */
/*     ------------------------------------------------------------     */

maxspeed =  tr[2] / FPI * 10000./90. * r1;


/*     Loop on each point to compute min & max speeds and directions    */
/*     ------------------------------------------------------------     */

for (i=0; i<nl; i++)
  {
    /*     writing the number of the point under process                */
    /*     ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~     */

    printf ("%c%c%2d", BS, BS, i+1);

    /*     initialization of v1 table with the nearest pole of rotation */
    /*     ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~ */

    if (pole[i] == 0)
      for (j=0; j<36; j++)
        {
          v1[j][0] = va[j][0];
          v1[j][1] = va[j][1];
          v1[j][5] = va[j][2];
        }
    else
      for (j=0; j<36; j++)
        {
          v1[j][0] = vb[j][0];
          v1[j][1] = vb[j][1];
          v1[j][5] = va[j][2];
        }

    /*     if the point is within the error ellipse (the distance to    */
    /*     the pole is shorter than the small axis of the ellipse) :    */
    /*     no computation                                               */
    /*     ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~ */

    d1 = fndis (lr[i][0], lr[i][1], v1[0][0], v1[0][1]);
    a1 = fndir (d1, lr[i][2], confin*tr[4]);
    if (lr[i][1] > ((pole[i] == 0) ? tr[1] : optr[1]))
       a1 = 2 * PI - a1;
    d2 = fnelip (confin*tr[4], a1, confin*tr[5]);
    if (lr[i][2] <= d2)
      { for (j=7; j<15; j++)
            lr[i][j] = 0.0;
        continue;
      }
```

```
/*     distance between points on the ellipse and the data points   */
/*     ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~   */

for (j=0; j<36; j++)
    v1[j][2] = fndis (lr[i][0],lr[i][1],v1[j][0],v1[j][1]);

/*     angle center of ellipse - point - ellipse border             */
/*     ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~   */

for (j=0; j<36; j++)
    v1[j][3] = fndir (v1[j][5],lr[i][2], v1[j][2]);

/*     search for the minimum and maximum distances                 */
/*     ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~   */

dmin1[0] = dmax1[0] = 0.0;
dmin1[1] = dmax1[1] = v1[0][2];
jdmin1[1] = jdmax1[1] = 1;

for (j=0; j<36; j++)
  {
    if (v1[j][2] < dmin1[1])
      {
        dmin1[0] = j * 10;
        dmin1[1] = v1[j][2];
        jdmin1[1] = j + 1;
      }
    if (v1[j][2] > dmax1[1])
      {
        dmax1[0] = j * 10;
        dmax1[1] = v1[j][2];
        jdmax1[1] = j + 1;
      }
  }
dmin1[1] = dmin1[1] / FPI;
dmax1[1] = dmax1[1] / FPI;

/*     search for the two separate maximum angles                   */
/*     ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~   */

if (v1[1][3] >= v1[0][3])
  {
    /*     shape of curve is up-down-up-down-up                     */
    j = 1;
    while (v1[j][3] >= v1[j-1][3])
      {                             /* searching for 1st maximum */
        smaxa1[0] = j * 10.;
        smaxa1[1] = v1[j][3];
        jsmaxa1[1] = j + 1;
        j++;
      }
    while (v1[j][3] < v1[j-1][3])    /* going to 1st minimum */
      j++;
    while (v1[j][3] >= v1[j-1][3])
      {                             /* searching for 2nd maximum */
        smaxb1[0] = j * 10.;
        smaxb1[1] = v1[j][3];
        jsmaxb1[1] = j + 1;
        j++;
      }
    if (j-1 == 35)
      {
        smaxb1[0] = 0.;
        smaxb1[1] = v1[0][3];
        jsmaxb1[1] = 1;
      }
    else
      {
        while (v1[j][3] >= v1[j-1][3] && j < 36)
          {                         /* searching for 2nd maximum */
            smaxb1[0] = j * 10.;
            smaxb1[1] = v1[j][3];
            jsmaxb1[1] = j + 1;
            j++;
          }
        if (j-1 == 35)      /* 2nd maximum might be last j   */
          {                 /* but must be compared to first */
            if (v1[0][3] > smaxb1[1]) /* value (j = 0)          */
              {
                smaxb1[0] = 0.;
                smaxb1[1] = v1[0][3];
                jsmaxb1[1] = 1;
              }
          }
      }
  }
else
  {
    /*     shape of curve is down-up-down-up-down                   */
    j = 1;
    while (v1[j][3] < v1[j-1][3])    /* going to 1st minimum */
      j++;
    while (v1[j][3] >= v1[j-1][3])
      {                             /* searching for 1st maximum */
        smaxa1[0] = j * 10.;
        smaxa1[1] = v1[j][3];
        jsmaxa1[1] = j + 1;
        j++;
      }
    while (v1[j][3] < v1[j-1][3] && j < 36)
      j++;                          /* going to 2nd minimum */
    if (j-1 == 35)
      {
        smaxb1[0] = 0.;
        smaxb1[1] = v1[0][3];
        jsmaxb1[1] = 1;
      }
    else
      {
        while (v1[j][3] >= v1[j-1][3] && j < 36)
          {                         /* searching for 2nd maximum */
            smaxb1[0] = j * 10.;
            smaxb1[1] = v1[j][3];
            jsmaxb1[1] = j + 1;
            j++;
          }
        if (j-1 == 35)      /* 2nd maximum might be last j   */
          {                 /* but must be compared to first */
            if (v1[0][3] > smaxb1[1]) /* value (j = 0)          */
              {
                smaxb1[0] = 0.;
                smaxb1[1] = v1[0][3];
                jsmaxb1[1] = 1;
              }
          }
      }
  }

ivar[2] = 0;        /* initialization */

/*     min distance (ellipse border points 1 deg apart)             */
/*     ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~   */

if ((i > 1) && (pole[i] == pole[i-1]) && (jdmin1[0] == jdmin1[1]))
   ivar[2] = 1;
ival[0] = jdmin1[1];  ival[1] = 36;  ival[2] = 1;  ival[3] = 0;
val[0] = 350.0;  val[1] = dmin1[0] - 10.0;  val[2] = dmin1[0] + 10.0;
val[3] = 1.0;
ivar[0] = -1;  ivar[1] = 0;
jdmin2[i] = disdir (confin, ((pole[i] == 0) ? tr : optr), v1, dmin1,
           ival, val, ivar, lr[i][0], lr[i][1], v2dis1, dmin2, lr[i][2]);
```

```
/*     min distance (ellipse border points 0.1 deg apart)          */
/*     ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~ */

if ((i > 1) && (pole[i] == pole[i-1]) && (jdmin1[0] == jdmin1[1])
            && (jdmin2[0] == jdmin2[1]))
   ivar[2] = 1;
ival[0] = jdmin2[1];  ival[1] = 10;  ival[2] = 11;  ival[3] = 1;
val[0] = 359.0;  val[1] = v2dis1[jdmin2[1]-2][0] / FPI;
val[2] = v2dis1[jdmin2[1]][0] / FPI;  val[3] = 0.1;
ivar[0] = -1;  ivar[1] = 0;
jdmin3 = disdir (confin, ((pole[i] == 0) ? tr : optr), v2dis1, dmin2,
        ival, val, ivar, lr[i][0], lr[i][1], v3dis1, dmin3, lr[i][2]);

/*     max distance (ellipse border points 1 deg apart)            */
/*     ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~ */

if ((i > 1) && (pole[i] == pole[i-1]) && (jdmax1[0] == jdmax1[1]))
   ivar[2] = 1;
ival[0] = jdmax1[1];  ival[1] = 36;  ival[2] = 1;  ival[3] = 0;
val[0] = 350.0;  val[1] = dmax1[0] - 10.0;  val[2] = dmax1[0] + 10.0;
val[3] = 1.0;
ivar[0] = 1;  ivar[1] = 0;
jdmax2[1] = disdir (confin, ((pole[i] == 0) ? tr : optr), v1, dmax1,
           ival, val, ivar, lr[i][0], lr[i][1], v2dis2, dmax2, lr[i][2]);

/*     max distance (ellipse border points 0.1 deg apart)          */
/*     ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~ */

if ((i > 1) && (pole[i] == pole[i-1]) && (jdmax1[0] == jdmax1[1])
            && (jdmax2[0] == jdmax2[1]))
   ivar[2] = 1;
ival[0] = jdmax2[1];  ival[1] = 10;  ival[2] = 11;  ival[3] = 1;
val[0] = 359.0;  val[1] = v2dis2[jdmax2[1]-2][0] / FPI;
val[2] = v2dis2[jdmax2[1]][0] / FPI;  val[3] = 0.1;
ivar[0] = 1;  ivar[1] = 0;
jdmax3 = disdir (confin, ((pole[i] == 0) ? tr : optr), v2dis2, dmax2,
        ival, val, ivar, lr[i][0], lr[i][1], v3dis2, dmax3, lr[i][2]);

/*     1st max angle (ellipse border points 1 deg apart)           */
/*     ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~ */

if ((i > 1) && (pole[i] == pole[i-1]) && (jsmaxa1[0] == jsmaxa1[1]))
   ivar[2] = 1;
ival[0] = jsmaxa1[1];  ival[1] = 36;  ival[2] = 1;  ival[3] = 0;
val[0] = 350.0;  val[1] = smaxa1[0] - 10.0;  val[2] = smaxa1[0] + 10.0;
val[3] = 1.0;
ivar[0] = 1;  ivar[1] = 1;
jsmaxa2[1] = disdir (confin, ((pole[i] == 0) ? tr : optr), v1, smaxa1,
            ival, val, ivar, lr[i][0], lr[i][1], v2dir1, smaxa2, lr[i][2]);

/*     1st max angle (ellipse border points 0.1 deg apart)         */
/*     ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~ */

if ((i > 1) && (pole[i] == pole[i-1]) && (jsmaxa1[0] == jsmaxa1[1])
            && (jsmaxa2[0] == jsmaxa2[1]))
   ivar[2] = 1;
ival[0] = jsmaxa2[1];  ival[1] = 10;  ival[2] = 11;  ival[3] = 1;
val[0] = 359.0;  val[1] = v2dir1[jsmaxa2[1]-2][0] / FPI;
val[2] = v2dir1[jsmaxa2[1]][0] / FPI;  val[3] = 0.1;
ivar[0] = 1;  ivar[1] = 1;
jsmaxa3 = disdir (confin, ((pole[i] == 0) ? tr : optr), v2dir1, smaxa2,
         ival, val, ivar, lr[i][0],lr[i][1], v3dir1, smaxa3, lr[i][2]);

/*     2nd max angle (ellipse border points 1 deg apart)           */
/*     ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~ */

if ((i > 1) && (pole[i] == pole[i-1]) && (jsmaxb1[0] == jsmaxb1[1]))
   ivar[2] = 1;
ival[0] = jsmaxb1[1];  ival[1] = 36;  ival[2] = 1;  ival[3] = 0;
val[0] = 350.0;  val[1] = smaxb1[0] - 10.0;  val[2] = smaxb1[0] + 10.0;
val[3] = 1.0;
ivar[0] = 1;  ivar[1] = 1;
jsmaxb2[1] = disdir (confin, ((pole[i] == 0) ? tr : optr), v1, smaxb1,
            ival, val, ivar, lr[i][0], lr[i][1], v2dir2, smaxb2, lr[i][2]);

/*     2nd max angle (ellipse border points 0.1 deg apart)         */
/*     ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~ */

if ((i > 1) && (pole[i] == pole[i-1]) && (jsmaxb1[0] == jsmaxb1[1])
            && (jsmaxb2[0] == jsmaxb2[1]))
   ivar[2] = 1;
ival[0] = jsmaxb2[1];  ival[1] = 10;  ival[2] = 11;  ival[3] = 1;
val[0] = 359.0;  val[1] = v2dir2[jsmaxb2[1]-2][0] / FPI;
val[2] = v2dir2[jsmaxb2[1]][0] / FPI;  val[3] = 0.1;
ivar[0] = 1;  ivar[1] = 1;
jsmaxb3 = disdir (confin, ((pole[i] == 0) ? tr : optr), v2dir2, smaxb2,
         ival, val, ivar, lr[i][0], lr[i][1], v3dir2, smaxb3, lr[i][2]);


/*     translation of dist. in deg into speeds in the chosen unit  */
/*     ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~ */

lr[i][7] = fnisos (dmin3[1]*FPI,tr[2]) / FPI * 10000./90. * r1;
lr[i][9] = fnisos (dmax3[1]*FPI,tr[2]) / FPI * 10000./90. * r1;
lr[i][12] = fnisos (v3dir1[jsmaxa3-1][3],tr[2]) / FPI * 10000./90. * r1;
lr[i][14] = fnisos (v3dir2[jsmaxb3-1][3],tr[2]) / FPI * 10000./90. * r1;

/*     corrections of max speed when close to 90 deg distance      */
/*     from pole of rotation (because of ellipse error)            */
/*     ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~ */

if (v3dis2[jdmax3-1][3] >= PI/2)
   lr[i][9] = maxspeed;

/*     directions computations for min & max speeds                */
/*     ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~ */

lr[i][8] = direc (v3dis1[jdmin3-1][1], v3dis1[jdmin3-1][2],
                  lr[i][0], lr[i][1], v3dis1[jdmin3-1][3]);
lr[i][10] = direc (v3dis2[jdmax3-1][1], v3dis2[jdmax3-1][2],
                   lr[i][0], lr[i][1], v3dis2[jdmax3-1][3]);

lr[i][8] /= FPI;
lr[i][10] /= FPI;

/*     directions computations for maximum angles                  */
/*     ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~ */
```

```
 lr[i][11] = direc (v3dir1[jsmaxa3-1][1], v3dir1[jsmaxa3-1][2],
                    lr[i][0], lr[i][1], v3dir1[jsmaxa3-1][3]);
 lr[i][13] = direc (v3dir2[jsmaxb3-1][1], v3dir2[jsmaxb3-1][2],
                    lr[i][0], lr[i][1], v3dir2[jsmaxb3-1][3]);

 lr[i][11] /= FPI;
 lr[i][13] /= FPI;

 /*     correcting min and max directions if on the same side       */
 /*     of mean                                                     */
 /*     ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~     */

 if (sign(lr[i][6]-lr[i][11]) == sign(lr[i][6]-lr[i][13]))
   {                                /* max values on same side of mean */
     if (fabs(fabs(lr[i][6]-lr[i][11]) - v3dir1[jsmaxa3-1][4]/FPI) < 0.01)
       {
         if (lr[i][6] - lr[i][13] < 0)
            lr[i][13] -= 180.;
         else
            lr[i][13] += 180.;
       }
     else
       {
         if (lr[i][6] - lr[i][11] < 0)
            lr[i][11] -= 180.;
         else
            lr[i][11] += 180.;
       }
   }

/*     putting the min direction in col 11 and the max in col 13    */
/*     ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~     */

 if (lr[i][11] > lr[i][13])
   {
     temp = lr[i][11];
     lr[i][11] = lr[i][13];
     lr[i][13] = temp;
     temp = lr[i][14];
     lr[i][14] = lr[i][12];
     lr[i][12] = temp;
   }


 /*     putting the results aside to be compared with next loop      */
 /*     ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~     */

 jdmin1[0] = jdmin1[1];
 jdmin2[0] = jdmin2[1];
 jdmax1[0] = jdmax1[1];
 jdmax2[0] = jdmax2[1];
 jsmaxa1[0] = jsmaxa1[1];
 jsmaxa2[0] = jsmaxa2[1];
 jsmaxb1[0] = jsmaxb1[1];
 jsmaxb2[0] = jsmaxb2[1];

}
```

```
/*     PRINTING PART                                                */
/*     =============                                                */


do { printf ("\n\nPRINTER READY (Y/N) : %c", BEEP);
     while ( (ans = getchar()) == '\n')
          continue;
   } while ( (ans != 'Y') && (ans != 'y') );


/*     Preparing the printer                                        */
/*     ---------------------                                        */

fp = fopen("PRN","w");
fprintf (fp,"%cn %cL009 %c!\n", ESC, ESC, ESC);


/*     Printing the title and the subtitle                          */
/*     -----------------------------------                          */

na = (44 - strlen(&name[a-1][0]) ) / 2;
for (nb=0; nb<na; nb++)
    fprintf (fp," ");
fprintf (fp,"%cXPLATE COUPLE : %s\n", ESC, &name[a-1][0]);
fprintf (fp,"%cY%c\"%cL013 \n\n\n\n", ESC, ESC, ESC);
fprintf (fp,"Confidence interval is %s. \n", &ci[confin-1][0]);
fprintf (fp,"%cq %cL013 \n\n", ESC, ESC);


/*     Printing the column headings                                 */
/*     ----------------------------                                 */

fprintf (fp,"  LAT     LONG    DPOLE  DIST DEACH  SPEED   DIR   SPDMN");
fprintf (fp," VSPDMN  SPDMX VSPDMX  DIRMN SDIRMN  DIRMX SDIRMX \n");
fprintf (fp,"  deg      deg     deg    km    km   %5s   deg   %5s",
          &spd[unit-1][0], &spd[unit-1][0]);
fprintf (fp,"   deg   %5s   deg    deg   %5s   deg   %5s\n\n",
          &spd[unit-1][0], &spd[unit-1][0], &spd[unit-1][0]);


/*     Printing the data and results                                */
/*     -----------------------------                                */

/* 1 : lat of the point ; 2 : long of the point ; 3 : distance to the pole
 * 4 : distance from the first point ; 5 : distance from the preceding pnt.
 * 6 : speed of motion ; 7 : direction of motion ;
 * 8 : min speed within conf. interval ; 9 : its associated direction
 * 10 : max speed within conf. interval ; 11 : its associated direction
 * 12 : min direction within conf. inter. ; 13 : its associated speed
 * 14 : max direction within conf. inter. ; 15 : its associated speed   */

for (i=0; i<nl; i++)
  {
    fprintf (fp,"%7.3f %8.3f %6.2f %5.0f %5.0f", l[i][0], l[i][1],
                lr[i][2]/FPI, lr[i][3], lr[i][4]);
    for (j=5; j<15; j++)
```

```
            fprintf (fp," %6.2f", lr[i][j]);
        fprintf (fp,"\n");
      }


    /*     Looking for the existence of a point within the error ellipse    */
    /*     ------------------------------------------------------------     */

    for (i=0; i<nl; i++)
        if ( lr[i][9] == 0.0)
          {
            /*     printing explanations if the maximum speed is zero       */
            /*     ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~       */

            fprintf (fp,"\n%cE\n", ESC);
            fprintf (fp,"\nThe zeros in columns 8 through 15 mean that the ");
            fprintf (fp,"point is inside the pole of\nrotation error ellipse.\n");

            break;    /* end the test on existence of a point inside ellipse */
          }

    fprintf (fp,"\n");


    /*     Writing the data and the results into a file                     */
    /*     --------------------------------------------                     */

    fp = fopen (fnamou,"w");

    fprintf (fp,"PLATE COUPLE : %s\n", &name[a-1][0]);
    fprintf (fp,"Confidence interval is %s.\n", &ci[confin-1][0]);

    fprintf (fp,"  LAT     LONG    DPOLE  DIST DEACH   SPEED   DIR     SPDMN");
    fprintf (fp,"  VSPDMN   SPDMX  VSPDMX  DIRMN   SDIRMN  DIRMX   SDIRMX\n");
    fprintf (fp,"  deg      deg     deg    km    km    %5s   deg     %5s",
                &spd[unit-1][0], &spd[unit-1][0]);
    fprintf (fp,"    deg    %5s    deg    deg     %5s   deg     %5s\n",
                &spd[unit-1][0], &spd[unit-1][0], &spd[unit-1][0]);

    for (i=0; i<nl; i++)
      {
        fprintf (fp,"%7.3f %8.3f %6.3f %5.0f %5.0f", l[i][0], l[i][1],
                    lr[i][2]/FPI, lr[i][3], lr[i][4]);
        for (j=5; j<15; j++)
            fprintf (fp," %7.4f", lr[i][j]);
        fprintf (fp,"\n");
      }

      fclose (fp);

  }



/*****************************************************************************/
/*                                                                           */
/*     DEFINITION OF FUNCTIONS : SPHERICAL TRIGONOMETRY                      */
/*     ================================================                      */
/*****************************************************************************/
```

```
/*===========================================================================*/
/*                                                                           */
/*     Arc-cosine of angles with test on the value of the cosine             */
/*     ---------------------------------------------------------             */

double fnacos (double x)
  {
    double returnvalue;

    if ( (x < 1.0) && (x > -1.0) )
       returnvalue = acos (x);
    else
      {
        if (x >= 1.0)
           returnvalue = 0.0;
        else
           if (x <= -1.0)
              returnvalue = PI;
      }

    return (returnvalue);
  }


/*===========================================================================*/
/*                                                                           */
/*     Arc-sine of angles with test on the value of the sine                 */
/*     -----------------------------------------------------                 */

double fnasin (double x)
  {
    double returnvalue;

    if ( (x <1.0) && (x > -1.0) )
       returnvalue = asin(x);
    else
      {
        if (x >= 1.0)
           returnvalue = PI/2;
        else
           if (x <= -1.0)
              returnvalue = -PI/2;
      }

    return (returnvalue);
  }


/*===========================================================================*/
/*                                                                           */
/*     Distance between two points on the Earth (in radians) :               */
/*     X = lat of point 1, Y = long of point 1,                              */
/*     Z = lat of point 2, W = long of point 2                               */
/*     with X, Y, Z & W in radians                                           */
/*     ---------------------------------------------------                   */

double fndis (double x, double y, double z, double w)
```

```
  {
    return (fnacos (sin(x)*sin(z) + cos(x)*cos(z)*cos(w-y)) );
  }


/*=============================================================*/
/*                                                             */
/*     Half length (in radians) of the side facing angle Y in an */
/*     isosceles triangle with other sides of equal length X   */
/*     (X & Y are in radians)                                  */
/*     ------------------------------------------------------  */

double fnisos (double x, double y)
  {
    return ( fnasin (sin(fabs(x)) * sin(fabs(y))) );
  }


/*=============================================================*/
/*                                                             */
/*     For any triangle, value (in radians) of the angle facing the */
/*     side with length X, the other side being of lengths Y and Z */
/*     (X, Y & Z are given in radians)                         */
/*     ------------------------------------------------------  */

double fndir (double x, double y, double z)
  {
    return ( fnacos ((cos(fabs(x))-cos(fabs(y))*cos(fabs(z))) /
                     (sin(fabs(y))*sin(fabs(z)))) );
  }


/*=============================================================*/
/*                                                             */
/*     For any triangle, length (in radians) of the side facing angle Z, */
/*     with other sides of lengths X and Y (X, Y & Z in radians) */
/*     ------------------------------------------------------  */

double fnside (double x, double y, double z)
  {
    return (fnacos (cos(fabs(x))*cos(fabs(y)) +
                    sin(fabs(x))*sin(fabs(y))*cos(fabs(z))) );
  }


/*=============================================================*/
/*                                                             */
/*     Distance ellipse center-ellipse border as a function of the angle */
/*     Y, axis 1 length X, axis 2 length Z (X, Y & Z are in radians) */
/*     ------------------------------------------------------  */

double fnelip (double x, double y, double z)
  {
    return (x * z / sqrt (z*z*sin(y)*sin(y)+x*x*cos(y)*cos(y)) );
  }



/*=============================================================*/
```

```
/*                                                             */
/*     SUBROUTINES                                             */
/*     ===========                                             */
/*                                                             */
/*=============================================================*/

/*=============================================================*/
/*                                                             */
/*     Computation of the direction of plate motion at one data point */
/*     given the rotation pole                                 */
/*     ------------------------------------------------------  */

/*     RLAT & RLONG : lat & long of the rotation pole          */
/*     PLAT & PLONG : lat & long of the data point             */
/*     RDIS : distance between the data point and the rotation pole */
/*     RDIR : direction                                        */

double direc (double rlat, double rlong, double plat, double plong, double rdis)
  {
    double rdir, aK, aK1, deltalong;

    /*     angle made by the 3 points : rotation pole - point - North pole */
    /*     ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~ */

    aK1 = fndir (PI/2-fabs(rlat), rdis, PI/2-sign(rlat)*plat);

    /*     to have the direction of plate motion from 0 to PI      */
    /*     ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~ */

    deltalong = rlong-plong;
    if (fabs(deltalong) > PI)
      deltalong = -(deltalong - sign(deltalong)*PI);

    if (aK1 <= PI/2)
      {
        if (deltalong*rlat >= 0)
          rdir = aK1 + PI/2;
        else
          rdir = PI/2 - aK1;
      }
    else
      {
        if (deltalong*rlat >= 0)
          rdir = aK1 - PI/2;
        else
          rdir = PI - (aK1 - PI/2);
      }

    return (rdir);
  }     /* direc */


/*=============================================================*/
/*                                                             */
/*     Computation of the position (lat & long) of one point on the */
/*     ellipse border given the angle from an origin axis on the ellipse */
/*     ------------------------------------------------------  */
```

```
/*     the origin axis is the first main axis of the ellipse from N          */
/*     toward E                                                              */
/*     the angle is given clockwise from the axis                            */
/*     SI : angle                                                            */
/*     POS : lat & long of the point on the ellipse border, and distance     */
/*           from center to border                                           */
/*     TR : parameter table of nearest ellipse                               */
/*     CONFIN : confidence interval : 1 for 70%, 2 for 95%                   */

void posit (double si, int confin, double tr[], double pos[])
  {
    double ang, delta;

    /*     length of the radius of the ellipse for the given angle          */
    /*     ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~ */

    pos[2] = fnelip (confin*tr[5], si, confin*tr[4]);

    /*     latitude of the point on the ellipse border                       */
    /*     ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~ */

    if (tr[0] < 0.0)
       ang = tr[3] - PI;
    else
       ang = tr[3];

    pos[0] = sign (tr[0]) * (PI/2 - fnside (pos[2],PI/2-fabs(tr[0]),fabs(si+ang)));

    /*     longitude of the point on the ellipse border                      */
    /*     ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~ */

    delta = fndir (pos[2],PI/2-fabs(tr[0]),PI/2-fabs(pos[0]));
    if ( fmod (tr[3]+si,2*PI) <= PI )
       pos[1] = tr[1] + delta;
    else
       pos[1] = tr[1] - delta;

  }  /* posit */


/*=========================================================================*/
/*                                                                         */
/*     Computation of min or max distance or max angle for points          */
/*     on a given portion of ellipse border to data points                 */
/*     ----------------------------------------------------------          */

/*     JJ : variable returned                                              */
/*     TR : parameter table of nearest ellipse                             */
/*     PLAT & PLONG : point latitude and longitude                         */
/*     TAB1 & EXTR1 : results of preceding pass                            */
/*     TAB2 & EXTR2 : results of present pass                              */
/*     MEANDIS : distance point-center of ellipse                          */
/*     IVAL[0] : line number of angle with best value from preceding pass  */
/*     IVAL[1] : line number of maximum angle of interval to copy if       */
/*               ival[0] gives a value of 0 degree                         */
/*     IVAL[2] : line number of minimum angle of interval to copy if       */
/*               ival[0] gives a value of VAL[0]                           */
/*     IVAL[3] : pass number (0 for 1 deg interval, 1 for 0.1 deg interval */
/*     VAL[0] : 350 (ival[3]=0) or 359 (ival[3]=1)                         */
/*     VAL[1] : value of minimum angle of interval to search into          */
/*     VAL[2] : value of maximum angle of interval to search into          */
/*     VAL[3] : angle between individual points on ellipse border (1 or .1)*/
/*     IVAR[0] : -1 = search for a minimum, 1 = search for a maximum        */
/*     IVAR[1] : 0 = search for distance, 1 = search for an angle           */
/*     IVAR[2] : 0 = complete computations, 1 = partial computations       */
/*     CONFIN : confidence interval : 1 for 70%, 2 for 95%                 */

int disdir (confin, tr, tab1, extr1, ival, val, ivar, plat, plong, tab2, extr2, meandis)

int confin, ival[], ivar[];
double tr[], tab1[][6], extr1[], val[], plat, plong, tab2[][6], extr2[], meandis;

  {
    int js, jj;
    double sj1, tab1[3];

    /*     filling up some of the cells of the output table                  */
    /*     ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~ */

    if (ivar[2] == 0)
      {
        tab2[10][0] = extr1[0] * FPI;
        tab2[10][1] = tab1[ival[0]-1][ival[3]];
        tab2[10][2] = tab1[ival[0]-1][1+ival[3]];
        tab2[10][5] = tab1[ival[0]-1][5];

        if (extr1[0] == 0.0)
          {
            tab2[0][0] = val[0] * FPI;
            tab2[0][1] = tab1[ival[1]-1][ival[3]];
            tab2[0][2] = tab1[ival[1]-1][1+ival[3]];
            tab2[0][5] = tab1[ival[1]-1][5];
          }
        else
          {
            tab2[0][0] = val[1] * FPI;
            tab2[0][1] = tab1[ival[0]-2][ival[3]];
            tab2[0][2] = tab1[ival[0]-2][1+ival[3]];
            tab2[0][5] = tab1[ival[0]-2][5];
          }

        if (extr1[0] == val[0])
          {
            tab2[20][0] = 0.0;
            tab2[20][1] = tab1[ival[2]-1][ival[3]];
            tab2[20][2] = tab1[ival[2]-1][1+ival[3]];
            tab2[20][5] = tab1[ival[2]-1][5];
          }
        else
          {
            tab2[20][0] = val[2] * FPI;
            tab2[20][1] = tab1[ival[0]][ival[3]];
            tab2[20][2] = tab1[ival[0]][1+ival[3]];
            tab2[20][5] = tab1[ival[0]][5];
          }
      }
```

```
/*     position of the other points on the ellipse border              */
/*     ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~ */

sj1 = tab2[0][0];

for (js=1; js<10; js++)
  {
    tab2[js][0] = fmod (sj1+js*val[3]*FPI, 2*PI);
    posit (tab2[js][0],confin,tr,tabl);
    tab2[js][1] = tabl[0];
    tab2[js][2] = tabl[1];
    tab2[js][5] = tabl[2];
  }

if ((sj1/FPI) == val[0])
   sj1 = -val[3] * 10 * FPI;

for (js=11; js<20; js++)
  {
    tab2[js][0] = fmod (sj1+js*val[3]*FPI,2*PI);
    posit (tab2[js][0],confin,tr,tabl);
    tab2[js][1] = tabl[0];
    tab2[js][2] = tabl[1];
    tab2[js][5] = tabl[2];
  }

/*     computing distance between points on ellipse and data points    */
/*     ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~ */

for (js=0; js<21; js++)
    tab2[js][3] = fndis (plat,plong,tab2[js][1],tab2[js][2]);

/*     angle in deg center of ellipse-point-border                     */
/*     ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~ */

/*     test on distance vs. direction computation                      */

if (ivar[1] != 0)
  {
    for (js=0; js<21; js++)
        tab2[js][4] = fndir (tab2[js][5],meandis,tab2[js][3]);
  }

/*     search for min (or max) distance (or direction)                 */
/*     ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~ */

extr2[0] = tab2[0][0];
extr2[1] = tab2[0][3+ivar[1]];
jj = 0;
for (js=1; js<21; js++)
  {
    if ((tab2[js][3+ivar[1]]*ivar[0]) > extr2[1]*ivar[0])
      {
        extr2[0] = tab2[js][0];
        extr2[1] = tab2[js][3+ivar[1]];
        jj = js;
      }
  }
  extr2[0] = extr2[0] / FPI;
  extr2[1] = extr2[1] / FPI;
  ivar[2] = 0;

  return (jj + 1);
}


/*=========================================================================*/
/*                                                                         */
/*     CLEARSCREEN (VOID) : ANSI sequence ESC [2J                          */
/*     ------------------------------------------                          */

void clearscreen (void)
  {
    printf ("%c[2J",ESC);
  }


/*=========================================================================*/
/*                                                                         */
/*     LOCATE (ROW,COLUMN) : ANSI sequence ESC [ row ; column H            */
/*     ------------------------------------------------------              */

void locate (int row, int column)
  {
    printf ("%c[%d;%dH",ESC,row,column);
  }
```

C

PROGRAM PLATES

This program computes the mean speed and direction of plate motion, the lower and upper values (confidence intervals : 7013r 95%) of speed (with the corresponding direction) and direction (with the corresponding speed), using Model RM2 (Minster & Jordan, 1978).

The data must be located in a separate file having at the beginning, on the first line the number of points in %3d format (up to 300 points are allowed). Then there are the latitude and longitude of each point in decimal degrees (format : %6.3f %7.3f) : 3 points per line.

Do you want to use an old data file (1)
or to create a new one (2) : 1

Name of your old data file : FMATPNT.DAT

★

READING THE DATA

★

LIST OF THE PLATE COUPLES :

1 : North America - Pacific	2 : Cocos - Pacific
3 : Nazca - Pacific	4 : Eurasia - Pacific
5 : India - Pacific	6 : Antarctica - Pacific
7 : Cocos - North America	8 : Africa - North America
9 : Eurasia - North America	10 : North America - Caribbean
11 : Cocos - Caribbean	12 : Nazca - Caribbean
13 : Cocos - Nazca	14 : North America - South America
15 : Caribbean - South America	16 : Nazca - South America
17 : Africa - South America	18 : Antarctica - South America
19 : India - Africa	20 : Arabia - Africa
21 : Africa - Eurasia	22 : India - Eurasia
23 : Arabia - Eurasia	24 : India - Arabia
25 : Nazca - Antarctica	26 : Africa - Antarctica
27 : India - Antarctica	

Give the number of the plate couple you have chosen : 7

What confidence interval do you want : 70% (1) or 95% (2) : 1

Do you want the results in cm/yr (1) or km/my (2) : 1

Name of the file to put data + results : FMATPNT1.DAT

★

COMPUTING THE MAIN RESULTS

★

ERROR ELLIPSE COMPUTATIONS : Point No. 1

★

ERROR ELLIPSE COMPUTATIONS : Point No. 2

.
.
.
.
.

★

ERROR ELLIPSE COMPUTATIONS : Point No. 29

★

ERROR ELLIPSE COMPUTATIONS : Point No. 30

PRINTER READY (Y/N) : Y

D

PLATE COUPLE : Cocos - North America

Confidence interval is 70%.

LAT deg	LONG deg	DPOLE deg	DIST km	DEACH km	SPEED cm/yr	DIR deg	SPDMN cm/yr	VSPDMN deg	SPDMX cm/yr	VSPDMX deg	DIRMN deg	SDIRMN cm/yr	DIRMX deg	SDIRMX cm/yr
18.678	-104.963	18.54	0	0	5.26	39.93	4.77	41.28	5.74	38.75	36.57	5.41	43.53	5.05
18.373	-104.555	19.03	55	55	5.39	40.02	4.91	41.33	5.88	38.86	36.74	5.54	43.52	5.19
17.767	-103.733	20.02	165	110	5.66	40.16	5.18	41.48	6.14	39.06	37.04	5.81	43.49	5.46
17.372	-102.778	20.97	273	110	5.92	39.70	5.44	40.82	6.40	38.69	36.73	6.06	42.84	5.73
16.965	-101.892	21.88	376	104	6.17	39.39	5.69	40.43	6.64	38.45	36.55	6.30	42.38	5.98
16.645	-101.195	22.60	458	82	6.36	39.16	5.88	40.14	6.83	38.27	36.41	6.49	42.05	6.17
16.340	-100.167	23.56	571	115	6.61	38.56	6.14	39.45	7.08	37.76	35.94	6.74	41.30	6.44
16.038	-99.358	24.36	663	93	6.82	38.23	6.35	39.06	7.29	37.48	35.70	6.94	40.87	6.65
15.773	-98.655	25.05	743	81	7.01	37.97	6.53	38.74	7.47	37.26	35.51	7.12	40.53	6.84
15.495	-97.793	25.88	839	97	7.22	37.60	6.75	38.32	7.69	36.94	35.22	7.33	40.06	7.06
15.293	-97.230	26.43	904	64	7.36	37.40	6.89	38.08	7.83	36.77	35.08	7.47	39.80	7.21
15.218	-95.813	27.57	1045	152	7.66	36.28	7.19	36.85	8.12	35.75	34.07	7.75	38.56	7.52
15.007	-95.333	28.07	1102	57	7.78	36.21	7.32	36.76	8.24	35.70	34.04	7.87	38.44	7.65
14.702	-94.843	28.63	1164	63	7.93	36.27	7.46	36.81	8.38	35.77	34.14	8.02	38.46	7.80
14.168	-93.995	29.61	1271	109	8.17	36.38	7.71	36.90	8.63	35.90	34.32	8.26	38.50	8.05
13.817	-93.317	30.35	1354	83	8.36	36.36	7.90	36.85	8.81	35.90	34.34	8.44	38.43	8.23
13.483	-92.667	31.06	1433	79	8.53	36.34	8.08	36.81	8.98	35.90	34.36	8.62	38.36	8.41
12.812	-91.240	32.57	1605	171	8.91	36.21	8.46	36.64	9.34	35.80	34.32	8.99	38.13	8.79
12.505	-90.542	33.30	1688	83	9.08	36.12	8.64	36.54	9.52	35.74	34.28	9.16	38.01	8.97
12.167	-89.583	34.26	1798	111	9.31	35.91	8.87	36.29	9.74	35.56	34.12	9.39	37.74	9.21
11.777	-88.647	35.23	1908	111	9.54	35.79	9.11	36.15	9.97	35.46	34.04	9.61	37.57	9.44
11.555	-88.167	35.74	1966	58	9.66	35.75	9.23	36.10	10.09	35.43	34.03	9.73	37.51	9.57
10.833	-87.218	36.92	2094	131	9.94	35.99	9.51	36.33	10.35	35.68	34.31	10.00	37.70	9.84
9.815	-86.333	38.22	2232	149	10.24	36.55	9.82	36.89	10.64	36.22	34.91	10.31	38.21	10.14
9.150	-85.500	39.28	2348	117	10.47	36.74	10.06	37.08	10.88	36.41	35.14	10.54	38.36	10.38
8.655	-84.425	40.43	2478	130	10.73	36.62	10.32	36.94	11.12	36.32	35.06	10.79	38.20	10.64
8.355	-83.868	41.05	2547	70	10.86	36.60	10.46	36.91	11.26	36.31	35.06	10.93	38.16	10.77
8.178	-83.377	41.54	2604	57	10.97	36.50	10.57	36.80	11.36	36.22	34.98	11.03	38.05	10.88
7.833	-82.992	42.06	2660	57	11.08	36.61	10.68	36.91	11.47	36.33	35.10	11.15	38.14	11.00
7.317	-82.833	42.49	2703	60	11.17	36.98	10.78	37.30	11.56	36.69	35.49	11.24	38.51	11.09

FICHIER DE DONNEES : FMATPNT1.PRN

PLATE COUPLE : Cocos - North America
Confidence interval is 70%.

LAT deg	LONG deg	DPOLE deg	DIST km	DEACH km	SPEED cm/yr	DIR deg	SPDMN cm/yr	VSPDMN deg	SPDMX cm/yr	VSPDMX deg	DIRMN deg	SDIRMN cm/yr	DIRMX deg	SDIRMX cm/yr
18.678	-104.963	18.539	0	0	5.2598	39.9337	4.7726	41.2825	5.7442	38.7474	36.5726	5.4057	43.5294	5.0529
18.373	-104.555	19.031	55	55	5.3944	40.0197	4.9084	41.3264	5.8774	38.8643	36.7413	5.5399	43.5200	5.1891
17.767	-103.733	20.019	165	110	5.6633	40.1628	5.1798	41.3997	6.1434	39.0586	37.0389	5.8087	43.4855	5.4611
17.372	-102.778	20.971	273	110	5.9206	39.6980	5.4384	40.8200	6.3990	38.6856	36.7275	6.0606	42.8422	5.7261
16.965	-101.892	21.883	376	104	6.1658	39.3860	5.6852	40.4319	6.6419	38.4503	36.5457	6.3013	42.3803	5.9775
16.645	-101.195	22.602	458	82	6.3579	39.1605	5.8788	40.1405	6.8323	38.2700	36.4147	6.4905	42.0467	6.1743
16.340	-100.167	23.561	571	115	6.6126	38.5584	6.1351	39.4506	7.0849	37.7555	35.9371	6.7374	41.3016	6.4381
16.038	-99.358	24.358	663	93	6.8228	38.2307	6.3471	39.0606	7.2929	37.4790	35.7007	6.9430	40.8704	6.6541
15.773	-98.655	25.053	743	81	7.0053	37.9698	6.5313	38.7448	7.4733	37.2567	35.5139	7.1213	40.5260	6.8420
15.495	-97.793	25.880	839	97	7.2207	37.5956	6.7488	38.3176	7.6863	36.9408	35.2235	7.3311	40.0572	7.0639
15.293	-97.230	26.433	904	64	7.3642	37.3986	6.8939	38.0840	7.8279	36.7676	35.0784	7.4713	39.8021	7.2118
15.218	-95.813	27.572	1045	152	7.6572	36.2806	7.1893	36.8534	8.1180	35.7536	34.0731	7.7496	38.5552	7.5209
15.007	-95.333	28.071	1102	57	7.7845	36.2091	7.3184	36.7631	8.2434	35.6977	34.0401	7.8748	38.4414	7.6512
14.702	-94.843	28.633	1164	63	7.9274	36.2717	7.4635	36.8099	8.3838	35.7734	34.1422	8.0169	38.4615	7.7955
14.168	-93.995	29.611	1271	109	8.1741	36.3820	7.7143	36.9019	8.6261	35.9048	34.3174	8.2626	38.5024	8.0455
13.817	-93.317	30.349	1354	83	8.3586	36.3596	7.9019	36.8528	8.8074	35.8989	34.3426	8.4448	38.4285	8.2332
13.483	-92.667	31.056	1433	79	8.5341	36.3356	8.0803	36.8111	8.9797	35.8958	34.3620	8.6185	38.3577	8.4118
12.812	-91.240	32.573	1605	171	8.9064	36.2070	8.4592	36.6407	9.3449	35.8029	34.3208	8.9857	38.1348	8.7915
12.505	-90.542	33.304	1688	83	9.0835	36.1222	8.6396	36.5358	9.5186	35.7356	34.2753	9.1600	38.0077	8.9724
12.167	-89.583	34.260	1798	111	9.3131	35.91[illegible]	8.8734	36.2914	9.7436	35.5560	34.1153	9.3851	37.7436	9.2081
11.777	-88.647	35.231	1908	111	9.5434	35.78[illegible]	9.1083	36.1477	9.9689	35.4559	34.0385	9.6115	37.5706	9.4433
11.555	-88.167	35.742	1966	58	9.6636	35.7552	9.2310	36.0979	10.0865	35.4273	34.0258	9.7301	37.5104	9.5658
10.833	-87.218	36.920	2094	131	9.9374	35.9912	9.5113	36.3294	10.3538	35.6754	34.3103	10.0043	37.7000	9.8413
9.815	-86.333	38.224	2232	149	10.2359	36.5472	9.8173	36.8909	10.6444	36.2202	34.9103	10.3065	38.2112	10.1381
9.150	-85.500	39.280	2348	117	10.4738	36.7382	10.0612	37.0772	10.8760	36.4149	35.1375	10.5442	38.3647	10.3771
8.655	-84.425	40.427	2478	130	10.7281	36.6202	10.3218	36.9368	11.1237	36.3168	35.0602	10.7949	38.2034	10.6365
8.355	-83.868	41.049	2547	70	10.8640	36.6007	10.4613	36.9115	11.2559	36.3071	35.0611	10.9294	38.1624	10.7745
8.178	-83.377	41.544	2604	57	10.9716	36.5037	10.5716	36.8002	11.3606	36.2185	34.9809	11.0348	38.0476	10.8849
7.833	-82.992	42.056	2660	57	11.0817	36.6133	10.6849	36.9103	11.4675	36.3273	35.1049	11.1454	38.1423	10.9951
7.317	-82.833	42.491	2703	60	11.1747	36.9849	10.7809	37.2975	11.5574	36.6888	35.4850	11.2419	38.5060	11.0852

CONVERSIONS D'ECHELLES FRACTIONNAIRES
EN POUCES-PAR-DEGRE DE LONGITUDE
ET INVERSEMENT

A - Listing du programme `SCALE'

B - Copie d'écran d'un premier exemple d'exécution du programme `SCALE'

C - Sortie sur traceur du résultat de cette première exécution

D - Copie d'écran d'un second exemple d'exécution du programme `SCALE'

E - Sortie sur traceur du résultat de cette seconde exécution

```
10 REM  ================================================================
20 REM
30 REM                          PROGRAM  SCALE'
40 REM
50 REM                                        JACQUELINE ROUMP, AUG. 1984
60 REM  ================================================================
70 PAGE
80 INIT
90 SET DEGREES
100 REM    PRESENTATION OF THE PROGRAM
110 REM    ===========================
120 CHARSIZE 4
130 PRINT "                    SCALE"
140 PRINT
150 PRINT
160 CHARSIZE 3
170 PRINT "PROGRAM TO COMPUTE SCALE IN MERCATOR PROJECTION FROM"
180 PRINT "INCHES PER DEGREE TO 1/X AND FROM 1/X TO INCH/DEG"
190 PRINT
200 PRINT "ELLIPSOID : WGS 72"
210 PRINT "ALL NUMBERS MUST BE >0"
220 PRINT
230 REM  ----------------------------------------------------------------
240 REM
250 REM    QUESTIONS AND VERIFICATIONS
260 REM    ===========================
270 PRINT "TITLE : ";
280 INPUT T$
290 PRINT
300 REM
310 REM    LATITUDES
320 REM    ---------
330 PRINT "BEGIN LATITUDE (DD.MM) : ";
340 INPUT A
350 C9=A
360 C$="Y"
370 IF A=>0 THEN 400
380 GOSUB 1660
390 GOSUB 470
400 A0=A-INT(A)
410 IF A0<0.6 THEN 440
420 GOSUB 1700
430 GOSUB 470
440 IF A<=90 THEN 490
450 PRINT "LATITUDE >= 90 DEGREES"
460 GOSUB 1620
470 IF B$=C$ THEN 280
480 GO TO 1880
490 PRINT "END LATITUDE (DD.MM) : ";
500 INPUT A1
510 IF A1=>0 THEN 540
520 GOSUB 1660
530 GOSUB 610
540 A9=A1-INT(A1)
550 IF A9<0.6 THEN 580
560 GOSUB 1700
570 GOSUB 610
580 IF A1<=90 THEN 630
590 PRINT "LATITUDE > 90 DEGREES"
600 GOSUB 1620
610 IF B$=C$ THEN 490
620 GO TO 1880
630 IF A=A1 THEN 850
640 PRINT "INCREMENT (DD.MM) : ";
650 INPUT A2
660 IF A2>0 THEN 680
670 A2=-A2
680 A5=A2-INT(A2)
690 IF A5<0.6 THEN 720
700 GOSUB 1700
710 GOSUB 790
720 B=(A-INT(A))*100/60+INT(A)
730 B1=(A1-INT(A1))*100/60+INT(A1)
740 B2=(A2-INT(A2))*100/60+INT(A2)
750 C0=ABS(B1-B)-B2
760 IF C0=>0 THEN 810
770 PRINT "INCREMENT > LATITUDE DIFFERENCE"
780 GOSUB 1620
790 IF B$=C$ THEN 640
800 GO TO 1880
810 PRINT
820 REM
830 REM      SCALE
840 REM      -----
850 PRINT "SCALE : ";
860 INPUT A3
870 PRINT "TYPE OF SCALE ? (IN/DEG OR 1/) : ";
880 INPUT E$
890 IF A-A1=>0 THEN 910
900 A2=-A2
910 PRINT "AT LATITUDE (DD.MM) ? ";
920 INPUT L1
930 L1=INT(L1)+(L1-INT(L1))*100/60
940 REM ----------------------------------------------------------------
950 REM
960 REM      COMPUTATION PART
970 REM      ================
980 REM EQUATORIAL RADIUS = 6378.135 KM
990 REM LENGTH OF A DEGREE AT THE EQUATOR
1000 F3=0.99333056822
1010 F=6378.135*2*PI/360
1020 REM ONE INCH =2.54 CM, SO 1 INCH/DEGREE = 1/ :
1030 F2=F*100000/2.54
1040 IF L1=0 THEN 1070
1050 S1=A3/COS(ATN(F3*TAN(L1)))
1060 GO TO 1080
1070 S1=A3
1080 IF ABS(A-A1)>0 THEN 1190
1090 REM
1100 REM      ONE SCALE VALUE
1110 REM      ---------------
1120 B=(A-INT(A))*100/60+INT(A)
1130 X=F2*COS(ATN(F3*TAN(B)))/S1
1140 PRINT "LAT (DD.MM) = ";A,"SCALE = ";X
1150 GO TO 6860
1160 REM
1170 REM      MORE THAN ONE SCALE VALUE
1180 REM      -------------------------
1190 J=ABS(INT((B-B1+1.0E-5)/B2))+1
1200 DIM T(J,8)
```

```
1210 L=1
1220 REM ----------
1230 FOR B9=B TO B1 STEP B2
1240 T(L,7)=B9
1250 X=F2*COS(ATN(F3*TAN(B9)))/S1
1260 T(L,8)=X
1270 T(L,1)=INT(B9+1.0E-5)
1280 B8=(B9+1.0E-5-T(L,1))*60
1290 T(L,3)=INT(B8+1.0E-5)
1300 V8=(B8+1.0E-5-T(L,3))*100
1310 IF V8-INT(V8)>0.5 THEN 1340
1320 T(L,4)=INT(V8)
1330 GO TO 1350
1340 T(L,4)=INT(V8)+1
1350 V8=(B9-T(L,1))*1000
1360 IF V8-INT(V8)>0.5 THEN 1390
1370 T(L,2)=INT(V8)
1380 GO TO 1400
1390 T(L,2)=INT(V8)+1
1400 T(L,5)=INT(X)
1410 V7=X-INT(X)
1420 IF E$="IN/DEG" THEN 1490
1430 IF V7<0.9995 THEN 1450
1440 T(L,5)=T(L,5)+1
1450 V8=V7*1000
1460 GO TO 1500
1470 IF V7<0.95 THEN 1420
1480 T(L,5)=T(L,5)+1
1490 V8=V7*10
1500 IF V8-INT(V8)>0.5 THEN 1530
1510 T(L,6)=INT(V8)
1520 GO TO 1540
1530 T(L,6)=INT(V8)+1
1540 L=L+1
1550 NEXT B9
1560 REM ----------
1570 GO TO 1770
1580 REM --------------------
1590 REM
1600 REM     SUBROUTINES FOR ERRORS
1610 REM     ======================
1620 PRINT "TRY AGAIN (Y/N)"
1630 INPUT B$
1640 RETURN
1650 REM ----------
1660 PRINT "NUMBER < 0"
1670 GOSUB 1620
1680 RETURN
1690 REM --------------------
1700 PRINT ".MM>=.60"
1710 GOSUB 1620
1720 RETURN
1730 REM --------------------
1740 REM
1750 REM     QUESTIONS FOR THE GRAPH
1760 REM     =======================
1770 PRINT "BEGIN LAT (DD.MM) = ";C9;"  END LAT (DD.MM) = ";A1
1780 M=T(1,7) MIN B9
1790 N=(T(1,7) MAX B9)-B2
1800 PRINT "TICKS AND LABELS EVERY ? TICKS ON LAT AXIS"
1810 INPUT R9,V9
1820 R=(R9-INT(R9))*100/60+INT(R9)
1830 V=(V9-INT(V9))*100/60+INT(V9)
1840 PRINT "NUMBER OF DECIMAL CHARACTERS TO PRINT THE LATITUDES : ";
1850 INPUT V1
1860 PRINT
1870 PRINT "SCALE FROM ";T(1,8);" TO ";X
1880 PRINT "SMALLER SCALE, LARGER SCALE"
1890 INPUT O,P
1900 IF O<P THEN 1950
1910 PRINT "LARGER SCALE<SMALLER SCALE"
1920 GOSUB 1620
1930 IF B$=C$ THEN 1870
1940 GO TO 6860
1950 PRINT "TICKS AND LABELS EVERY ? TICKS ON SCALE AXIS"
1960 INPUT S,D
1970 PRINT "NUMBER OF DECIMAL CHARACTERS TO PRINT THE SCALE : ";
1980 INPUT V2
1990 MO=N-M
2000 PO=P-O
2010 REM ------------------------------------------------------------
2020 REM
2030 REM     PRINTING PART
2040 REM     =============
2050 PRINT
2060 PRINT "PLACE THE PEN AT THE LEFT END OF THE PLOTTER"
2070 GOSUB 6910
2080 Z4=0.35
2090 Z5=0.4
2100 Z9=270
2110 Z2=0
2120 GOSUB 6950
2130 REM
2140 REM     PRINTING THE TITLE TWICE
2150 REM     ------------------------
2160 ZO=26.7
2170 Z$=T$
2180 Z3=LEN(Z$)
2190 Z1=-11+Z3/2*Z4
2200 GOSUB 6920
2210 GOSUB 6940
2220 Z1=-11.01+Z3/2*Z4
2230 GOSUB 6920
2240 GOSUB 6940
2250 REM
2260 REM     NEW CHARACTERS AND PRINTING OF THE ORIGINAL SCALE
2270 REM     ------------------------------------------------
2280 Z4=0.25
2290 Z5=0.2
2300 GOSUB 6950
2310 Y$=STR(A3)
2320 IF E$="1/" THEN 2360
2330 X$="SCALE :"&Y$
2340 X$=X$&" IN/DEG"
2350 GO TO 2390
2360 Z3=LEN(Z$)
2370 Y$=SEG(Y$,2,Z3-1)
2380 X$="SCALE : 1/"&Y$
2390 X$=X$&" AT"
2400 Y$=STR(L1)
```

```
2410 X$=X$&Y$
2420 X$=X$&" DEG."
2430 Z0=23.7
2440 Z1=-3
2450 Z$=X$
2460 Z3=LEN(Z$)
2470 GOSUB 6920
2480 GOSUB 6940
2490 Y9=-5
2500 REM
2510 REM     SETTING CHARACTER TYPE FOR THE TABLE PRINTING
2520 REM     ----------------------------------------------
2530 Z4=0.2
2540 Z5=0.2
2550 GOSUB 6950
2560 Z0=0
2570 Z1=0
2580 GOSUB 6920
2590 REM
2600 REM     FIRST PAGE OF ONE LINE OF 3 PAGES
2610 REM     ---------------------------------
2620 L=1
2630 REM FIRST PAGE OF ONE LINE OF 3 PAGES
2640 I=1
2650 REM
2660 REM     BEGINNING OF THE LOOP TO DRAW ONE LINE OF 3 PAGES
2670 REM     ------------------------------------------------
2680 REM
2690 REM          SETTING NEW ORIGIN
2700 REM          ------------------
2710 GOSUB 6930
2720 REM
2730 REM          DRAWING THE FRAME
2740 REM          -----------------
2750 Z0=29.7
2760 Z2=1
2770 GOSUB 6920
2780 Z1=-21
2790 GOSUB 6920
2800 Z0=0
2810 GOSUB 6920
2820 Z1=0
2830 GOSUB 6920
2840 REM
2850 REM          PRINTING THE TITLE OF THE THREE COLUMNS
2860 REM          ---------------------------------------
2870 Z2=0
2880 Z0=26.7+Y9
2890 Z1=-3
2900 Z$="LATITUDE    LATITUDE         SCALE"
2910 Z3=LEN(Z$)
2920 GOSUB 6920
2930 GOSUB 6940
2940 IF E$="1/" THEN 2970
2950 Z$="DD MM.MM      DD.DDD           1/X"
2960 GO TO 2980
2970 Z$="DD MM.MM      DD.DDD         IN/DEG"
2980 Z3=LEN(Z$)
2990 Z0=Z0-0.5
3000 GOSUB 6920
3010 GOSUB 6940
3020 Z0=Z0-0.6
3030 REM
3040 REM          PRINTING THE LATITUDE IN DEGREES & MINUTES
3050 REM          ------------------------------------------
3060 Z$=STR(T(L,1))
3070 Z3=LEN(Z$)
3080 Z1=-3-(2-Z3)*Z4
3090 GOSUB 6920
3100 GOSUB 6940
3110 Z$=STR(T(L,3))
3120 Z3=LEN(Z$)
3130 Z1=-3-(5-Z3)*Z4
3140 GOSUB 6920
3150 GOSUB 6940
3160 Z$="."
3170 Z3=1
3180 Z1=-3-5*Z4
3190 GOSUB 6920
3200 GOSUB 6940
3210 Z$=STR(T(L,4))
3220 Z3=LEN(Z$)
3230 IF Z3=3 THEN 3300
3240 IF Z3=2 THEN 3270
3250 Z$="00"
3260 GO TO 3310
3270 Z$=SEG(Z$,2,1)
3280 Z$="0"&Z$
3290 GO TO 3310
3300 Z$=SEG(Z$,2,2)
3310 Z3=2
3320 Z1=-3-6*Z4
3330 GOSUB 6920
3340 GOSUB 6940
3350 REM
3360 REM          PRINTING THE DECIMAL LATITUDE VALUE
3370 REM          -----------------------------------
3380 Z$=STR(T(L,1))
3390 Z3=LEN(Z$)
3400 Z1=-3-(15-Z3)*Z4
3410 GOSUB 6920
3420 GOSUB 6940
3430 Z$="."
3440 Z3=1
3450 Z1=-3-15*Z4
3460 GOSUB 6920
3470 GOSUB 6940
3480 Z$=STR(T(L,2))
3490 Z3=LEN(Z$)
3500 IF Z3=4 THEN 3610
3510 IF Z3=3 THEN 3580
3520 IF Z3=2 THEN 3550
3530 Z$="000"
3540 GO TO 3620
3550 Z$=SEG(Z$,2,1)
3560 Z$="00"&Z$
3570 GO TO 3620
3580 Z$=SEG(Z$,2,2)
3590 Z$="0"&Z$
3600 GO TO 3620
```

```
3610 Z$=SEG(Z$,2,3)
3620 Z3=3
3630 Z1=-3-16*Z4
3640 GOSUB 6920
3650 GOSUB 6940
3660 REM
3670 REM          PRINTING THE SCALE VALUE
3680 REM          ------------------------
3690 Z$=STR(T(L,5))
3700 Z3=LEN(Z$)
3710 IF E$="IN/DEG" THEN 3740
3720 Z1=-3-(30-Z3)*Z4
3730 GO TO 3750
3740 Z1=-3-(34-Z3)*Z4
3750 GOSUB 6920
3760 GOSUB 6940
3770 Z$="."
3780 Z3=1
3790 IF E$="IN/DEG" THEN 3820
3800 Z1=-3-30*Z4
3810 GO TO 3830
3820 Z1=-3-34*Z4
3830 GOSUB 6920
3840 GOSUB 6940
3850 Z$=STR(T(L,6))
3860 Z3=LEN(Z$)
3870 IF E$="IN/DEG" THEN 4030
3880 IF Z3=4 THEN 3990
3890 IF Z3=3 THEN 3960
3900 IF Z3=2 THEN 3930
3910 Z$="000"
3920 GO TO 4000
3930 Z$=SEG(Z$,2,1)
3940 Z$="00"&Z$
3950 GO TO 4000
3960 Z$=SEG(Z$,2,2)
3970 Z$="0"&Z$
3980 GO TO 4000
3990 Z$=SEG(Z$,2,3)
4000 Z3=3
4010 Z1=-3-31*Z4
4020 GO TO 4090
4030 IF Z3=2 THEN 4060
4040 Z$="0"
4050 GO TO 4070
4060 Z$=SEG(Z$,2,1)
4070 Z3=1
4080 Z1=-3-35*Z4
4090 GOSUB 6920
4100 GOSUB 6940
4110 L=L+1
4120 REM   TEST : END OF THE TABLE
4130 REM
4140 IF L>J THEN 4390
4150 Z0=Z0-0.4
4160 REM   TEST : PAGE FULL
4170 REM
4180 IF Z0<3 THEN 4200
4190 GO TO 3060
4200 I=I+1
4210 REM
4220 REM    TEST : LINE OF 3 PAGES FULL
4230 REM    ---------------------------
4240 IF I>3 THEN 4310
4250 Z0=0
4260 Z1=-26
4270 GOSUB 6920
4280 Z1=0
4290 Y9=0
4300 GO TO 2710
4310 Z0=32
4320 Z1=-52
4330 GOSUB 6920
4340 Z0=0
4350 Z1=0
4360 GO TO 2640
4370 REM ------------------------------------------------------------------
4380 REM
4390 REM     DRAWING PART
4400 REM     ============
4410 REM   TEST : LINE OF 3 PAGES FULL
4420 REM
4430 I=I+1
4440 IF I>3 THEN 4570
4450 REM
4460 REM     NEW PAGE
4470 REM     --------
4480 Z0=0
4490 Z1=-26
4500 GOSUB 6920
4510 Z1=0
4520 GOSUB 6930
4530 GO TO 4690
4540 REM
4550 REM     NEW LINE
4560 REM     --------
4570 Z0=32
4580 Z1=-52
4590 GOSUB 6920
4600 REM
4610 REM     NEW ORIGIN
4620 REM     ----------
4630 Z0=0
4640 Z1=0
4650 GOSUB 6930
4660 REM
4670 REM     DRAWING THE FRAME
4680 REM     -----------------
4690 Z0=29.7
4700 Z2=1
4710 GOSUB 6920
4720 Z1=-21
4730 GOSUB 6920
4740 Z0=0
4750 GOSUB 6920
4760 Z1=0
4770 GOSUB 6920
4780 Z2=0
4790 REM
4800 REM     THE TITLE
```

```
4810 REM     ---------
4820 REM   SETTING CHARACTER TYPE FOR THE TITLE
4830 REM
4840 Z4=0.35
4850 Z5=0.4
4860 GOSUB 6950
4870 REM   PRINTING THE TITLE
4880 REM
4890 Z0=26.7
4900 Z$=T$
4910 Z3=LEN(Z$)
4920 Z1=-11+Z3/2*Z4
4930 GOSUB 6920
4940 GOSUB 6940
4950 Z1=-11.01+Z3/2*Z4
4960 GOSUB 6920
4970 GOSUB 6940
4980 REM   CHARACTERS AND PRINTING OF THE ORIGINAL SCALE
4990 REM
5000 Z4=0.25
5010 Z5=0.2
5020 GOSUB 6950
5030 Y$=STR(A3)
5040 IF E$="1/" THEN 5080
5050 X$="SCALE :"&Y$
5060 X$=X$&" IN/DEG"
5070 GO TO 5110
5080 Z3=LEN(Y$)
5090 Y$=SEG(Y$,2,Z3-1)
5100 X$="SCALE : 1/"&Y$
5110 X$=X$&" AT"
5120 Y$=STR(L1)
5130 X$=X$&Y$
5140 X$=X$&" DEG."
5150 Z0=23.7
5160 Z1=-3
5170 Z$=X$
5180 Z3=LEN(Z$)
5190 GOSUB 6920
5200 GOSUB 6940
5210 REM   GOING TO THE ORIGIN OF THE VERTICAL AXIS
5220 REM
5230 Z0=4
5240 Z1=-4
5250 GOSUB 6920
5260 REM   SETTING CHARACTER SIZE
5270 REM
5280 Z4=0.2
5290 Z5=0.2
5300 GOSUB 6950
5310 REM     VERTICAL AXIS
5320 REM     -------------
5330 REM   SETTING THE PRINTING FORMAT OF NUMBERS
5340 REM
5350 Z7=V2
5360 IF Z7<>0 THEN 5380
5370 Z7=-1
5380 REM   SETTING PARAMETERS
5390 REM
5400 L=17.5
5410 G0=P0/S/L
5420 J1=INT(P0/S)
5430 Z(15)=0/S
5440 REM   DRAWING THE AXIS AND TICK MARKS, AND PRINTING THE LABELS
5450 REM
5460 FOR K=1 TO J1+1
5470 Z0=4+(K-1)/G0
5480 Z2=1
5490 GOSUB 6920
5500 Z1=-4+0.2
5510 GOSUB 6920
5520 IF K-1-INT((K-1)/Z(15))*Z(15)<>0 THEN 5710
5530 Z8=0+(K-1)*S
5540 Z8=INT(Z8)+INT((Z8-INT(Z8))*10↑Z7)/10↑Z7
5550 Y8=Z8-INT(Z8)
5560 V$=STR(Y8)
5570 X7=LEN(V$)
5580 IF X7=Z7+3 THEN 5660
5590 W$=""
5600 FOR J2=1 TO Z7+3-X7
5610 W$=W$&"0"
5620 NEXT J2
5630 Z$=STR(Z8)
5640 Z$=Z$&W$
5650 GO TO 5670
5660 Z$=STR(Z8)
5670 Z1=Z1+0.2+(LEN(Z$)-0.5)*Z4
5680 Z2=0
5690 GOSUB 6920
5700 GOSUB 6960
5710 Z1=-4
5720 GOSUB 6920
5730 NEXT K
5740 REM   DRAWING THE END OF THE AXIS
5750 REM
5760 Z2=1
5770 Z0=4+L
5780 GOSUB 6920
5790 REM   PRINTING THE TITLE
5800 REM
5810 Z2=0
5820 Z$="SCALE"
5830 Z3=LEN(Z$)
5840 Z0=4+L+1
5850 Z1=-4+Z3/2*Z4
5860 GOSUB 6920
5870 GOSUB 6940
5880 IF E$="IN/DEG" THEN 5910
5890 Z$="IN/DEG"
5900 GO TO 5920
5910 Z$="1/"
5920 Z3=LEN(Z$)
5930 Z0=4+L+0.5
5940 Z1=-4+Z3/2*Z4
5950 GOSUB 6920
5960 GOSUB 6940
5970 REM
5980 REM     HORIZONTAL AXIS
5990 REM     ---------------
6000 REM   SETTING THE PRINTING FORMAT OF NUMBERS
```

```
6010 REM
6020 Z7=V1
6030 IF Z7<>0 THEN 6070
6040 Z7=-1
6050 REM   GOING TO THE ORIGIN OF THE AXIS
6060 REM
6070 Z0=4
6080 Z1=-4
6090 GOSUB 6920
6100 REM   SETTING PARAMETERS
6110 REM
6120 L=15
6130 G0=M0/R/15
6140 J1=INT(M0/R+1.0E-5)
6150 Z(15)=V/R
6160 REM   DRAWING THE AXIS AND TICK MARKS, AND PRINTING THE LABELS
6170 REM
6180 FOR K=1 TO J1+1
6190 Z2=1
6200 Z1=-4-(K-1)/G0
6210 GOSUB 6920
6220 Z0=4-0.2
6230 GOSUB 6920
6240 IF K-1-INT((K-1)/Z(15))*Z(15)<>0 THEN 6380
6250 D1=INT(C9)
6260 D2=((C9-INT(C9))*100/60+(K-1)*R)*60/100
6270 D3=INT(D2*100/60+1.0E-5)
6280 D4=(D2*100/60-INT(D2*100/60))*60/100
6290 IF D4<0.6 THEN 6310
6300 D4=0
6310 Z8=D1+D3+INT(D4*100+1.0E-5)/100
6320 Z$=STR(Z8)
6330 Z0=Z0-0.2-1.5*Z5
6340 Z1=Z1+(LEN(Z$)/2-0.5)*Z4
6350 Z2=0
6360 GOSUB 6920
6370 GOSUB 6960
6380 Z0=4
6390 Z1=-4-(K-1)/G0
6400 GOSUB 6920
6410 NEXT K
6420 REM   DRAWING THE END OF THE AXIS
6430 REM
6440 Z2=1
6450 Z1=-4-L
6460 GOSUB 6920
6470 REM   PRINTING THE TITLE
6480 REM
6490 Z2=0
6500 Z$="LATITUDE"
6510 Z3=8
6520 Z0=4-0.2-5.5*Z5
6530 Z1=-4-L+Z3*Z4
6540 GOSUB 6920
6550 GOSUB 6940
6560 Z$="DD.MM"
6570 Z3=5
6580 Z0=4-0.2-7.5*Z5
6590 Z1=-4-L+6.5*Z4
6600 GOSUB 6920
6610 GOSUB 6940
6620 REM
6630 REM     DRAWING THE CURVE
6640 REM     -----------------
6650 G0=P0/17
6660 G1=M0/15
6670 REM   GOING TO THE FIRST POINT
6680 REM
6690 Z0=4+(T(1,8)-0)/G0
6700 Z1=-4-(T(1,7)-M)/G1
6710 GOSUB 6920
6720 Z2=1
6730 REM   DRAWING THE WHOLE CURVE
6740 REM
6750 FOR I=2 TO J
6760 Z0=4+(T(I,8)-0)/G0
6770 Z1=-4-(T(I,7)-M)/G1
6780 GOSUB 6920
6790 NEXT I
6800 REM   GOING OUT OF THE PAGE
6810 REM
6820 Z0=32
6830 Z1=-25
6840 Z2=0
6850 GOSUB 6920
6860 END
6870 REM ------------------------------------------------------------------
6880 REM
6890 REM     BENSON PLOTTER SUBROUTINES
6900 REM     ==========================
6910 GO TO 6980
6920 GO TO 7220
6930 GO TO 7510
6940 GO TO 7570
6950 GO TO 7670
6960 GO TO 7810
6970 REM
6980 REM     INITIALIZATION OF THE BENSON 1332 PLOTTER
6990 REM     -----------------------------------------
7000 DIM Z(25),Z$(80)
7010 ON SRQ THEN 7200
7020 PRINT "BENSON ADDRESS ? ";
7030 INPUT Z(25)
7040 PRINT @Z(25):"76 !";
7050 Z$=CHR(34)
7060 PRINT @Z(25):Z$;"#$%&'()*+,-./0123456789:;<=>?";
7070 PRINT @Z(25):"@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]↑←";
7080 PRINT @Z(25): USING "A/L":" "
7090 PRINT @Z(25):"76                                    ";
7100 PRINT @Z(25):"                                      ";
7110 PRINT @Z(25): USING "A/L":" "
7120 Z(1)=0
7130 Z(2)=0
7140 Z(3)=200
7150 Z(5)=60
7160 Z(6)=0
7170 Z(13)=0
7180 Z(20)=Z(3)
7190 Z(21)=Z(3)
7200 RETURN
```

```
7210 REM
7220 REM     MOVING THE PEN
7230 REM     --------------
7240 Z(7)=INT(Z0*Z(20))
7250 Z(8)=INT(Z1*Z(21))
7260 Z(9)=Z(7)-Z(1)
7270 Z(10)=Z(8)-Z(2)
7280 Z(1)=Z(7)
7290 Z(2)=Z(8)
7300 Z(7)=ABS(Z(9))
7310 Z(8)=ABS(Z(10))
7320 IF Z(7)>16383 THEN 7410
7330 IF Z(8)>16383 THEN 7410
7340 Z(7)=0
7350 Z(11)=Z(9)
7360 Z(12)=Z(10)
7370 PRINT @Z(25),"PL ";Z(11),Z(12),Z2,
7380 PRINT @Z(25), USING "A/L",";" "
7390 IF Z(7)>1 THEN 7300
7400 RETURN
7410 Z(7)=INT((Z(7)+16383)/16384)
7420 Z(8)=INT((Z(8)+16383)/16384)
7430 IF Z(7)>Z(8) THEN 7450
7440 Z(7)=Z(8)
7450 Z(11)=INT(Z(9)/Z(7))
7460 Z(12)=INT(Z(10)/Z(7))
7470 Z(9)=Z(9)-Z(11)
7480 Z(10)=Z(10)-Z(12)
7490 GO TO 7370
7500 REM
7510 REM     CHANGING THE ORIGIN
7520 REM     -------------------
7530 Z(1)=INT(Z0*Z(20))
7540 Z(2)=INT(Z1*Z(21))
7550 RETURN
7560 REM
7570 REM     WRITING A STRING OF CHARACTERS
7580 REM     ------------------------------
7590 PRINT @Z(25),"TX ";Z3," ";Z$,
7600 PRINT @Z(25), USING "A/L",";" "
7610 Z(1)=Z(1)+Z(5)*Z3
7620 Z(2)=Z(2)+Z(6)*Z3
7630 PRINT @Z(25),"NT ";Z(5)*Z3," ";Z(6)*Z3,
7640 PRINT @Z(25), USING "A/L",";" "
7650 RETURN
7660 REM
7670 REM     INITIALIZATION OF CHARACTER TYPE AND ORIENTATION
7680 REM     -----------------------------------------------
7690 Z(5)=Z4*COS(Z9)
7700 Z(6)=Z4*SIN(Z9)
7710 Z(16)=INT(4*Z(5)*Z(20)/3)
7720 Z(17)=INT(-4*Z5*Z(20)*SIN(Z9)/3)
7730 Z(18)=INT(4*Z(6)*Z(21)/3)
7740 Z(19)=INT(4*Z5*Z(21)*COS(Z9)/3)
7750 PRINT @Z(25),"SZ ";Z(16)," ";Z(17)," ";Z(18)," ";Z(19),
7760 PRINT @Z(25), USING "A/L",";" "
7770 Z(5)=INT(Z(5)*Z(20))
7780 Z(6)=INT(Z(6)*Z(21))
7790 RETURN
7800 REM
7810 REM     PRINTING OF NUMBERS
7820 REM     -------------------
7830 Z7=INT(Z7)
7840 Z(9)=ABS(INT(Z7))
7850 IF Z7>-2 THEN 8030
7860 IF Z(9)<10 THEN 7880
7870 Z(9)=9
7880 Z$=STR(Z(9))
7890 Z$=Z$&"E"
7900 Z(9)=Z(9)+7
7910 IF Z8>0 THEN 7950
7920 Z(9)=Z(9)+1
7930 PRINT @Z(25),"TX ";Z(9)," ",
7940 GO TO 7960
7950 PRINT @Z(25),"TX ";Z(9),
7960 PRINT @Z(25), USING Z$,Z8,
7970 PRINT @Z(25), USING "A/L",";" "
7980 Z(1)=Z(1)+Z(9)*Z(5)
7990 Z(2)=Z(2)+Z(9)*Z(6)
8000 PRINT @Z(25),"NT ";Z(9)*Z(5)," ";Z(9)*Z(6),
8010 PRINT @Z(25), USING "A/L",";" "
8020 RETURN
8030 Z(7)=INT(LGT(ABS(Z8)+0.5))+1
8040 IF Z(7)>0 THEN 8060
8050 Z(7)=1
8060 IF Z8=>0 THEN 8080
8070 Z(7)=Z(7)+1
8080 Z$=STR(Z(7))
8090 Z$=Z$&"D"
8100 IF Z7>-1 THEN 8130
8110 Z(9)=Z(7)
8120 GO TO 7930
8130 Z(8)=LEN(Z$)+2
8140 Z$=Z$&". D"
8150 IF Z(9)<10 THEN 8170
8160 Z(9)=9
8170 GO TO Z(9)+1 OF 8180,8200,8220,8240,8260,8280,8300,8320,8340,8360
8180 Z$=REP(" ",Z(8)+1,1)
8190 GO TO 8370
8200 Z$=REP("1",Z(8),1)
8210 GO TO 8370
8220 Z$=REP("2",Z(8),1)
8230 GO TO 8370
8240 Z$=REP("3",Z(8),1)
8250 GO TO 8370
8260 Z$=REP("4",Z(8),1)
8270 GO TO 8370
8280 Z$=REP("5",Z(8),1)
8290 GO TO 8370
8300 Z$=REP("6",Z(8),1)
8310 GO TO 8370
8320 Z$=REP("7",Z(8),1)
8330 GO TO 8370
8340 Z$=REP("8",Z(8),1)
8350 GO TO 8370
8360 Z$=REP("9",Z(8),1)
8370 Z(9)=Z(9)+Z(7)+1
8380 GO TO 7930
```

B

★

SCALE

PROGRAM TO COMPUTE SCALE IN MERCATOR PROJECTION FROM
INCHES PER DEGREE TO 1/X AND FROM 1/X TO INCH/DEG

ELLIPSOID : WGS 72
ALL NUMBERS MUST BE >0

TITLE : EQUATOR - 35 DEGREES

BEGIN LATITUDE (DD.MM) : 0
END LATITUDE (DD.MM) : 35
INCREMENT (DD.MM) : 1

SCALE : 16
TYPE OF SCALE ? (IN/DEG OR 1/) : IN/DEG
AT LATITUDE (DD.MM) ? 0
BEGIN LAT (DD.MM) = 0 END LAT (DD.MM) = 35
TICKS AND LABELS EVERY ? TICKS ON LAT AXIS
1,5
NUMBER OF DECIMAL CHARACTERS TO PRINT THE LATITUDES : 0

SCALE FROM 273915.983073 TO 224871.138578
SMALLER SCALE, LARGER SCALE
200000,280000
TICKS AND LABELS EVERY ? TICKS ON SCALE AXIS
2000,10000
NUMBER OF DECIMAL CHARACTERS TO PRINT THE SCALE : 0

PLACE THE PEN AT THE LEFT END OF THE PLOTTER
BENSON ADDRESS ? 41

C

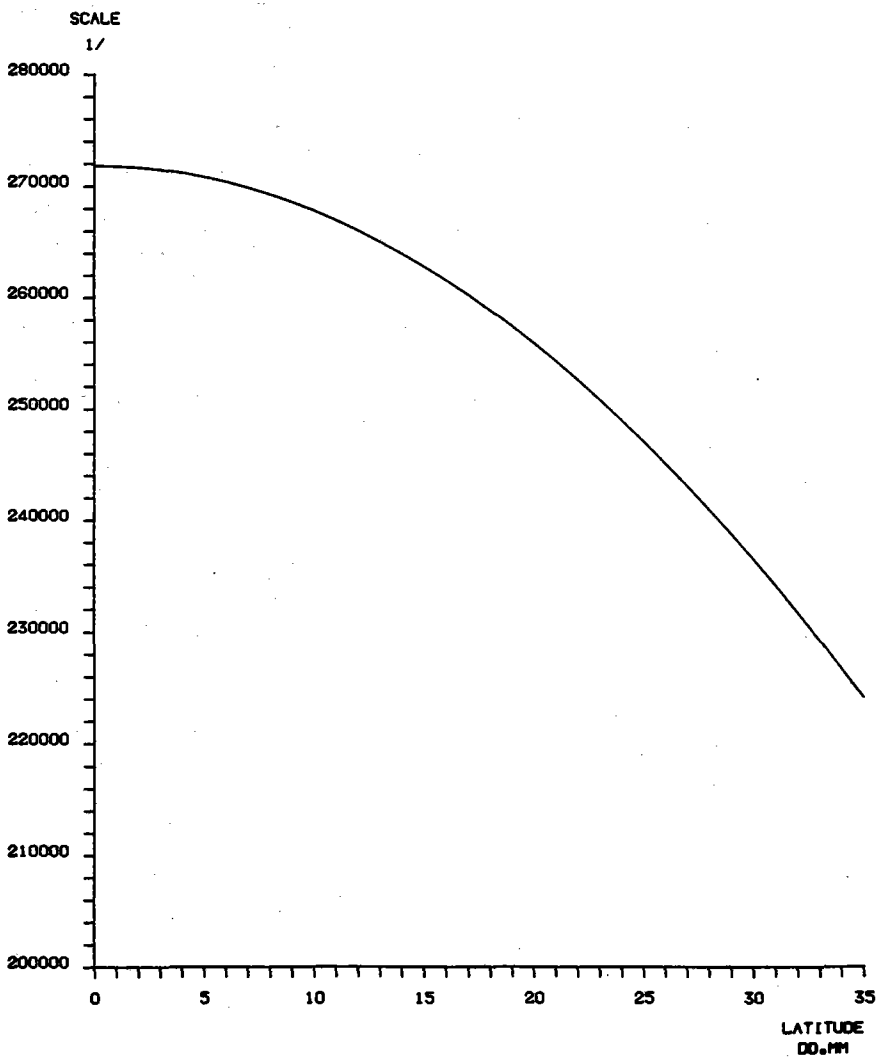
EQUATOR - 35 DEGREES

SCALE = 16 IN/DEG AT 0 DEG.

LATITUDE DD	LATITUDE MM.MM	LATITUDE DD.DDD	SCALE 1/X
0	0.00	0.000	273915.0
1	0.00	1.000	273874.8
2	0.00	2.000	273751.3
3	0.00	3.000	273545.6
4	0.00	4.000	273257.6
5	0.00	5.000	272887.4
6	0.00	6.000	272435.2
7	0.00	7.000	271901.1
8	0.00	8.000	271285.2
9	0.00	9.000	270587.6
10	0.00	10.000	269808.7
11	0.00	11.000	268948.5
12	0.00	12.000	268007.3
13	0.00	13.000	266985.4
14	0.00	14.000	265882.0
15	0.00	15.000	264700.4
16	0.00	16.000	263438.0
17	0.00	17.000	262096.1
18	0.00	18.000	260675.1
19	0.00	19.000	259175.3
20	0.00	20.000	257597.2
21	0.00	21.000	255941.2
22	0.00	22.000	254207.7
23	0.00	23.000	252397.3
24	0.00	24.000	250510.3
25	0.00	25.000	248547.4
26	0.00	26.000	246509.1
27	0.00	27.000	244395.0
28	0.00	28.000	242208.6
29	0.00	29.000	239947.5
30	0.00	30.000	237613.4
31	0.00	31.000	235206.9
32	0.00	32.000	232728.8
33	0.00	33.000	230179.6
34	0.00	34.000	227560.1
35	0.00	35.000	224871.1

EQUATOR - 35 DEGREES

SCALE = 16 IN/DEG AT 0 DEG.



D

SCALE

PROGRAM TO COMPUTE SCALE IN MERCATOR PROJECTION FROM
INCHES PER DEGREE TO 1/X AND FROM 1/X TO INCH/DEG

ELLIPSOID : WGS 72
ALL NUMBERS MUST BE >0

TITLE : CARIBBEANS

BEGIN LATITUDE (DD.MM) : 0
END LATITUDE (DD.MM) : 35
INCREMENT (DD.MM) : 1

SCALE : 200000
TYPE OF SCALE ? (IN/DEG OR 1/) : 1/
AT LATITUDE (DD.MM) ? 33
BEGIN LAT (DD.MM) = 0 END LAT (DD.MM) = 35
TICKS AND LABELS EVERY ? TICKS ON LAT AXIS
1.5
NUMBER OF DECIMAL CHARACTERS TO PRINT THE LATITUDES : 0

SCALE FROM 18.4143666238 TO 15.11726161
SMALLER SCALE, LARGER SCALE
15.19
TICKS AND LABELS EVERY ? TICKS ON SCALE AXIS
.2,1
NUMBER OF DECIMAL CHARACTERS TO PRINT THE SCALE : 0

PLACE THE PEN AT THE LEFT END OF THE PLOTTER
BENSON ADDRESS ? 41

E

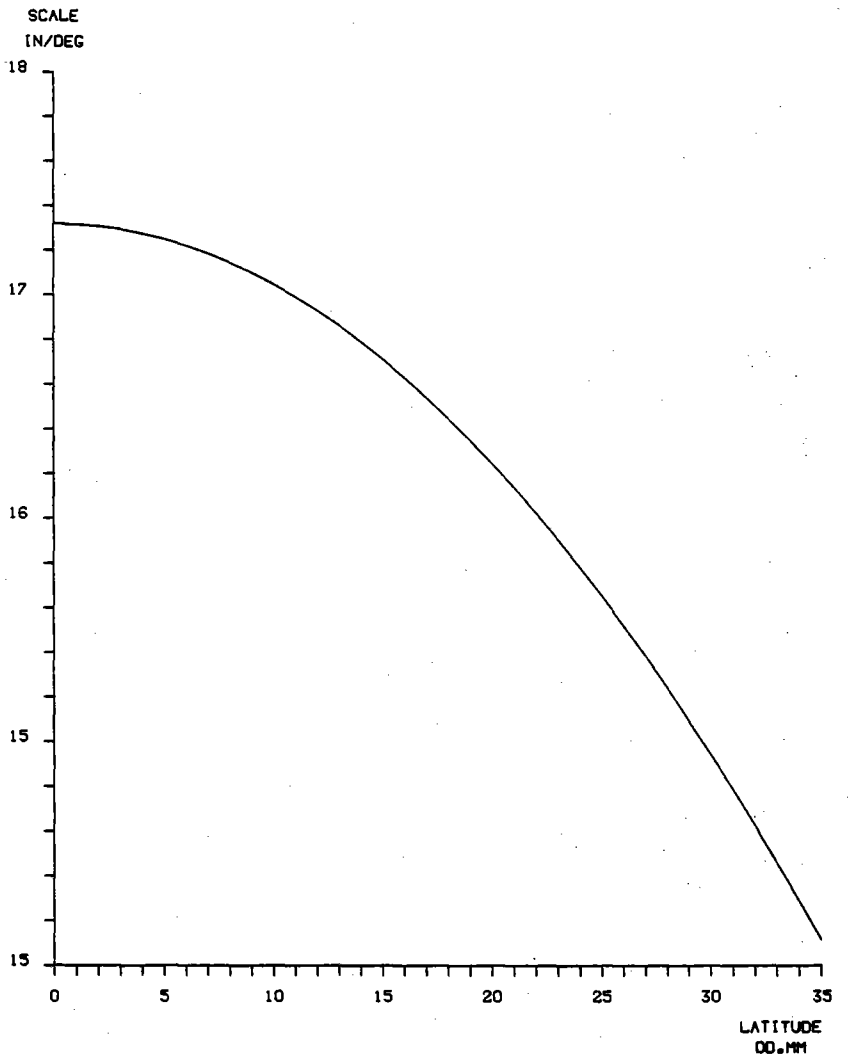
CARIBBEANS

SCALE = 1/200000 AT 33 DEG.

LATITUDE DD	LATITUDE MM.MM	LATITUDE DD.DDD	SCALE IN/DEG
0	0.00	0.000	18.414
1	0.00	1.000	18.412
2	0.00	2.000	18.403
3	0.00	3.000	18.389
4	0.00	4.000	18.370
5	0.00	5.000	18.345
6	0.00	6.000	18.315
7	0.00	7.000	18.279
8	0.00	8.000	18.238
9	0.00	9.000	18.191
10	0.00	10.000	18.138
11	0.00	11.000	18.080
12	0.00	12.000	18.017
13	0.00	13.000	17.948
14	0.00	14.000	17.874
15	0.00	15.000	17.795
16	0.00	16.000	17.710
17	0.00	17.000	17.620
18	0.00	18.000	17.524
19	0.00	19.000	17.423
20	0.00	20.000	17.317
21	0.00	21.000	17.206
22	0.00	22.000	17.089
23	0.00	23.000	16.968
24	0.00	24.000	16.841
25	0.00	25.000	16.709
26	0.00	26.000	16.572
27	0.00	27.000	16.430
28	0.00	28.000	16.283
29	0.00	29.000	16.131
30	0.00	30.000	15.974
31	0.00	31.000	15.812
32	0.00	32.000	15.646
33	0.00	33.000	15.474
34	0.00	34.000	15.298
35	0.00	35.000	15.117

CARIBBEANS

SCALE = 1/200000 AT 33 DEG.



PROFIL DU FOND DU FOSSE D'AMERIQUE CENTRALE

4.1 - CREATION DES FICHIERS DE DONNEES DE POINTS LE LONG DU FOSSE

A - Exemple de fichier de données original écrit en BASIC

B - Listing du programme `RANPROFL' de conversion du fichier de données écrit en BASIC en deux fichiers de données à accés aléatoire

C - Copie d'écran d'un exemple d'exécution du programme `RANPROFL'

4.2 - CALCUL DE LA POSITION DES POINTS LE LONG DU PROFIL

A - Listing du programme `PROFILE'

B - Copie d'écran d'un exemple d'exécution du programme `PROFILE'*

C - Exemple de sortie sur imprimante*

4.3 - DESSIN DU PROFIL

A - Listing du programme `MATPROFILE'

B - Copie d'écran de l'exécution du programme `MATPROFILE'

C - Sortie sur traceur réduite de 71 %

* les lettres a à g de la partie B correspondent à des sorties sur imprimante (partie C) portant le même nom.

4.1

A

```
500 '  ═══════════════════════════════════════════════════════
510 '
520 '                          DATA 'MATRENCH'
530 '
540 '                                        Jacqueline ROUMP, Aug. 1984
550 '  ═══════════════════════════════════════════════════════
560 '
570 '     The data  are latitude, longitude and depth of points along the
580 '  Middle America Trench axis.
590 '     The latitude and longitude are in degrees.minutes :
600 '  the degrees are the integer part of the value,
610 '  the two first decimal are the minutes
620 '  and the third is the tenth of minute
630 '     The data are essentially from Seabeam maps at a 16 inches/degree
640 '  scale, with a 50 m contour interval.
650 '
660 '     This file must be saved this way : SAVE "MATRENCH.DAT",A
670 '     The data must be put in two sequential data files, by the program"
680 '  'RANPROFL' before being used by the program 'PROFILE' for the"
690 '  computation."
700 '
710 '-----------------------------------------------------------------------
720 '
730 N1=34                              'N1 : number of break points
740 N2=650                             'N2 : number of points along the profile
750 '
760 '-----------------------------------------------------------------------
770 '
780 '     DATA FOR THE BREAKPOINT TABLE
790 '     ═══════════════════════════
800 '
810 DATA 18.462,-105.038, 18.407,-104.578, 18.224,-104.333, 17.46,-103.44
820 DATA 17.223,-102.467, 16.579,-101.535, 16.387,-101.117, 16.204,-100.10
830 DATA 16.023,-99.215,  15.464,-98.393,  15.297,-97.476,  15.176,-97.138
840 DATA 15.131,-95.488,  15.004,-95.20,   14.421,-94.506,  14.101,-93.597
850 DATA 13.49,-93.19,    13.29,-92.40,    12.487,-91.144,  12.303,-90.325
860 DATA 12.10,-89.35,    11.466,-88.388,  11.333,-88.10,   10.50,-87.131
870 DATA 9.489,-86.20,    9.09,-85.30,     8.393,-84.255,   8.213,-83.521
880 DATA 8.107,-83.226,   7.50,-82.595,    7.19,-82.50,     7.118, -82.199
890 DATA 6.566,-81.518,   6.529,-79.503
900 '
910 '-----------------------------------------------------------------------
920 '
930 '     DATA FOR THE PROFILE TABLE
940 '     ═══════════════════════
950 '
960 DATA 18.462,-105.038,5000, 18.423,-104.596,5000, 18.411,-104.582,4500
970 DATA 18.403,-104.574,4500, 18.40,-104.573,4550,  18.265,-104.418,4550
980 DATA 18.239,-104.368,4500, 18.231,-104.351,4450, 18.217,-104.315,4250
990 DATA 18.216,-104.313,4300, 18.20,-104.296,4300,  18.186,-104.288,4200
1000 DATA 18.156,-104.279,4350, 18.148,-104.274,4350, 18.134,-104.265,4350
1010 DATA 18.104,-104.24,4400,  18.101,-104.237,4450, 18.089,-104.222,4550
1020 DATA 18.085,-104.21,4650,  18.065,-104.174,4750, 18.034,-104.10,4850
1030 DATA 18.007,-104.051,4850, 17.599,-104.037,4850, 17.594,-104.029,4900
1040 DATA 17.561,-103.568,4900, 17.43,-103.422,4900,  17.396,-103.351,4850
1050 DATA 17.372,-103.29,4800,  17.359,-103.253,4750, 17.344,-103.215,4750
```

•
•
•
•
•
•
•
•
•
•
•

```
2800 DATA 9.073,-85.31,3550,  9.073,-85.30,3500,  9.074,-85.296,3450
2810 DATA 9.074,-85.294,3400, 9.077,-85.282,3400, 9.077,-85.279,3350
2820 DATA 9.077,-85.275,3350, 9.077,-85.26,3550,  9.079,-85.242,3600
2830 DATA 9.077,-85.232,3650, 9.072,-85.207,3700, 9.034,-85.122,3700
2840 DATA 9.023,-85.109,3700, 9.012,-85.09,3700,  9.004,-85.066,3650
2850 DATA 9.001,-85.06,3600,  8.593,-85.046,3600, 8.579,-85.029,3600
2860 DATA 8.546,-84.562,3550, 8.522,-84.519,3550, 8.527,-84.491,3600
2870 DATA 8.48,-84.387,3600,  8.466,-84.366,3500, 8.456,-84.351,3450
2880 DATA 8.45,-84.344,3350,  8.442,-84.335,3300, 8.428,-84.316,3200
2890 DATA 8.423,-84.312,3150, 8.416,-84.301,3100, 8.409,-84.29,2800
2900 DATA 8.395,-84.259,2750, 8.392,-84.253,2750, 8.389,-84.25,2800
2910 DATA 8.38,-84.242,2900,  8.357,-84.213,2900, 8.351,-84.207,2850
2920 DATA 8.341,-84.189,2800, 8.334,-84.172,2750, 8.312,-84.131,2600
2930 DATA 8.304,-84.123,2500, 8.30,-84.115,2450,  8.295,-84.094,2450
2940 DATA 8.285,-84.066,2350, 8.244,-84.012,2250, 8.239,-84.003,2200
2950 DATA 8.232,-83.588,2150, 8.226,-83.566,2100, 8.228,-83.546,2000
2960 DATA 8.213,-83.521,1750, 8.206,-83.51,1750,  8.195,-83.478,1750
2970 DATA 8.189,-83.449,1750, 8.185,-83.438,1850, 8.139,-83.29,1850
2980 DATA 8.136,-83.281,1750, 8.134,-83.276,1500, 8.126,-83.271,1800
2990 DATA 8.104,-83.24,1800
3000 DATA 8.107,-83.226,1500, 8.104,-83.219,1500, 8.101,-83.215,1500
3010 DATA 8.10,-83.204,1750,  8.079,-83.163,1900, 8.05,-83.149,1900
3020 DATA 7.50,-82.595,2000,  7.365,-82.566,2750, 7.285,-82.549,3150
3030 DATA 7.242,-82.535,3250, 7.22,-82.52,3350,   7.16,-82.413,3400
3040 DATA 7.147,-82.35,3400,  7.134,-82.292,3300, 7.123,-82.228,3250
3050 DATA 7.122,-82.219,3000, 7.12,-82.209,3000,  7.118,-82.199,2750
3060 DATA 7.115,-82.19,2800,  7.08,-82.168,2900,  7.057,-82.15,2950
3070 DATA 7.018,-82.11,3050,  6.572,-82.018,3050, 6.563,-81.578,2900
3080 DATA 6.566,-81.55,2850,  6.567,-81.541,2800, 6.566,-81.518,1300
3090 DATA 6.564,-81.497,1500, 6.542,-81.469,1650, 6.52,-81.446,1700
3100 DATA 6.51,-81.422,1750,  6.509,-81.388,1850, 6.504,-81.363,1900
3110 DATA 6.50,-81.35,1950,   6.49,-81.326,2000,  6.479,-81.30,2050
3120 DATA 6.469,-81.259,2150, 6.468,-81.231,2250, 6.466,-81.215,2300
3130 DATA 6.466,-81.196,2450, 6.463,-81.176,2500, 6.461,-81.149,2550
3140 DATA 6.46,-81.124,2750,  6.462,-81.107,2900, 6.462,-81.082,3000
3150 DATA 6.462,-81.037,3200, 6.457,-81.021,3250, 6.455,-81.016,3300
3160 DATA 6.46,-80.594,3350,  6.529,-79.503,3350
```

B

```
10 '  ============================================================
20 '
30 '                          PROGRAM `RANPROFL'
40 '
50 '                                          Jacqueline ROUMP, Sept. 1984
60 '  ============================================================
70 '
80 CLEAR : CLS : OPTION BASE 1
90 '
100 '    PRESENTATION
110 '    ============
120 PRINT "   This program will load a data file and put it in a random"
130 PRINT "access data file to be used then by the program `PROFILE'."
140 PRINT
150 PRINT "   The original final must begin with line 500, and the last line"
160 PRINT "must not exceed 4999"
170 PRINT
180 PRINT "   The file begins with the two variables N1 and N2 containing the"
190 PRINT "number of breakpoints along the profile (N1), and the number of"
200 PRINT "points of the profile (N2)."
210 PRINT
220 PRINT "   The breakpoints data are only Latitudes and Longitudes in"
230 PRINT "degrees.minutes (DD.MMM)=degrees.minutes and tenths of minute."
240 PRINT
250 PRINT "   The points data are Latitudes and Longitudes in DD.MMM, and"
260 PRINT "depths in round meters."
270 PRINT
280 '
290 '    INPUT OF THE DATA INTO THE PROGRAM
300 '    ==================================
310 INPUT "DISK READY (Y/N) (if the data have already been loaded type 0 ";R$
320           R$=LEFT$(R$,1) : IF R$()"Y" AND R$()"y" AND R$()"0" THEN 310
330                            IF R$="0" THEN 410
340 PRINT
350 PRINT "Names of the data files on the disk : "
360 FILES "*.dat"                          'list of the disk data files
370 PRINT
380   INPUT "Name of your data file : ";X$
390 CHAIN MERGE X$,500,ALL                 'merging the data file
400 '
410 '  ------------------------------------------------------------------
420 '     PLACE FOR THE DATA FILE
5000 ' ------------------------------------------------------------------
5010 '
5020 '    READING THE DATA
5030 '    ================
5040 DIM BRK(N1,2)                          'breakpoints table
5050 DIM PNTS(N2,3)                         'profile points table
5060 '
5070 '    Reading the breakpoints data
5080 '    ----------------------------
5090 '---------------
5100 FOR I=1 TO N1
5110    READ BRK(I,1), BRK(I,2)
5120 NEXT I
5130 '---------------
5140 '
5150 '    Reading the profile points data
5160 '    -------------------------------
5170 '---------------
5180 FOR I=1 TO N2
5190    READ PNTS(I,1), PNTS(I,2), PNTS(I,3)
5200 NEXT I
5210 '---------------
5220 ' ------------------------------------------------------------------------
5230 '
5240 '    CREATION OF THE RANDOM ACCESS DATA FILE FOR BREAPOINTS
5250 '    ======================================================
5260 '    Opening the file
5270 '    ----------------
5280 OPEN "R", #1, "BRKPROFL.DAT",8
5290 FIELD #1, 4 AS A$, 4 AS B$
5300 '
5310 '   Input of the breakpoints number into the first record of the file
5320 '   ------------------------------------------------------------------
5330 LSET A$=MKS$(N1)
5340 LSET B$=MKS$(N1)
5350 PUT #1,1
5360 '
5370 '   Input of the data into the file
5380 '   -------------------------------
5390 '---------------
5400 FOR I%=1 TO N1
5410    LSET A$=MKS$(BRK(I%,1))
5420    LSET B$=MKS$(BRK(I%,2))
5430    PUT #1,I%+1
5440 NEXT I%
5450 '---------------
5460 '
5470 '    Closing the file
5480 '    ----------------
5490 CLOSE #1
5500 '
5510 ' ------------------------------------------------------------------------
5520 '
5530 '    CREATION OF THE RANDOM ACCESS DATA FILE FOR PROFILE POINTS
5540 '    =========================================================
5550 '    Opening the file
5560 '    ----------------
5570 OPEN "R", #1, "PNTPROFL.DAT", 10
5580 FIELD #1, 4 AS A$, 4 AS B$, 2 AS C$
5590 '
5600 '   Input of the profile points number into the first record of the file
5610 '   ------------------------------------------------------------------
5620 LSET A$=MKS$(N2)
5630 LSET B$=MKS$(N2)
5640 LSET C$=MKI$(N2)
5650 PUT #1,1
5660 '
5670 '   Input of the data into the file
5680 '   -------------------------------
5690 '---------------
5700 FOR I%=1 TO N2
```

```
5710     LSET A$=MKS$(PNTS(I%,1))
5720     LSET B$=MKS$(PNTS(I%,2))
5730     LSET C$=MKI$(PNTS(I%,3))
5740     PUT #1, I%+1
5750 NEXT I%
5760 '----------------
5770 '
5780 '    Closing the file
5790 '    ----------------
5800 CLOSE #1
5810 '
5820 END
```

C

★

This program will load a data file and put it in a random access data file to be used then by the program 'PROFILE'.

The original final must begin with line 500, and the last line must not exceed 4999

The file begins with the two variables N1 and N2 containing the number of breakpoints along the profile (N1), and the number of points of the profile (N2).

The breakpoints data are only Latitudes and Longitudes in degrees.minutes (DD.MMM)=degrees.minutes and tenths of minute.

The points data are Latitudes and Longitudes in DD.MMM, and depths in round meters.

DISK READY (Y/N) (if the data have already been loaded type 0 ? Y

Names of the data files on the disk :
MATRENCH.DAT

Name of your data file : ? MATRENCH.DAT

4.2

A

```
10 '  ═══════════════════════════════════════════════════════════════
20 '
30 '                              PROGRAM `PROFILE'
40 '
50 '                                                 Jacqueline ROUMP, Aug. 1984
60 '  ═══════════════════════════════════════════════════════════════
70 '
80 CLEAR : CLS : OPTION BASE 1
90 '
100 '    PRESENTATION OF THE PROGRAM ON THREE PAGES
110 '    ═══════════════════════════════════════════
120 '    Page 1
130 '    ------
140 PRINT
150 PRINT "   This program computes the position of points along a profile and
"
160 PRINT "and will give a schematic line drawing of this profile."
170 PRINT
180 PRINT "   The data are located in a separate file. First data having the"
190 PRINT "latitude and longitude of breaK points and other important points"
200 PRINT "along the profile, and then data containing the latitude, longitude
"
210 PRINT "and depth of all the points of the profile."
220 PRINT
230 PRINT "   The latitude and longitude are in degrees & minutes and written"
240 PRINT "in a special format : -104.358 means 104°W 35.8' (the minus sign is
"
250 PRINT "for South latitude and West longitude)."
260 PRINT
270 PRINT "   The first line of the data file must  be numbered 500 and must"
280 PRINT "read the values of the two variables N1 and N2 containing : for N1"
290 PRINT "the number of breaK points, and for N2 the number of points along"
300 PRINT "the profile. For example :"
310 PRINT
320 PRINT "500 N1=30 : N2=560"
330 PRINT
340 PRINT "   The last line number must not exceed 4999. The data file name"
350 PRINT "must have .DAT as extension, and must have been saved in an ASCII"
360 PRINT "format."
370 '
380 LOCATE 25,20 : SCREEN 0,1                'last line, middle ; reverse video
390      PRINT "< PRESS ANY KEY TO CONTINUE >"
400      A$=INKEY$ : IF A$="" THEN 400       'enter any character
410 SCREEN 0,0 : CLS                         'normal video, clear screen
420 '
430 '    Page 2
440 '    ------
450 PRINT
460 PRINT "   This data file must be put into two random access data files by"
470 PRINT "the program `RANPROFL' before any processing within this program."
480 PRINT
490 PRINT "   The program will give you a list on the line printer of the lat"
500 PRINT "and long of the points along the profile in degrees and minutes,"
510 PRINT "and in decimal degrees the distance from one point to the next,"
520 PRINT "and the distance from the first, corrected by the distance between"
530 PRINT "breaK points, and the depth. The distances are in km, and the"
540 PRINT "depth in meters."
550 PRINT
560 PRINT "   Then you will have a printed list of the lat and long of the"
570 PRINT "breaK points in degrees and minutes, and in decimal degrees, the"
580 PRINT "distance in km between the point and the preceding one, and the"
590 PRINT "cumulated sum computed from the points along the profile, the last"
600 PRINT "column will give the distance along the profile from the first"
610 PRINT "point."
620 PRINT
630 PRINT "   You may use another feature of this program : have the distance"
640 PRINT "from the origin of points crossing meridians and parallel, spaced"
650 PRINT "by a chosen interval."
660 '
670 LOCATE 25,20 : SCREEN 0,1
680      PRINT "< PRESS ANY KEY TO CONTINUE >"
690      A$=INKEY$ : IF A$="" THEN 690
700 SCREEN 0,0 : CLS
710 '
720 '    Page 3
730 '    ------
740 PRINT
750 PRINT "   The profile will be drawn on the screen without any indication,"
760 PRINT "it is just done for you to have a schematic view, the length of"
770 PRINT "the screen will show only a profile 640 km long. If the section is"
780 PRINT "longer it will be broken, and will be read from top to bottom of"
790 PRINT "screen."
800 PRINT
810 PRINT "   You will be allowed to make a screen copy on the line printer,"
820 PRINT "but it will take about half and hour. If your profile need more"
830 PRINT "than one screen it will loop until the end, and you will be"
840 PRINT "allowed to make copies."
850 PRINT
860 '
870 '    INPUT OF THE DATA INTO THE PROGRAM
880 '    ═══════════════════════════════════
890 INPUT "DISK READY (Y/N) ";R$ : R$=LEFT$(R$,1)
900     IF R$<>"Y" AND R$<>"y" THEN 890
910 PRINT
920 PRINT "Names of the data files on the disk : "
930 FILES "*.DAT"                            'list of the disk files
940 PRINT
950   INPUT "Name of your breakpoints data file : ";X$
960   INPUT "Name of your profile points data file : ";Y$
970 CLS
980 '
990 '   INPUT OF THE DATA
1000 '   ═════════════════
1010 '   Input of the breakpoints data
1020 '   -----------------------------
1030 '  opening the file
1040 '  ----------------
1050 OPEN "R", £1, X$, 8
1060 FIELD £1, 4 AS A$, 4 AS B$
1070 '
1080 '  reading the number of value records
1090 '  -----------------------------------
1100 GET £1,1
1110 N1=CVS(A$)
```

```
1120 DIM BRK(N1,7)
1130 '
1140 '  reading the value records
1150 '  -------------------------
1160 '--------------
1170 FOR I%=2 TO N1+1
1180    GET £1, I%
1190    BRK(I%-1,1)=CVS(A$)
1200    BRK(I%-1,2)=CVS(B$)
1210 NEXT I%
1220 '--------------
1230 CLOSE £1                                'closing the file
1240 '
1250 '   Input of the profile points data
1260 '   --------------------------------
1270 '  opening the file
1280 '  ----------------
1290 OPEN "R", £1, Y$, 10
1300 FIELD £1, 4 AS A$, 4 AS B$, 2 AS C$
1310 '
1320 '  reading the number of value records
1330 '  -----------------------------------
1340 GET £1,1
1350 N2=CVS(A$)
1360 DIM PNTS(N2,7)
1370 '
1380 '  reading the value records
1390 '  -------------------------
1400 '--------------
1410 FOR I%=2 TO N2+1
1420    GET £1, I%
1430    PNTS(I%-1,1)=CVS(A$)
1440    PNTS(I%-1,2)=CVS(B$)
1450    PNTS(I%-1,3)=CVI(C$)
1460 NEXT I%
1470 '--------------
1480 CLOSE £1                                'closing the file
1490 '-------------------------------------------------------------------------
1500 '
1510 '    ADDITIONAL QUESTIONS
1520 '    ====================
1530 PRINT : PRINT
1540   INPUT "TITLE ";T$
1550 PRINT
1560 PRINT "Are the data continously increasing or decreasing along"
1570   INPUT "                Latitudes (1) or along Longitudes (2) : ";LL
1580       IF LL<>1 AND LL<>2 THEN 1550
1590 PRINT "Are the ";
1600       IF L=1 THEN PRINT "Latitudes "; ELSE PRINT "Longitudes ";
1610   INPUT "increasing (1) or decreasing (2) along the profile : ";DIR
1620       IF DIR <>1 AND DIR <>2 THEN 1590
1630       IF DIR=2 THEN DIR=-1
1640 '
1650 '    Creation of tables of particular points
1660 '    ---------------------------------------
1670 PRINT
1680 PRINT "Would you like to have up to 3 tables (1 with irregular intervals"
1690   INPUT "and 2 with regular intervals of lat or long) (Y/N) ";R$
1700 R$=LEFT$(R$,1) : IF R$<>"Y" AND R$<>"y" AND R$<>"N" AND R$<>"n" THEN 1670
1710                  IF R$="N" OR R$="n" THEN 2170
1720 PRINT
1730   INPUT "How many tables do you want to create (up to 3) ";N
1740       IF N<0 OR N>3 THEN BEEP : GOTO 1730
1750 '---------------
1760 FOR I=1 TO N
1770    PRINT
1780    PRINT "TABLE ";I
1790      INPUT "          Subtitle : "; S$(I)
1800      INPUT "          What kind : irregular intervals (1) (only one)
                    or regular intervals (2) ";S(I,1)
1810          IF S(I,1)<>1 AND S(I,1)<>2 THEN BEEP : GOTO 1800
1820          IF S(I,1)=2 THEN 2050
1830          IF I=1 THEN 1890
1840          IF I=2 THEN IF S(I,1)=2 GOTO 1890 ELSE 1800
1850          IF I=3 THEN IF S(I,1)=2 AND S(1,2)=2 THEN 1890 ELSE 1800
1860    '
1870    '  table with irregular intervals
1880    '  ------------------------------
1890      INPUT "          How many points in the table with irregular interva
ls ";S1
1900    DIM S1(S1,6)
1910    S(I,2)=0
1920           '---------------
1930            FOR J=1 TO S1
1940               PRINT "             Point ";J;
1950                 INPUT " Lat in degrees & minutes ";S1(J,1)
1960               PRINT SPC(20) ;
1970                 INPUT "   Long in degrees and minutes ";S1(J,2)
1980            NEXT J
1990           '---------------
2000    GOTO 2140
2010    '
2020    '  tables with regular intervals
2030    '  -----------------------------
2040   PRINT SPC(5);
2050      INPUT "     Are the data Latitudes (1) or Longitudes (2) ";S(I,2)
2060          IF S(I,2)<>1 AND S(I,2)<>2 THEN BEEP : GOTO 690
2070      INPUT "             first value (DD.MM) ";S(I,3)
2080      INPUT "             last value (DD.MM) ";S(I,4)
2090          IF S(I,4)-S(I,3)>0 THEN S(I,9)=1 ELSE S(I,9)=-1
2100          IF S(I,2)=LL THEN 2110 ELSE 2120
2110          IF S(I,9)<>DIR THEN PRINT "Exchange first and last values "
                    : GOTO 2050
2120      INPUT "             interval (DD.MM) ";S(I,5)
2130          IF S(I,4)<S(I,3) THEN S(I,5)=-S(I,5)
2140 NEXT I
2150 '---------------
2160 '
2170 CLS : LOCATE 12,5                       'clear screen ; middle screen
2180        PRINT "PLEASE, WAIT ..."
2190 '
2200 '-------------------------------------------------------------------------
2210 '
2220 '    COMPUTATION PART
2230 '    ================
2240 '  Transformation of the latitudes and longitudes in degrees & minutes
```

```
2250 '  of the breakpoints table into decimal degrees
2260 '  ------------------------------------------------------------------
2270 '---------------
2280 FOR I=1 TO N1
2290    B1=BRK(I,1) : B2=BRK(I,2)
2300    BRK(I,3)=FIX(B1)+(B1-FIX(B1))*100/60
2310    BRK(I,4)=FIX(B2)+(B2-FIX(B2))*100/60
2320 NEXT I
2330 '---------------
2340 '
2350 '  Transformation of the latitudes and longitudes in degrees & minutes
2360 '  of the profile points table into decimal degrees
2370 '  ------------------------------------------------------------------
2380 '---------------
2390 FOR I=1 TO N2
2400    P1=PNTS(I,1) : P2=PNTS(I,2)
2410    PNTS(I,4)=FIX(P1)+(P1-FIX(P1))*100/60
2420    PNTS(I,5)=FIX(P2)+(P2-FIX(P2))*100/60
2430 NEXT I
2440 '---------------
2450 '
2460 FPI£=3.1415926535898987£/180                       'PI/180
2470 '
2480 '    Definition of fonctions to compute the distance between two points
2490 '    on the Earth
2500 '    ------------------------------------------------------------------
2510 DEF FNA(X,Y,Z,W)=SIN(X*FPI£)*SIN(Z*FPI£)+COS(X*FPI£)*COS(Z*FPI£)*COS((W-Y
)*FPI£)        'cosine of the angular distance between a point of X lat and
                    Y long and a point of Z lat and W long
2520 DEF FNB(X)=(-ATN(X/SQR(1-X*X))+1.570796327£)/FPI£*10000/90
            'distance in km for and angular distance of X
2530 Y=6378.135*FPI£          'length of one degree of longitude at the Equator
2540 DEF FNC(X)=COS(ATN(.99333056822£*TAN(X*FPI£)))*Y
            'length of one degree of longitude at latitude X
2550 '
2560 '    Distance between the breakpoints
2570 '    --------------------------------
2580 C(1,5)=0
2590 '---------------
2600 FOR I=2 TO N1
2610    D£=FNA(BRK(I,3),BRK(I,4),BRK(I-1,3),BRK(I-1,4))
2620    BRK(I,5)=FNB(D£)
2630 NEXT I
2640 '---------------
2650 '
2660 '  Distance between the profile points, angular distances being usually
2670 '  too small for the precision of Z100 the Pythagoras theorem is used
2680 '  to compute the distance
2690 '  ------------------------------------------------------------------
2700 '---------------
2710 FOR I=2 TO N2
2720    E=PNTS(I-1,4)-PNTS(I,4)
2730    H=E*Y
2740    L=(PNTS(I-1,5)-PNTS(I,5))*FNC(E/2+PNTS(I-1,4))
2750    PNTS(I,6)=SQR(H*H+L*L)
2760 NEXT I
2770 '---------------
2780 '
2790 '    Summing the distance along the profile between breakpoints
2800 '    ----------------------------------------------------------
2810 J=2
2820 '---------------
2830 FOR I=2 TO N2
2840       IF PNTS(I,3+LL)*DIR)BRK(J,2+LL)*DIR THEN J=J+1
2850    E=PNTS(I,4)+(BRK(J-1,3)-PNTS(I,4))/2
2860       IF I=N2 THEN 2900
2870       IF BRK(J,6)=0 THEN H=(BRK(J-1,3)-PNTS(I,4))*Y ELSE 2890
2880    L=(BRK(J-1,4)-PNTS(I,5))*FNC(E) : D=SQR(H*H+L*L) : GOTO 2930
2890       IF PNTS(I+1,3+LL)*DIR(=BRK(J,2+LL)*DIR THEN 2920
2900    E=PNTS(I,4)+(PNTS(I,4)-BRK(J,3))/2 : H=(PNTS(I,4)-BRK(J,3))*Y
2910    L=(BRK(J,4)-PNTS(I,5))*FNC(E) : D=SQR(H*H+L*L)+PNTS(I,6) : GOTO 2930
2920    D=PNTS(I,6)
2930    BRK(J,6)=BRK(J,6)+D
2940 NEXT I
2950 '---------------
2960 '
2970 '    Summing up the distance between breakpoints along the profile
2980 '    -------------------------------------------------------------
2990 BRK(1,7)=0
3000 '---------------
3010 FOR I=2 TO N1
3020    BRK(I,7)=BRK(I-1,7)+BRK(I,5)
3030 NEXT I
3040 '---------------
3050 '
3060 '    Summing up the distances along the profile smoothed by using
3070 '    the distance between breakpoints
3080 '    -------------------------------------------------------------
3090 J=2
3100 PNTS(1,6)=0 : PNTS(1,7)=0
3110 '---------------
3120 FOR I=2 TO N2
3130       IF PNTS(I,3+LL)*DIR)BRK(J,2+LL)*DIR THEN J=J+1
3140       IF PNTS(I,3+LL)=BRK(J,2+LL) THEN PNTS(I,7)=BRK(J,7) : GOTO 3160
3150    PNTS(I,7)=PNTS(I-1,7)+PNTS(I,6)*BRK(J,5)/BRK(J,6)
3160 NEXT I
3170 '---------------
3180 '
3190 '    Building up the irregular interval table
3200 '    ----------------------------------------
3210 '---------------
3220 FOR I=1 TO 3
3230       IF S(I,1)=1 THEN 3280
3240 NEXT I
3250 '---------------
3260 GOTO 3520
3270 '
3280 '  transformation of lat and long in degrees.minutes into decimal degrees
3290 '  ------------------------------------------------------------------
3300 '---------------
3310 FOR J=1 TO S1
3320    T1=S1(J,1) : T2=S1(J,2)
3330    S1(J,3)=FIX(T1)+(T1-FIX(T1))*100/60
3340    S1(J,4)=FIX(T2)+(T2-FIX(T2))*100/60
3350 NEXT J
3360 '---------------
```

```
3370 K=1
3380 '---------------
3390 FOR J=1 TO N2
3400       IF S1(K,2+LL)*DIR)PNTS(J,3+LL)*DIR THEN 3490
3410       IF S1(K,2+LL)=PNTS(J,3+LL) THEN S1(K,5)=PNTS(J,7) : GOTO 3470
3420    E=PNTS(J-1,4)+(S1(K,3)-PNTS(J-1,4))/2
3430    H=(PNTS(J-1,4)-S1(K,3))*Y
3440    L=(S1(K,4)-PNTS(J-1,5))*FNC(E)
3450    D=SQR(H*H+L*L)
3460    S1(K,5)=D*(PNTS(J,7)-PNTS(J-1,7))/PNTS(J,6)+PNTS(J-1,7)
3470    K=K+1
3480       IF K)S1 THEN 3520
3490 NEXT J
3500 '---------------
3510 '
3520 '    Buiding the regular interval tables
3530 '    -----------------------------------
3540 '  table S2
3550 '  --------
3560 '---------------
3570 FOR I=1 TO 3
3580       IF S(I,1)=2 THEN 3630
3590 NEXT I
3600 '---------------
3610 GOTO 4210
3620 '
3630 S(I,6)=FIX(S(I,3))+(S(I,3)-FIX(S(I,3)))/60*100
3640 S(I,7)=FIX(S(I,4))+(S(I,4)-FIX(S(I,4)))/60*100
3650 S(I,8)=FIX(S(I,5))+(S(I,5)-FIX(S(I,5)))/60*100
3660 S2=ABS((S(I,6)-S(I,7))/S(I,8))+1
3670 DIM S2(S2,3)
3680 S2(1,1)=S(I,3)
3690 S2(1,2)=S(I,6)
3700 IN=S(I,8)
3710 '---------------
3720 FOR J=2 TO S2
3730    S2(J,2)=S2(1,2)+(J-1)*IN
3740    S2(J,1)=FIX(S2(J,2))+(S2(J,2)-FIX(S2(J,2)))*60/100
3750 NEXT J
3760 '---------------
3770 J=1
3780 T=3+S(I,2)
3790 '---------------
3800 FOR K=2 TO N2
3810       IF PNTS(K,T)*S(I,9)(S2(J,2)*S(I,9) THEN 3840
3820    S2(J,3)=PNTS(K-1,7)+(S2(J,2)-PNTS(K-1,T))*(PNTS(K,7)-PNTS(K-1,7))/(PNT
S(K,T)-PNTS(K-1,T))
3830       IF J(S2 THEN J=J+1 ELSE 3870
3840 NEXT K
3850 '---------------
3860 '
3870 '  table S3
3880 '  --------
3890    IF I=3 THEN 4210
3900 '---------------
3910 FOR I=I+1 TO 3
3920       IF S(I,1)=2 THEN 3970
3930 NEXT I
3940 '---------------
3950 GOTO 4210
3960 '
3970 S(I,6)=FIX(S(I,3))+(S(I,3)-FIX(S(I,3)))/60*100
3980 S(I,7)=FIX(S(I,4))+(S(I,4)-FIX(S(I,4)))/60*100
3990 S(I,8)=FIX(S(I,5))+(S(I,5)-FIX(S(I,5)))/60*100
4000 S3=ABS((S(I,6)-S(I,7))/S(I,8))+1
4010 S3(1,1)=S(I,3)
4020 S3(1,2)=S(I,6)
4030 IN=S(I,8)
4040 '---------------
4050 FOR J=2 TO S3
4060    S3(J,2)=S3(1,2)+(J-1)*IN
4070    S3(J,1)=FIX(S3(J,2))+(S3(J,2)-FIX(S3(J,2)))*60/100
4080 NEXT J
4090 '---------------
4100 J=1
4110 T=3+S(I,2)
4120 '---------------
4130 FOR K=2 TO N2
4140       IF PNTS(K,T)*S(I,9)(S3(J,2)*S(I,9) THEN 4170
4150    S3(J,3)=PNTS(K-1,7)+(S3(J,2)-PNTS(K-1,T))*(PNTS(K,7)-PNTS(K-1,7))/(PNT
S(K,T)-PNTS(K-1,T))
4160       IF J(S3 THEN J=J+1 ELSE 4210
4170 NEXT K
4180 '---------------
4190 '-------------------------------------------------------------------------
4200 '
4210 '    PRINTING OF THE RESULTS ON THE LINE PRINTER
4220 '    ===========================================
4230 '
4240 BEEP
4250 PRINT : PRINT
4260 '
4270 '    Printing of the breakpoints table
4280 '    ---------------------------------
4290   INPUT "Ready to print the table with the break points (Y/N) ";R$
4300            R$=LEFT$(R$,1) : IF R$()"Y" AND R$()"y" THEN 4290
4310 '
4320 '  preparing the printer
4330 '  ---------------------
4340 LPRINT CHR$(27)"E"CHR$(27)"L012"CHR$(27)"!"CHR$(27)"n"
4350 LPRINT SPC(30-LEN(T$)/2)CHR$(27)"X"T$              'printing the title
4360 LPRINT CHR$(27)"Y" : LPRINT CHR$(27)"N"
4370 LPRINT : LPRINT : LPRINT
4380 LPRINT "   Break points along the profile"     'printing the subtitle
4390 LPRINT CHR$(27)"E" : LPRINT CHR$(27)CHR$(34)
4400 LPRINT
4410 '
4420 '  printing the column headings
4430 '  ----------------------------
4440 LPRINT "Point   Latitude   Longitude   Latitude  Longitude     Deach    D
cumul    Distorg"
4450 LPRINT " No.      in degrees-minutes      in decimal degrees      in km
in km      in km"
4460 '
4470 '  printing the point distances
4480 '  ----------------------------
```

```
4490 LPRINT
4500 '---------------
4510 FOR I=1 TO N1
4520    A=BRK(I,1) : B=BRK(I,2)
4530    LPRINT USING "£££  ";I;
4540    LPRINT USING "   ££"";FIX(A);
4550          LPRINT USING "££.£'  ";(A-FIX(A))*100;
4560    LPRINT USING "£££££";FIX(B);
4570          LPRINT USING "££.£'  ";ABS(B-FIX(B))*100;
4580    LPRINT USING "  ££.£££"  ";BRK(I,3);
4590    LPRINT USING "£££££.£££"  ";BRK(I,4);
4600    LPRINT USING "  £££.££  ";BRK(I,5);
4610    LPRINT USING "  £££.££  ";BRK(I,6);
4620    LPRINT USING "  ££££.££";BRK(I,7)
4630 NEXT I
4640 '---------------
4650 '
4660 '    Printing of the profile table
4670 '    -----------------------------
4680 PRINT : PRINT
4690   INPUT "Ready to print the table with all the profile points (Y/N) ";R$
4700            R$=LEFT$(R$,1) : IF R$()"Y" AND R$()"y" THEN 4690
4710 '
4720 '  preparing the printer
4730 '  ---------------------
4740 LPRINT CHR$(27)"E"CHR$(27)"L012"CHR$(27)"!"CHR$(27)"n"
4750 LPRINT SPC(30-LEN(T$)/2)CHR$(27)"X"T$              'printing the title
4760 LPRINT CHR$(27)"Y" : LPRINT CHR$(27)"N"
4770 LPRINT : LPRINT : LPRINT
4780 LPRINT "   All the points of the profile"      'printing the subtitle
4790 LPRINT CHR$(27)"E" : LPRINT CHR$(27)CHR$(34)
4800 LPRINT
4810 '
4820 '  printing the column headings
4830 '  ----------------------------
4840 LPRINT "Point   Latitude   Longitude   Latitude  Longitude    Deach    Di
storg    Depth"
4850 LPRINT " No.     in degrees-minutes     in decimal degrees    in Km    in
 Km      in m"
4860 LPRINT
4870 '---------------
4880 FOR I=1 TO N2
4890    A=PNTS(I,1) : B=PNTS(I,2)
4900    LPRINT USING "£££  ";I;
4910    LPRINT USING "   ££"";FIX(A);
4920          LPRINT USING "££.£'  ";(A-FIX(A))*100;
4930    LPRINT USING "£££££";FIX(B);
4940          LPRINT USING "££.£'  ";ABS(B-FIX(B))*100;
4950    LPRINT USING "  ££.£££"  ";PNTS(I,4);
4960    LPRINT USING "£££££.£££"  ";PNTS(I,5);
4970    LPRINT USING " £££.££  ";PNTS(I,6);
4980    LPRINT USING "  ££££.££  ";PNTS(I,7);
4990    LPRINT USING "   £££££";PNTS(I,3)
5000 NEXT I
5010 '---------------
5020 '
5030 '    Printing of the irregular intervals table
5040 '    -----------------------------------------
5050 '---------------
5060 FOR I=1 TO 3
5070       IF S(I,1)=1 THEN 5120
5080 NEXT I
5090 '---------------
5100 GOTO 5480
5110 '
5120 PRINT : PRINT
5130   PRINT "Ready to print the table with irregular intervals : ";
                              S$(I); : INPUT " (Y/N) ";R$
5140            R$=LEFT$(R$,1) : IF R$()"Y" AND R$()"y" THEN 5580
5150 '
5160 '  preparing the printer
5170 '  ---------------------
5180 LPRINT CHR$(27)"E"CHR$(27)"L012"CHR$(27)"!"CHR$(27)"n"
5190 LPRINT SPC(30-LEN(T$)/2)CHR$(27)"X"T$              'printing the title
5200 LPRINT CHR$(27)"Y" : LPRINT CHR$(27)"N"
5210 LPRINT : LPRINT : LPRINT
5220 LPRINT "    ";S$(I)                                'printing the subtitle
5230 LPRINT CHR$(27)"E" : LPRINT CHR$(27)CHR$(34)
5240 LPRINT
5250 '
5260 '  printing the column headings
5270 '  ----------------------------
5280 LPRINT "Point   Latitude   Longitude   Latitude  Longitude     Distorg"
5290 LPRINT " No.     in degrees-minutes     in decimal degrees      in Km"
5300 '
5310 '  printing the point distances
5320 '  ----------------------------
5330 LPRINT
5340 '---------------
5350 FOR J=1 TO S1
5360    A=S1(J,1) : B=S1(J,2)
5370    LPRINT USING "£££  ";J;
5380    LPRINT USING "   ££"";FIX(A);
5390    LPRINT USING "££.£'  ";(A-FIX(A))*100;
5400    LPRINT USING "£££££";FIX(B);
5410    LPRINT USING "££.£'  ";ABS(B-FIX(B))*100;
5420    LPRINT USING "  ££.£££"  ";S1(J,3);
5430    LPRINT USING "£££££.£££"  ";S1(J,4);
5440    LPRINT USING "   ££££.££";S1(J,5)
5450 NEXT J
5460 '---------------
5470 '
5480 '    Printing of the first table with regular intervals
5490 '    --------------------------------------------------
5500 '---------------
5510 FOR I=1 TO 3
5520       IF S(I,1)=2 THEN 5570
5530 NEXT I
5540 '---------------
5550 GOTO 6360
5560 '
5570 PRINT : PRINT
5580   PRINT "Ready to print the first table with regular intervals : ";
                              S$(I); : INPUT " (Y/N) ";R$
5590            R$=LEFT$(R$,1) : IF R$()"Y" AND R$()"y" THEN 5580
5600 '
```

```
5610 '  preparing the printer
5620 '  ---------------------
5630 LPRINT CHR$(27)"E"CHR$(27)"L012"CHR$(27)"!"CHR$(27)"n"
5640 LPRINT SPC(30-LEN(T$)/2)CHR$(27)"X"T$              'printing the title
5650 LPRINT CHR$(27)"Y" : LPRINT CHR$(27)"N"
5660 LPRINT : LPRINT : LPRINT
5670 LPRINT "   ";S$(I)                                 'printing the subtitle
5680 LPRINT CHR$(27)"E" : LPRINT CHR$(27)CHR$(34)
5690 LPRINT
5700 '
5710 '  printing the column headings
5720 '  ---------------------------
5730 IF S(I,2)=1 THEN S1$=" Latitude" ELSE S1$="Longitude"
5740 LPRINT "Point     ";S1$;"    ";S1$;"     Distorg"
5750 LPRINT " No.      in deg-min   decim. deg      in km"
5760 '
5770 '  printing the point distances
5780 '  ---------------------------
5790 LPRINT
5800 '---------------
5810 FOR K=1 TO S2
5820    A=S2(K,1)
5830    LPRINT USING "£££   ";K;
5840    LPRINT USING "   ££££"";FIX(A);
5850    LPRINT USING "££.£'  ";ABS((A-FIX(A)))*100;
5860    LPRINT USING " ££££.£££"  ";S2(K,2);
5870    LPRINT USING "    ££££.££";S2(K,3)
5880 NEXT K
5890 '---------------
5900 '
5910 '  is there a second table with regular intervals ?
5920 '  ------------------------------------------------
5930       IF I=3 THEN 6360
5940 FOR I=I+1 TO 3
5950       IF S(I,1)=2 THEN 6010
5960 NEXT I
5970 GOTO 6360
5980 '
5990 '    Printing of the second table with regular intervals
6000 '    --------------------------------------------------
6010 PRINT : PRINT
6020   PRINT "Ready to print the second table with regular intervals : ";
                              S$(I); : INPUT " (Y/N) ";R$
6030            R$=LEFT$(R$,1) : IF R$<>"Y" AND R$<>"y" THEN 5580
6040 '
6050 '  preparing the printer
6060 '  ---------------------
6070 LPRINT CHR$(27)"E"CHR$(27)"L012"CHR$(27)"!"CHR$(27)"n"
6080 LPRINT SPC(30-LEN(T$)/2)CHR$(27)"X"T$              'printing the title
6090 LPRINT CHR$(27)"Y" : LPRINT CHR$(27)"N"
6100 LPRINT : LPRINT : LPRINT
6110 LPRINT "   ";S$(I)                                 'printing the subtitle
6120 LPRINT CHR$(27)"E" : LPRINT CHR$(27)CHR$(34)
6130 LPRINT
6140 '
6150 '  printing the column headings
6160 '  ---------------------------
6170     IF S(I,2)=1 THEN S1$="Latitude " ELSE S1$="Longitude"
6180 LPRINT "Point     ";S1$;"    ";S1$;"      Distorg"
6190 LPRINT " No.      in deg-min   decim. deg      in km"
6200 '
6210 '  printing the point distances
6220 '  ---------------------------
6230 LPRINT
6240 '---------------
6250 FOR K=1 TO S3
6260    A=S3(K,1)
6270    LPRINT USING "£££   ";K;
6280    LPRINT USING "   ££"";FIX(A);
6290    LPRINT USING "££.£'  ";ABS((A-FIX(A)))*100;
6300    LPRINT USING "   ££.£££  ";S3(K,2);
6310    LPRINT USING "   ££££.££";S3(K,3)
6320 NEXT K
6330 '---------------
6340 '-------------------------------------------------------------------------
6350 '
6360 '    DRAWING OF THE PROFILE ON THE SCREEN
6370 '    ====================================
6380 CLS
6390 '
6400 '    Presentation and questions
6410 '    --------------------------
6420 PRINT
6430 PRINT "Now you will see the drawing of the profile on the screen."
6440 PRINT
6450 PRINT "The screen is 640 points wide : what distance in km do you want"
6460   INPUT "between 2 points : ";SL
6470 PRINT "The screen is 225 lines high : what distance in m do you want"
6480   INPUT "between 2 lines : ";SH
6490 PRINT
6500 '
6510 PRINT "You will have the choice between : "
6520 PRINT "    Make a copy of the screen on the printer"
6530 PRINT "    Change the parameters"
6540 PRINT "    Or end the program"
6550 PRINT "If you want to make a copy on the printer : Keep the printer"
6560 PRINT "ready, and type ";
6570   SCREEN 0,1 : PRINT "C"; : SCREEN 0,0             'C in reverse video
6580 PRINT " (no echo will appear on the screen)."
6590 PRINT "If you want to change the parameters : type ";
6600   SCREEN 0,1 : PRINT "P"; : SCREEN 0,0             'P in reverse video
6610 PRINT "Typing any other key will end the program"
6620 PRINT "   If the profile is too long to be drawn on only one screen"
6630 PRINT "page, a small dot" : PRINT "will appear at the bottom left of"
6640 PRINT "the screen and you will hear a beep :"
6650 PRINT "if you want to make a copy on the printer type ";
6660   SCREEN 0,1 : PRINT "C" : SCREEN 0,0
6670 PRINT "if you want to change the parameters type ";
6680   SCREEN 0,1 : PRINT "P" : SCREEN 0,0
6690 PRINT "if you want to see the next part of the profile type ";
6700   SCREEN 0,1 : PRINT "N" : SCREEN 0,0
6710 '
6720 LOCATE 25,10 : SCREEN 0,1                   'last line, reverse video
6730      PRINT "< PRESS ANY KEY TO HAVE THE DRAWING AND BE PATIENT >"
6740      A$=INKEY$ : IF A$="" THEN 6740          'enter any character
6750 '
```

```
6760 SCREEN 1,0 : CLS          'setting the screen to graphics and clearing it
6770 '
6780 '    Drawing
6790 '    -------
6800 LINE (0,0)-(639,222),,B              'drawing of the frame
6810 '
6820 '  drawing of the lines
6830 '  --------------------
6840 J=1 : D=0 : GOTO 6900     'first line
6850 J=K                       'other lines
6860 '
6870 '  looking for the last point of the line
6880 '  --------------------------------------
6890 '---------------
6900 FOR I=J TO N2
6910       IF PNTS(I,7)/SL<(640+PNTS(J,7)/SL) THEN 6930
6920       K=I-1 : GOTO 6980
6930 NEXT I
6940     IF I=N2+1 THEN K=N2
6950 '
6960 '  looking for the minimum and maximum depths of the line
6970 '  ------------------------------------------------------
6980 MIN=PNTS(J,3) : MAX=PNTS(J,3)
6990 '---------------
7000 FOR I=J+1 TO K
7010       IF PNTS(I,3)<MIN THEN MIN=PNTS(I,3)
7020       IF PNTS(I,3)>MAX THEN MAX=PNTS(I,3)
7030 NEXT I
7040 '---------------
7050     IF D+(MAX-MIN)/SH+5)221 THEN PSET (0,224) ELSE 7070
                          'if the last line goes out of the screen
7060 BEEP : GOTO 7200
7070 PSET (0,PNTS(J,3)/SH+D-MIN/SH+5)       'first point of the line
7080 '---------------
7090 FOR I=J+1 TO K
7100    LINE -(PNTS(I,7)/SL-PNTS(J,7)/SL,PNTS(I,3)/SH+D-MIN/SH+5)
                   'drawing of the line linking the points of the profile
7110 NEXT I
7120 '---------------
7130 D=D+(MAX-MIN)/SH+5
      'distance on the screen from the top to the min. depth of the next line
7140     IF K<N2 THEN 6850
7150 '
7160 '  inquiring about the next work to do : continue to draw the lines
7170 '  on the screen, change the parameters of the drawing, copy
7180 '  the drawing on the printer
7190 '  ------------------------------------------------------------
7200 A$=INKEY$ : IF A$="" THEN 7200
7210     IF A$="N" OR A$="n" THEN D=0 ELSE 7250
7220 CLS                                    'clears the screen
7230 LINE (0,0)-(639,222),,B : GOTO 7070    'new frame
7240 GOTO 7070
7250     IF A$="P" OR A$="p" THEN CLS ELSE 7280
7260 SCREEN 0                               'normal screen
7270 GOTO 6450
7280     IF A$="C" OR A$="c" THEN 7310 ELSE 7690
7290 '-------------------------------------------------------------------------
7300 '
```

```
7310 '    SCREEN COPY ON THE LINE PRINTER
7320 '    ===============================
7330 '
7340 LPRINT CHR$(27)"E"CHR$(27)"L012"CHR$(27)"!"CHR$(27)"n"
       'setting the left margin and the character type for the title
7350 LPRINT SPC(30-LEN(T$)/2)CHR$(27)"X"T$          'printing the title
7360 LPRINT CHR$(27)"Y" : LPRINT CHR$(27)"N"
7370 LPRINT : LPRINT : LPRINT
7380 LPRINT "   Screen copy of the profile"       'printing the subtitle
7390 LPRINT CHR$(27)CHR$(34)
7400 LPRINT
7410 LPRINT " The drawing is";640/SL;"km wide."
7420 LPRINT " It is 221 lines high, one interval between two lines represents"
7430 LPRINT SH;"m."
7440 LPRINT CHR$(27)"p" : LPRINT CHR$(27)"T02"
                                  'setting the printer for graphics
7450 '
7460 '  Copy line after line of the 640 dots of each line
7470 '  -------------------------------------------------
7480 '---------------
7490 FOR IS=0 TO 222
7500    X1$="" : X2$="" : X3$="" : X4$=""
7510            '
7520            'reading each point along one line from 0 to 639
7530            '-----------------------------------------------
7540            '---------------
7550             FOR JS=0 TO 159 : X1$=X1$+CHR$(POINT(JS,IS)/7) : NEXT JS
7560             FOR JS=160 TO 319 : X2$=X2$+CHR$(POINT(JS,IS)/7) : NEXT JS
7570             FOR JS=320 TO 479 : X3$=X3$+CHR$(POINT(JS,IS)/7) : NEXT JS
7580             FOR JS=480 TO 639 : X4$=X4$+CHR$(POINT(JS,IS)/7) : NEXT JS
7590            '---------------
7600    LPRINT CHR$(27)"G0640"X1$X2$X3$X4$
7610 NEXT IS
7620 '---------------
7630 '
7640 LPRINT CHR$(27)"A" : LPRINT CHR$(27)"E"
                              'setting the printer back to character printing
7650 GOTO 7200              'to test a new character entry on the keyboard
7660 '
7670 '    The end
7680 '    -------
7690 SCREEN 0               'setting the screen back to normal
7700 END
```

B

★

This program computes the position of points along a profile and and will give a schematic line drawing of this profile.

The data are located in a separate file. First data having the latitude and longitude of break points and other important points along the profile, and then data containing the latitude, longitude and depth of all the points of the profile.

The latitude and longitude are in degrees & minutes and written in a special format : -104.358 means 104°W 35.8' (the minus sign is for South latitude and West longitude).

The first line of the data file must be numbered 500 and must read the values of the two variables N1 and N2 containing : for N1 the number of break points, and for N2 the number of points along the profile. For example :

500 N1=30 : N2=560

The last line number must not exceed 4999. The data file name must have .DAT as extension, and must have been saved in an ASCII format.

[redacted]

★

This data file must be put into two random access data files by the program 'RANPROFL' before any processing within this program.

The program will give you a list on the line printer of the lat and long of the points along the profile in degrees and minutes, and in decimal degrees the distance from one point to the next, and the distance from the first, corrected by the distance between break points, and the depth. The distances are in km, and the depth in meters.

Then you will have a printed list of the lat and long of the break points in degrees and minutes, and in decimal degrees, the distance in km between the point and the preceding one, and the cumulated sum computed from the points along the profile, the last column will give the distance along the profile from the first point.

You may use another feature of this program : have the distance from the origin of points crossing meridians and parallel, spaced by a chosen interval.

[redacted]

★

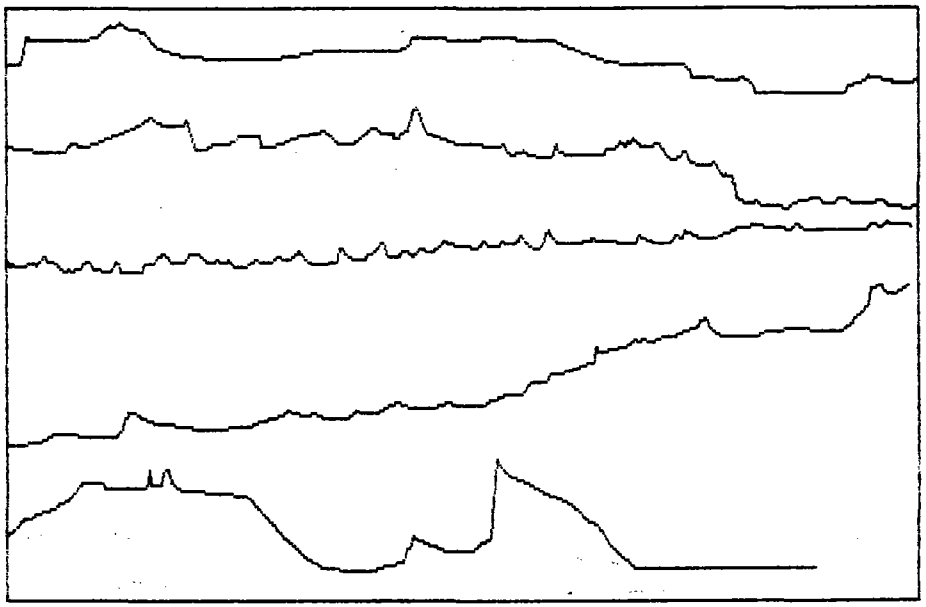
The profile will be drawn on the screen without any indication, it is just done for you to have a schematic view, the length of the screen will show only a profile 640 km long. If the section is longer it will be broken, and will be read from top to bottom of screen.

You will be allowed to make a screen copy on the line printer, but it will take about half and hour. If your profile need more than one screen it will loop until the end, and you will be allowed to make copies.

```
DISK READY (Y/N) ? Y

Names of the data files on the disk :
MATRENCH.DAT BRKPROFL.DAT PNTPROFL.DAT

Name of your breakpoints data file : ? BRKPROFL.DAT
Name of your profile points data file : ? PNTPROFL.DAT

TITLE ? MIDDLE AMERICA TRENCH AXIS

Are the data continously increasing or decreasing along
            Latitudes (1) or along Longitudes (2) : ? 2
Are the Longitudes increasing (1) or decreasing (2) along the profile : ? 1

Would you like to have up to 3 tables (1 with irregular intervals
and 2 with regular intervals of lat or long) (Y/N) ? Y

How many tables do you want to create (up to 3) ? 2

TABLE  1
        Subtitle : ? Circle Sections
        What kind : irregular intervals (1) (only one)
                    or regular intervals (2) ? 1
        How many points in the table with irregular intervals ? 10
           Point  1  Lat in degrees & minutes ? 18.487
                     Long in degrees and minutes ? -104.578
           Point  2  Lat in degrees & minutes ? 16.1
                     Long in degrees and minutes ? -99.418
           Point  3  Lat in degrees & minutes ? 16.23
                     Long in degrees and minutes ? -99.215
           Point  4  Lat in degrees & minutes ? 15.131
                     Long in degrees and minutes ? -95.488
           Point  5  Lat in degrees & minutes ? 15.004
                     Long in degrees and minutes ? -95.2
           Point  6  Lat in degrees & minutes ? 11.333
                     Long in degrees and minutes ? -88.1
           Point  7  Lat in degrees & minutes ? 10.503
                     Long in degrees and minutes ? -87.134
           Point  8  Lat in degrees & minutes ? 8.213
                     Long in degrees and minutes ? -83.521
           Point  9  Lat in degrees & minutes ? 7.134
                     Long in degrees and minutes ? -82.292
           Point  10  Lat in degrees & minutes ? 6.529
                     Long in degrees and minutes ? -79.503
```

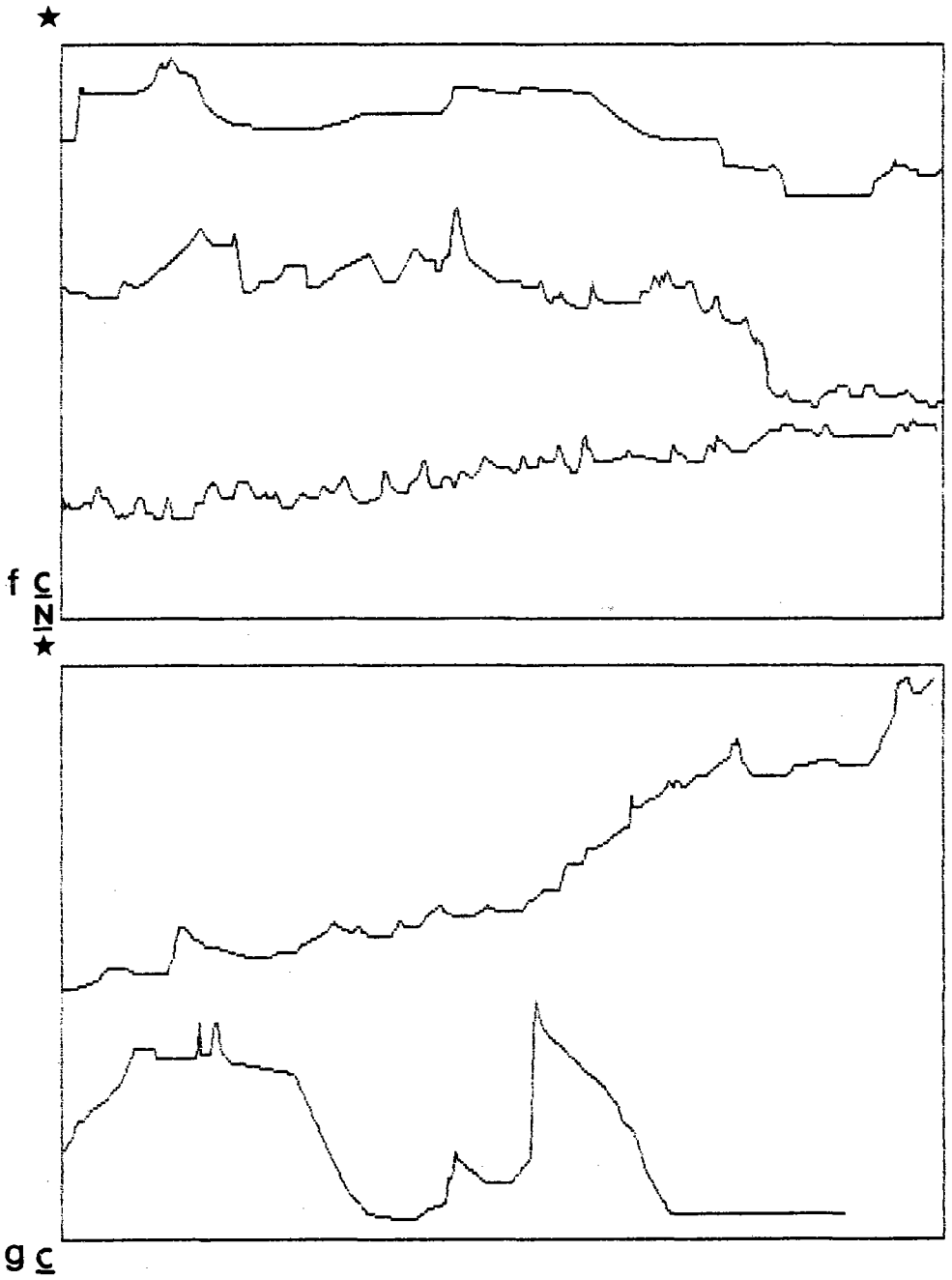
```
TABLE  2
        Subtitle : ? Longitudes
        What kind : irregular intervals (1) (only one)
                    or regular intervals (2) ? 2
    Are the data Latitudes (1) or Longitudes (2) ? 2
          first value (DD.MM) ? -105
          last value (DD.MM) ? -80
          interval (DD.MM) ? 1

★
   PLEASE, WAIT ...

Ready to print the table with the break points (Y/N) ? y
a
Ready to print the table with all the profile points (Y/N) ? Y
b
Ready to print the table with irregular intervals : Circle Sections (Y/N) ? Y
c
Ready to print the first table with regular intervals : Longitudes (Y/N) ? Y
d
★
Now you will see the drawing of the profile on the screen.

The screen is 640 points wide : what distance in km do you want
between 2 points : ? 1
The screen is 225 lines high : what distance in m do you want
between 2 lines : ? 50

You will have the choice between :
    Make a copy of the screen on the printer
    Change the parameters
    Or end the program
If you want to make a copy on the printer : keep the printer
ready, and type C (no echo will appear on the screen).
If you want to change the parameters : type P
Typing any other key will end the program
   If the profile is too long to be drawn on only one screen
page, a small dot
will appear at the bottom left of
the screen and you will hear a beep :
if you want to make a copy on the printer type C
if you want to change the parameters type P
if you want to see the next part of the profile type C
```

★



e C
P

```
The screen is 640 points wide : what distance in km do you want
between 2 points : ? 1
The screen is 225 lines high : what distance in m do you want
between 2 lines : ? 25

You will have the choice between :
    Make a copy of the screen on the printer
    Change the parameters
    Or end the program
If you want to make a copy on the printer : keep the printer
ready, and type C (no echo will appear on the screen).
If you want to change the parameters : type P
Typing any other key will end the program
   If the profile is too long to be drawn on only one screen
page, a small dot
will appear at the bottom left of
the screen and you will hear a beep :
if you want to make a copy on the printer type C
if you want to change the parameters type P
if you want to see the next part of the profile type C
```



C
a

MIDDLE AMERICA TRENCH AXIS

Break points along the profile

Point No.	Latitude in degrees-minutes	Longitude in degrees-minutes	Latitude in decimal degrees	Longitude in decimal degrees	Deach in km	Dcumul in km	Distorg in km
1	18°46.2′	-105° 3.8′	18.770°	-105.063°	0.00	0.00	0.00
2	18°40.7′	-104°57.8′	18.678°	-104.963°	14.74	14.68	14.74
3	18°22.4′	-104°33.3′	18.373°	-104.555°	54.73	55.42	69.48
4	17°46.0′	-103°44.0′	17.767°	-103.733°	109.86	113.06	179.34
5	17°22.3′	-102°46.7′	17.372°	-102.778°	110.24	112.17	289.58
6	16°57.9′	-101°53.5′	16.965°	-101.892°	104.41	106.13	393.98
7	16°38.7′	-101°11.7′	16.645°	-101.195°	82.19	82.95	476.17
8	16°20.4′	-100°10.0′	16.340°	-100.167°	114.64	117.19	590.81
9	16° 2.3′	-99°21.5′	16.038°	-99.358°	92.45	94.12	683.26
10	15°46.4′	-98°39.3′	15.773°	-98.655°	80.58	83.18	763.84
11	15°29.7′	-97°47.6′	15.495°	-97.793°	97.22	101.30	861.06
12	15°17.6′	-97°13.8′	15.293°	-97.230°	64.31	68.81	925.37
13	15°13.1′	-95°48.8′	15.218°	-95.813°	152.07	154.15	1077.44
14	15° 0.4′	-95°20.0′	15.007°	-95.333°	56.47	57.12	1133.91
15	14°42.1′	-94°50.6′	14.702°	-94.843°	62.48	63.78	1196.39
16	14°10.1′	-93°59.7′	14.168°	-93.995°	108.74	111.28	1305.12
17	13°49.0′	-93°19.0′	13.817°	-93.317°	82.77	84.55	1387.90
18	13°29.0′	-92°40.0′	13.483°	-92.667°	79.28	81.37	1467.18
19	12°48.7′	-91°14.4′	12.812°	-91.240°	171.44	174.73	1638.62
20	12°30.3′	-90°32.5′	12.505°	-90.542°	82.95	83.91	1721.57
21	12°10.0′	-89°35.0′	12.167°	-89.583°	110.56	113.94	1832.13
22	11°46.6′	-88°38.8′	11.777°	-88.647°	110.61	111.70	1942.74
23	11°33.3′	-88°10.0′	11.555°	-88.167°	57.65	58.21	2000.39
24	10°50.0′	-87°13.1′	10.833°	-87.218°	130.78	132.36	2131.17
25	9°48.9′	-86°20.0′	9.815°	-86.333°	148.86	150.82	2280.03
26	9° 9.0′	-85°30.0′	9.150°	-85.500°	117.39	127.91	2397.42
27	8°39.3′	-84°25.5′	8.655°	-84.425°	130.13	134.99	2527.55
28	8°21.3′	-83°52.1′	8.355°	-83.868°	69.65	71.37	2597.20
29	8°10.7′	-83°22.6′	8.178°	-83.377°	57.40	58.81	2654.60
30	7°50.0′	-82°59.5′	7.833°	-82.992°	57.11	58.47	2711.71
31	7°19.0′	-82°50.0′	7.317°	-82.833°	59.91	60.76	2771.62
32	7°11.8′	-82°19.9′	7.197°	-82.332°	56.90	57.12	2828.52
33	6°56.6′	-81°51.8′	6.943°	-81.863°	58.73	62.75	2887.25
34	6°52.9′	-79°50.3′	6.882°	-79.838°	223.48	229.62	3110.73

b

MIDDLE AMERICA TRENCH AXIS

All the points of the profile

Point No.	Latitude in degrees-minutes	Longitude in degrees-minutes	Latitude in decimal degrees	Longitude in decimal degrees	Deach in km	Distorg in km	Depth in m
1	18°46.2′	-105° 3.8′	18.770°	-105.063°	0.00	0.00	5000
2	18°42.3′	-104°59.6′	18.705°	-104.993°	10.34	10.38	5000
3	18°41.1′	-104°58.2′	18.685°	-104.970°	3.32	13.72	4500
4	18°40.3′	-104°57.4′	18.672°	-104.957°	2.05	15.74	4500
5	18°40.0′	-104°57.3′	18.667°	-104.955°	0.58	16.31	4550
6	18°26.5′	-104°41.8′	18.442°	-104.697°	37.01	52.86	4550
7	18°23.9′	-104°36.8′	18.398°	-104.613°	10.04	62.77	4500
8	18°23.1′	-104°35.1′	18.385°	-104.585°	3.34	66.07	4450
9	18°21.7′	-104°31.5′	18.362°	-104.525°	6.85	72.73	4250
10	18°21.6′	-104°31.3′	18.360°	-104.522°	0.40	73.12	4300
11	18°20.0′	-104°29.6′	18.333°	-104.493°	4.22	77.22	4300
12	18°18.6′	-104°28.8′	18.310°	-104.480°	2.96	80.09	4200
13	18°15.6′	-104°27.9′	18.260°	-104.465°	5.79	85.71	4350
14	18°14.8′	-104°27.4′	18.247°	-104.457°	1.73	87.39	4350
15	18°13.4′	-104°26.5′	18.223°	-104.442°	3.04	90.35	4350
16	18°10.4′	-104°24.0′	18.173°	-104.400°	7.10	97.25	4400
17	18°10.1′	-104°23.7′	18.168°	-104.395°	0.77	97.99	4450
18	18° 8.9′	-104°22.2′	18.148°	-104.370°	3.46	101.35	4550
19	18° 8.5′	-104°21.0′	18.142°	-104.350°	2.24	103.53	4650
20	18° 6.5′	-104°17.4′	18.108°	-104.290°	7.35	110.68	4750
21	18° 3.4′	-104°10.0′	18.057°	-104.167°	14.27	124.55	4850
22	18° 0.7′	-104° 5.1′	18.012°	-104.085°	9.99	134.26	4850
23	17°59.9′	-104° 3.7′	17.998°	-104.062°	2.88	137.06	4850
24	17°59.4′	-104° 2.9′	17.990°	-104.048°	1.69	138.70	4900
25	17°56.1′	-103°56.8′	17.935°	-103.947°	12.39	150.74	4900
26	17°43.0′	-103°42.2′	17.717°	-103.703°	35.43	185.55	4900
27	17°39.6′	-103°35.1′	17.660°	-103.585°	14.05	199.36	4850
28	17°37.2′	-103°29.0′	17.620°	-103.483°	11.67	210.83	4800
29	17°35.9′	-103°25.3′	17.598°	-103.422°	6.98	217.69	4750
30	17°34.4′	-103°21.5′	17.573°	-103.358°	7.28	224.84	4750
31	17°24.3′	-102°55.0′	17.405°	-102.917°	50.48	274.45	4750
32	17°23.7′	-102°52.5′	17.395°	-102.875°	4.57	278.94	4700
33	17°23.1′	-102°50.0′	17.385°	-102.833°	4.57	283.43	4600
34	17°22.8′	-102°49.7′	17.380°	-102.828°	0.77	284.19	4500
35	17°20.9′	-102°42.6′	17.348°	-102.710°	13.06	297.04	4500
36	17°14.1′	-102°31.2′	17.235°	-102.520°	23.81	320.46	4550
37	17°12.3′	-102°28.8′	17.205°	-102.480°	5.41	325.78	4550
38	17°10.0′	-102°27.4′	17.167°	-102.457°	4.94	330.64	4550
39	17° 9.6′	-102°26.6′	17.160°	-102.443°	1.60	332.22	4550
40	17° 9.4′	-102°26.3′	17.157°	-102.438°	0.65	332.85	4500
41	16°59.5′	-101°59.6′	16.992°	-101.993°	50.78	382.81	4550
42	16°56.4′	-101°47.5′	16.940°	-101.792°	22.24	404.84	4800
43	16°54.8′	-101°43.8′	16.913°	-101.730°	7.21	411.98	4850
623	6°57.2′	-82° 1.8′	6.953°	-82.030°	18.97	2869.79	3050
624	6°56.3′	-81°57.8′	6.938°	-81.963°	7.55	2876.86	2900
625	6°56.6′	-81°55.0′	6.943°	-81.917°	5.19	2881.72	2850
626	6°56.7′	-81°54.1′	6.945°	-81.902°	1.67	2883.28	2800
627	6°56.6′	-81°51.8′	6.943°	-81.863°	4.24	2887.25	1300
628	6°56.4′	-81°49.7′	6.940°	-81.828°	3.88	2891.03	1500
629	6°54.2′	-81°46.9′	6.903°	-81.782°	6.58	2897.43	1650
630	6°52.0′	-81°44.6′	6.867°	-81.743°	5.88	2903.16	1700
631	6°51.0′	-81°42.2′	6.850°	-81.703°	4.79	2907.83	1750
632	6°50.9′	-81°38.8′	6.848°	-81.647°	6.27	2913.92	1850
633	6°50.4′	-81°36.3′	6.840°	-81.605°	4.70	2918.50	1900
634	6°50.0′	-81°35.0′	6.833°	-81.583°	2.51	2920.94	1950
635	6°49.0′	-81°32.6′	6.817°	-81.543°	4.80	2925.60	2000
636	6°47.9′	-81°30.0′	6.798°	-81.500°	5.21	2930.67	2050
637	6°46.9′	-81°25.9′	6.782°	-81.432°	7.78	2938.24	2150
638	6°46.8′	-81°23.1′	6.780°	-81.385°	5.16	2943.27	2250
639	6°46.6′	-81°21.5′	6.777°	-81.358°	2.97	2946.16	2300
640	6°46.6′	-81°19.6′	6.777°	-81.327°	3.50	2949.56	2450
641	6°46.3′	-81°17.6′	6.772°	-81.293°	3.73	2953.19	2500
642	6°46.1′	-81°14.9′	6.768°	-81.248°	4.99	2958.05	2550
643	6°46.0′	-81°12.4′	6.767°	-81.207°	4.61	2962.53	2750
644	6°46.2′	-81°10.7′	6.770°	-81.178°	3.15	2965.60	2900
645	6°46.2′	-81° 8.2′	6.770°	-81.137°	4.61	2970.09	3000
646	6°46.2′	-81° 3.7′	6.770°	-81.062°	8.29	2978.16	3200
647	6°45.7′	-81° 2.1′	6.762°	-81.035°	3.09	2981.16	3250
648	6°45.5′	-81° 1.6′	6.758°	-81.027°	0.99	2982.13	3300
649	6°46.0′	-80°59.4′	6.767°	-80.990°	4.16	2986.18	3350
650	6°52.9′	-79°50.3′	6.882°	-79.838°	127.98	3110.73	3350

c

MIDDLE AMERICA TRENCH AXIS

Circle Sections

Point No.	Latitude in degrees-minutes	Longitude in degrees-minutes	Latitude in decimal degrees	Longitude in decimal degrees	Distorg in Km
1	18°40.7′	-104°57.8′	18.678°	-104.963°	14.73
2	16°10.0′	-99°41.8′	16.167°	-99.697°	644.89
3	16°23.0′	-99°21.5′	16.383°	-99.358°	683.26
4	15°13.1′	-95°48.8′	15.218°	-95.813°	1077.44
5	15° 0.4′	-95°20.0′	15.007°	-95.333°	1133.91
6	11°33.3′	-88°10.0′	11.555°	-88.167°	2000.39
7	10°50.3′	-87°13.4′	10.838°	-87.223°	2130.40
8	8°21.3′	-83°52.1′	8.355°	-83.868°	2597.20
9	7°13.4′	-82°29.2′	7.223°	-82.487°	2810.27
10	6°52.9′	-79°50.3′	6.882°	-79.838°	3110.73

d

MIDDLE AMERICA TRENCH AXIS

Longitudes

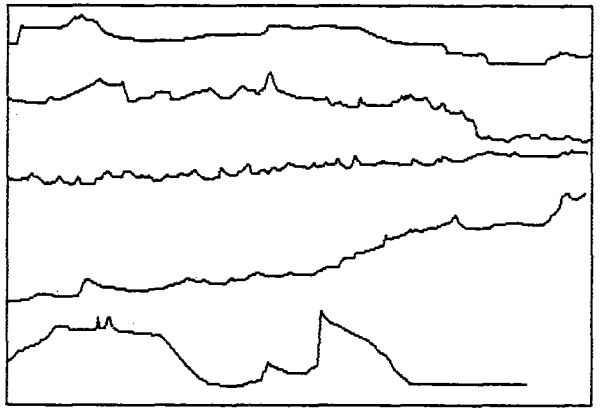
Point No.	Longitude in deg-min	Longitude decim. deg	Distorg in km
1	-105° 0.0′	-105.000°	9.39
2	-104° 0.0′	-104.000°	144.42
3	-103° 0.0′	-103.000°	265.09
4	-102° 0.0′	-102.000°	382.06
5	-101° 0.0′	-101.000°	497.40
6	-100° 0.0′	-100.000°	610.63
7	-99° 0.0′	-99.000°	725.24
8	-98° 0.0′	-98.000°	837.38
9	-97° 0.0′	-97.000°	949.86
10	-96° 0.0′	-96.000°	1056.72
11	-95° 0.0′	-95.000°	1175.97
12	-94° 0.0′	-94.000°	1304.26
13	-93° 0.0′	-93.000°	1425.93
14	-92° 0.0′	-92.000°	1549.75
15	-91° 0.0′	-91.000°	1667.31
16	-90° 0.0′	-90.000°	1784.83
17	-89° 0.0′	-89.000°	1901.13
18	-88° 0.0′	-88.000°	2021.38
19	-87° 0.0′	-87.000°	2166.20
20	-86° 0.0′	-86.000°	2329.97
21	-85° 0.0′	-85.000°	2455.63
22	-84° 0.0′	-84.000°	2579.68
23	-83° 0.0′	-83.000°	2710.45
24	-82° 0.0′	-82.000°	2872.98
25	-81° 0.0′	-81.000°	2985.07
26	-80° 0.0′	-80.000°	3093.25

e

MIDDLE AMERICA TRENCH AXIS

Screen copy of the profile

The drawing is 640 km wide.
It is 221 lines high, one interval between two lines represents 50 m.

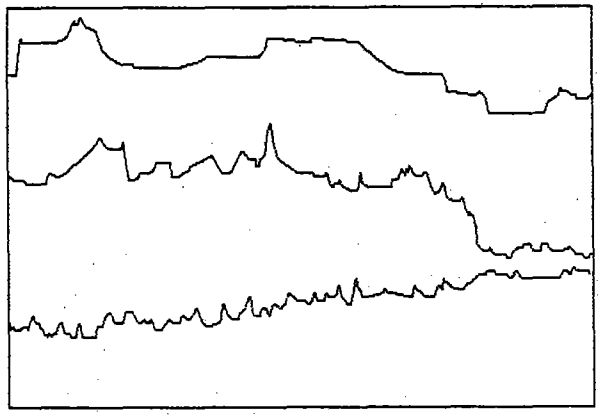


f

MIDDLE AMERICA TRENCH AXIS

Screen copy of the profile

The drawing is 640 km wide.
It is 221 lines high, one interval between two lines represents 25 m.

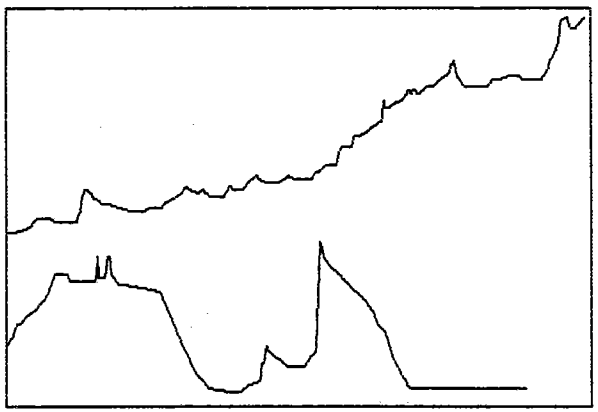


g

MIDDLE AMERICA TRENCH AXIS

Screen copy of the profile

The drawing is 640 km wide.
It is 221 lines high, one interval between two lines represents 25 m.



4.3

A

```
10 REM  ================================================================
15 REM
20 REM                          PROGRAM  MATPROFILE'
25 REM
30 REM                                        JACQUELINE ROUMP, AUG. 1984
35 REM  ================================================================
40 PAGE
45 INIT
50 SET DEGREES
55 CHARSIZE 4
60 PRINT "                       MAT PROFILE"
65 PRINT
70 PRINT
75 CHARSIZE 3
80 REM
85 REM     PRESENTATION
90 REM     ============
95 PRINT "PROGRAM TO DRAW THE MIDDLE AMERICA TRENCH AXIS PROFILE FROM "
100 PRINT "DATA PROCESSED ON A ZENITH 100 COMPUTER"
105 PRINT "THE PROFILE WILL BE DRAWN ON A BENSON 1332 PLOTTER, USING "
110 PRINT "COLORED OR BLACK PENS . IF BLACK PUT BLACK PENS IN THE 3"
115 PRINT "PENHOLDERS. IF IN COLOR, YOU MUST KNOW THAT :"
120 PRINT "  PEN 0 IS USED FOR THE AXES"
125 PRINT "  PEN 1 FOR THE TITLE, THE BREAKPOINTS AND THE CIRCLE LIMITS"
130 PRINT "  PEN 2 FOR THE PROFILE ITSELF"
135 PRINT "  PEN 2, AFTER CHANGED TO 4TH COLOR IS USED FOR THE COMMENTS"
140 PRINT "          ALONG THE PROFILE (MAIN FEATURES AND SURVEYS)"
145 REM
150 REM     DATA TABLES
155 REM     ===========
160 REM
165 REM     TABLE OF THE DATA DISTANCES AND DEPTHS ALONG THE PROFILE
170 REM     ------------------------------------------------------
175 NO=650
180 DIM A(NO,2)
185 DATA 0,5000,10.38,5000,13.72,4500,15.74,4500,16.31,4550,52.86,4550
190 DATA 62.77,4500,66.07,4450,72.73,4250,73.12,4300,77.22,4300
195 DATA 80.09,4200,85.71,4350,87.39,4350,90.35,4350,97.25,4400
200 DATA 97.99,4450,101.35,4550,103.53,4650,110.68,4750,124.55,4850
205 DATA 134.26,4850,137.06,4850,138.7,4900,150.74,4900,185.55,4900
210 DATA 199.36,4850,210.83,4800,217.69,4750,224.84,4750,274.45,4750
215 DATA 278.94,4700,283.43,4600,284.19,4500,297.04,4500,320.46,4550
220 DATA 325.78,4550,330.64,4550,332.22,4550,332.85,4500,382.81,4550
225 DATA 404.84,4800,411.98,4850,415.59,4900,421.71,4950,436.63,5000
230 DATA 474.84,5000,477.24,5050,479.12,5100,480.34,5250,483.01,5250
235 DATA 485.47,5250,507.55,5300,510.52,5300,515.26,5250,520.15,5300
240 DATA 522.59,5400,524.74,5550,536.69,5550,538.68,5550,586.35,5550
245 DATA 588.49,5500,589.63,5400,602.41,5300,604.32,5200,605.39,5250
250 DATA 610.44,5250,611.78,5250,614.78,5300,620.93,5300,621.46,5350
255 DATA 633.22,5350,638.03,5300,640.88,5300,644.89,5350,653.73,5350
260 DATA 658.8,5400,678.84,5400,680.97,5300,683.26,5250,691.91,5300
265 DATA 693.51,5300,699.93,5250,708.14,5150,714.88,5100,722.03,5000
270 DATA 730.94,4900,736.18,4800,738.08,4750,739.61,4750,743.88,4850
275 DATA 748.87,4900,757.98,4900,761.97,4900,763.84,4800,768.7,5300
280 DATA 770.25,5350,775.12,5350,780.61,5300,781.98,5250,790.86,5250
285 DATA 796.57,5200,798.75,5150,800.3,5100,804.99,5100,814.11,5100
290 DATA 814.65,5100,815.58,5300,825.21,5300,827.72,5250,836.16,5200
295 DATA 836.87,5150,837.9,5150,838.41,5150,861.06,5000,864.9,5100
300 DATA 869.05,5200,870.73,5250,879.22,5250,883.05,5200,891.26,5000
305 DATA 893.33,4950,896.26,4950,901.45,5050,905.76,5050,908.69,5050
```

•
•
•
•
•
•
•
•
•
•
•
•

```
700 DATA 2310.39,4300,2315.79,4250,2320.47,4200,2321.68,4200,2323,4200
705 DAT 2324.36,4150,2325.21,3900,2325.89,3900,2326.42,4000,2331.61,4000
710 DAT 2334.67,3950,2345.01,3900,2348.26,3850,2350.77,3800,2352.06,3750
715 DAT 2353.49,3750,2354.72,3800,2356.04,3800,2356.88,3750,2360.01,3750
720 DATA 2360.87,3800,2363.99,3800,2372.25,3700,2373.6,3700,2380.33,3700
725 DATA 2386.94,3600,2392.74,3550,2397.42,3500,2398.15,3450,2398.5,3400
730 DAT 2400.69,3400,2401.22,3350,2401.93,3350,2404.57,3350,2407.77,3600
735 DAT 2409.58,3650,2414.08,3700,2430.56,3700,2433.58,3700,2437.48,3700
740 DATA 2441.95,3650,2443.14,3600,2446,3600,2449.91,3600,2463.13,3550
745 DATA 2471.86,3550,2476.89,3600,2497.1,3600,2501.58,3500,2504.78,3450
750 DATA 2506.42,3350,2508.56,3300,2512.74,3200,2513.89,3150,2516.2,3100
755 DAT 2518.51,2800,2524.54,2750,2525.74,2750,2526.51,2800,2528.68,2900
760 DAT 2535.33,2900,2536.86,2850,2540.56,2800,2543.85,2750,2552.21,2600
765 DAT 2554.24,2500,2555.85,2450,2559.72,2450,2565.05,2350,2577.24,2250
770 DATA 2579.09,2200,2582.06,2150,2586.15,2100,2589.75,2000,2597.2,1750
775 DAT 2599.54,1750,2605.61,1750,2610.92,1750,2613.02,1850,2640.82,1850
780 DATA 2642.52,1750,2643.49,1500,2645.19,1800,2652.03,1800,2654.6,1500
785 DAT 2655.97,1500,2656.87,1500,2658.85,1750,2667.13,1900,2672.95,1900
790 DAT 2711.71,2000,2736.96,2750,2751.92,3150,2760.18,3250,2765.04,3350
795 DAT 2787.57,3400,2799.37,3400,2810.27,3300,2822.18,3250,2823.84,3000
800 DAT 2825.71,3000,2828.52,2750,2830.15,2800,2837.32,2900,2842.37,2950
805 DAT 2852.04,3050,2869.79,3050,2876.86,2900,2881.72,2850,2883.28,2800
810 DAT 2887.25,1300,2891.03,1500,2897.43,1650,2903.16,1700,2907.83,1750
815 DATA 2913.92,1850,2918.5,1900,2920.94,1950,2925.6,2000,2930.67,2050
820 DAT 2938.24,2150,2943.27,2250,2946.16,2300,2949.56,2450,2953.19,2500
825 DATA 2958.05,2550,2962.53,2750,2965.6,2900,2970.09,3000,2978.16,3200
830 DATA 2981.16,3250,2982.13,3300,2986.18,3350,3110.73,3350
835 READ A
840 REM
845 REM    TABLE OF BREAK POINTS NUMBERS AND DISTANCES ALONG THE PROFILE
850 REM    ------------------------------------------------------------
855 N1=34
860 DIM B(N1,2)
865 DATA 1,0,2,14.74,3,69.48,4,179.34,5,289.58,6,393.98,7,476.17
870 DATA 8,590.81,9,683.26,10,763.84,11,861.06,12,925.37,13,1077.44
875 DATA 14,1133.91,15,1196.39,16,1305.12,17,1387.9,18,1467.18
880 DATA 19,1638.62,20,1721.57,21,1832.13,22,1942.74,23,2000.39
885 DATA 24,2131.17,25,2280.03,26,2397.42,27,2527.55,28,2597.2
890 DATA 29,2654.6,30,2711.71,31,2771.62,32,2828.52,33,2887.25
895 DATA 34,3110.73
900 READ B
905 REM
```

```
910 REM  TABLE OF THE LONGITUDES NUMBERS AND DISTANCES ALONG THE PROFILE
915 REM  ------------------------------------------------------------
920 N2=26
925 DIM C(N2,2)
930 DATA -105,9.39,-104,144.42,-103,265.09,-102,382.06,-101,497.4
935 DATA -100,610.63,-99,725.24,-98,837.38,-97,949.86,-96,1056.72
940 DATA -95,1175.97,-94,1304.26,-93,1425.93,-92,1549.75,-91,1667.31
945 DATA -90,1784.83,-89,1901.13,-88,2021.38,-87,2166.2,-86,2329.97
950 DATA -85,2455.63,-84,2579.68,-83,2710.45,-82,2872.98,-81,2985.07
955 DATA -80,3093.25
960 READ C
965 REM
970 REM     TABLE OF THE CIRCLE SECTIONS LIMITS NUMBERS AND DISTANCES
975 REM     ---------------------------------------------------------
980 N3=10
985 DIM D(N3,2)
990 DATA 1,14.73,2,644.89,3,683.26,4,1077.44,5,1133.91,6,2000.39
995 DATA 7,2130.4,8,2597.2,9,2810.27,10,3110.73
1000 READ D
1010 REM ------------------------------------------------------------
1020 REM
1030 REM     INITIALIZATION OF THE BENSON PLOTTER
1040 REM     ===================================
1050 PRINT
1060 PRINT "PLOTTER READY (Y/N) ? ";
1070 INPUT R$
1080 R$=SEG(R$,1,1)
1090 IF R$<>"Y" THEN 1060
1100 GOSUB 5090
1110 Z1=A(1,1)/100
1120 Z0=A(1,2)/1000
1130 Z2=0
1140 REM
1150 REM     GOING TO THE FIRST POINT OF THE PROFILE POSITION IN CM
1160 REM     -----------------------------------------------------
1170 GOSUB 5310
1180 REM
1190 REM     INITIALIZATION OF THE PEN
1200 REM     -------------------------
1210 Z2=1
1220 REM
1230 REM     INITIALIZATION OF WRITING TYPE AND ORIENTATION
1240 REM     ---------------------------------------------
1250 Z6=2
1260 GOSUB 5840
1270 REM
1280 REM     DRAWING OF THE PROFILE
1290 REM     ======================
1300 FOR I=1 TO NO
1310 Z1=A(I,1)/100
1320 Z0=A(I,2)/1000
1330 GOSUB 5310
1340 NEXT I
1350 Z6=0
1360 GOSUB 5840
1370 REM
1380 REM     TOP HORIZONTAL AXIS
1390 REM     ===================
1400 Z4=0.2
1410 Z5=0.3
1420 Z9=90
1430 GOSUB 5700
1440 X0=0
1450 Y0=0
1460 L=A(NO,1)/100
1470 V0=0
1480 V1=100
1490 G0=1
1500 T$="DISTANCE IN KM"
1510 I=1
1520 Z7=-1
1530 REM   DRAWING OF THE AXIS WITH ABOVE PARAMETERS
1540 REM
1550 GOSUB 6520
1560 REM
1570 REM     LEFT VERTICAL AXIS
1580 REM     ==================
1590 X0=0
1600 Y0=0
1610 L=10000/1000
1620 V0=0
1630 V1=1000
1640 G0=1
1650 T$="DEPTH IN M"
1660 I=0
1670 Z7=-1
1680 REM   DRAWING OF THE AXIS WITH ABOVE PARAMETERS
1690 REM
1700 GOSUB 6520
1710 REM
1720 REM     BOTTOM HORIZONTAL AXIS
1730 REM     ======================
1740 Z2=0
1750 Z0=10000/1000
1760 Z1=0
1770 REM
1780 REM     GOING TO THE FIRST POINT OF THE AXIS
1790 REM     -----------------------------------
1800 GOSUB 5310
1810 REM
1820 REM     DRAWING THE AXIS, THE TICK MARKS AND PRINTING THE LABELS
1830 REM     -------------------------------------------------------
1840 FOR K=1 TO N2
1850 Z1=C(K,2)/100
1860 Z2=1
1870 GOSUB 5310
1880 Z0=10000/1000+0.2
1890 GOSUB 5310
1900 IF C(K,1)/2=INT(C(K,1)/2) THEN 1980
1910 Z2=0
1920 Z8=C(K,1)
1930 Y$=STR(Z8)
1940 Z0=10000/1000+0.2+1.5*Z5
1950 Z1=C(K,2)/100-LEN(Y$)/2*Z4
1960 GOSUB 5310
1970 GOSUB 5930
1980 Z0=10000/1000
1990 Z1=C(K,2)/100
2000 GOSUB 5310
2010 NEXT K
```

```
2020 REM
2030 REM     DRAWING THE END OF THE AXIS
2040 REM     --------------------------
2050 Z2=1
2060 Z1=A(NO,1)/100
2070 GOSUB 5310
2080 REM
2090 REM     PRINTING OF THE TITLE
2100 REM     ---------------------
2110 Z$="LONGITUDES"
2120 Z3=LEN(Z$)
2130 ZO=10000/1000+0.2+4*Z5
2140 Z1=Z1-Z3*Z4
2150 Z2=0
2160 GOSUB 5310
2170 GOSUB 5600
2180 ZO=XO
2190 Z1=YO
2200 Z2=0
2210 REM
2220 REM     RIGHT VERTICAL AXIS
2230 REM     ===================
2240 Z2=0
2250 GO=1
2260 L=10000/1000
2270 Z1=A(NO,1)/100
2280 ZO=0
2290 REM
2300 REM     GOING TO THE ORIGIN OF THE AXIS
2310 REM     ------------------------------
2320 GOSUB 5310
2330 Z(15)=INT(1.5/GO)+1
2340 J=INT(L/GO)
2350 REM
2360 REM     DRAWING THE AXIS AND TICK MARKS, AND PRINTING THE LABELS
2370 REM     -------------------------------------------------------
2380 FOR K=1 TO J+1
2390 ZO=XO+(K-1)*GO
2400 Z2=1
2410 GOSUB 5310
2420 Z1=A(NO,1)/100+0.2
2430 GOSUB 5310
2440 IF K-1-INT((K-1)/Z(15))*Z(15)<>0 THEN 2510
2450 Z8=VO+V1*(K-1)
2460 Z$=STR(Z8)
2470 Z1=Z1+Z4
2480 Z2=0
2490 GOSUB 5310
2500 GOSUB 5930
2510 Z1=A(NO,1)/100
2520 GOSUB 5310
2530 NEXT K
2540 REM
2550 REM     DRAWING THE END OF THE AXIS
2560 REM     --------------------------
2570 Z2=1
2580 ZO=XO+L
2590 GOSUB 5310
2600 REM
2610 REM     PRINTING OF THE TITLE
2620 REM     ---------------------
2630 T$="M"
2640 Z1=Z1+(LEN(T$)+LEN(Z$)+4)*Z4
2650 Z2=0
2660 GOSUB 5310
2670 Z$=T$
2680 Z3=LEN(Z$)
2690 GOSUB 5600
2700 ZO=XO
2710 Z1=YO
2720 REM
2730 REM     BREAK POINTS TICK MARKS DRAWING AND LABELS PRINTING
2740 REM     ==================================================
2750 Z2=0
2760 Z5=0.25
2770 GOSUB 5700
2780 Z6=1
2790 GOSUB 5840
2800 I=1
2810 ZO=8000/1000
2820 Z1=0
2830 GOSUB 5310
2840 FOR K=2 TO N1
2850 Z2=0
2860 ZO=8000/1000
2870 Z8=B(K-1,1)-1
2880 Y$=STR(Z8)
2890 IF (LEN(Y$)+1)*Z4>(B(K,2)-B(K-1,2))/100 THEN 2930
2900 Z1=B(K-1,2)/100+(B(K,2)-B(K-1,2))/2/100-LEN(Y$)/2*Z4
2910 GOSUB 5310
2920 GOSUB 5930
2930 Z1=B(K,2)/100
2940 GOSUB 5310
2950 Z2=1
2960 ZO=8000/1000-0.25
2970 GOSUB 5310
2980 ZO=8000/1000
2990 GOSUB 5310
3000 NEXT K
3010 Z4=0.25
3020 Z5=0.3
3030 GOSUB 5700
3040 Z$="EARTHQUAKES SECTIONS AND BREAK POINTS"
3050 Z3=LEN(Z$)
3060 ZO=8000/1000+3*Z5
3070 Z1=Z1-(Z3+1)*Z4
3080 Z2=0
3090 GOSUB 5310
3100 GOSUB 5600
3110 ZO=XO
3120 Z1=YO
3130 Z2=0
3140 Z4=0.2
3150 GOSUB 5700
3160 REM
3170 REM     CIRCLE SECTIONS TICK MARKS DRAWING AND NAME PRINTING
3180 REM     ===================================================
3190 Z2=0
3200 ZO=1000/1000
3210 Z1=0
```

```
3220 GOSUB 5310
3230 FOR K=1 TO N3
3240 Z0=1000/1000
3250 Z1=D(K,2)/100
3260 GOSUB 5310
3270 Z2=1
3280 Z0=1000/1000-0.3
3290 GOSUB 5310
3300 Z0=1000/1000
3310 GOSUB 5310
3320 Z2=0
3330 IF K/2=INT(K/2) THEN 3490
3340 GO TO (K+1)/2 OF 3350,3370,3390,3410,3430
3350 Z$="NORTHERN SET"
3360 GO TO 3440
3370 Z$="NORTHERN MIDDLE SET"
3380 GO TO 3440
3390 Z$="SOUTHERN MIDDLE SET"
3400 GO TO 3440
3410 Z$="SOUTHERN SET"
3420 GO TO 3440
3430 Z$="PANAMA FZ-80W"
3440 Z3=LEN(Z$)
3450 Z0=1000/1000
3460 Z1=D(K,2)/100+(D(K+1,2)-D(K,2))/2/100-Z3/2*Z4
3470 GOSUB 5310
3480 GOSUB 5600
3490 NEXT K
3500 Z4=0.25
3510 GOSUB 5700
3520 Z2=0
3530 Z$="CIRCLES SECTIONS"
3540 Z0=1000/1000+3*Z5
3550 Z3=LEN(Z$)
3560 Z1=Z4*2
3570 GOSUB 5310
3580 GOSUB 5600
3590 REM
3600 REM     TITLE PRINTING (DOUBLED)
3610 REM     ========================
3620 Z0=X0
3630 Z1=Y0
3640 Z2=0
3650 Z$="MIDDLE AMERICA TRENCH AXIS"
3660 Z3=LEN(Z$)
3670 Z4=0.35
3680 Z5=0.45
3690 GOSUB 5700
3700 Z0=-4*Z5
3710 GOSUB 4880
3720 Z1=Z1+0.02
3730 GOSUB 4880
3740 Z4=0.2
3750 Z5=0.25
3760 GOSUB 5700
3770 Z0=0
3780 Z1=0
3790 GOSUB 5310
3800 REM
3810 REM     MAIN FEATURES POSITIONS
3820 REM     ======================
3830 A$=CHR(13)
3840 PRINT A$
3850 PRINT "CHANGE PEN IN PENHOLDER #2 INTO THE 4TH COLOR"
3860 INPUT A$
3870 Z6=2
3880 GOSUB 5840
3890 Z0=5000/1000
3900 Z$="FRACTURE ZONE"
3910 Z3=LEN(Z$)
3920 Z1=1150/100-Z3/2*Z4
3930 Z2=0
3940 GOSUB 4880
3950 Z$="TEHUANTEPEC"
3960 Z3=LEN(Z$)
3970 Z0=5000/1000-2*Z5
3980 Z1=1150/100-Z3/2*Z4
3990 GOSUB 4880
4000 Z$="COCOS"
4010 Z3=LEN(Z$)
4020 Z0=3200/1000
4030 Z1=2630/100-Z3/2*Z4
4040 GOSUB 4880
4050 Z$="RIDGE"
4060 Z3=LEN(Z$)
4070 Z0=3200/1000+2*Z5
4080 Z1=2630/100-Z3/2*Z4
4090 GOSUB 4880
4100 Z0=3500/1000+2*Z5
4110 Z1=2920/100-Z3/2*Z4
4120 GOSUB 4880
4130 Z$="COIBA"
4140 Z3=LEN(Z$)
4150 Z0=3500/1000
4160 Z1=2920/100-Z3/2*Z4
4170 GOSUB 4880
4180 REM
4190 REM     MAIN SURVEYS POSITIONS
4200 REM     ======================
4210 REM
4220 REM     PRINTING OF THE NAMES OF THE SURVEYS
4230 REM     ------------------------------------
4240 Z$="MEXICO"
4250 Z3=LEN(Z$)
4260 Z0=4600/1000-4*Z5
4270 Z1=795/100-Z3/2*Z4
4280 GOSUB 4880
4290 Z$="GUATEMALA"
4300 Z3=LEN(Z$)
4310 Z0=5800/1000-4*Z5
4320 Z1=1685/100-Z3/2*Z4
4330 GOSUB 4880
4340 Z$="COSTA RICA II"
4350 Z3=LEN(Z$)
4360 Z0=4700/1000
4370 Z1=2300/100
4380 GOSUB 4880
4390 Z$="COSTA RICA I"
4400 Z3=LEN(Z$)
4410 Z0=3500/1000-4*Z5
```

```
4420 Z1=2400/100-Z3/2*Z4
4430 GOSUB 4880
4440 Z$="SURVEY"
4450 Z3=LEN(Z$)
4460 Z0=3500/1000-2*Z5
4470 Z1=2400/100-Z3/2*Z4
4480 GOSUB 4880
4490 Z0=4700/1000+2*Z5
4500 Z1=2315/100+3*Z4
4510 GOSUB 4880
4520 Z0=5800/1000-2*Z5
4530 Z1=1685/100-Z3/2*Z4
4540 GOSUB 4880
4550 Z0=4600/1000-2*Z5
4560 Z1=795/100-Z3/2*Z4
4570 GOSUB 4880
4580 REM
4590 REM     DRAWING OF THE LIMITS OF THE SURVEYS
4600 REM     -----------------------------------
4610 Z0=4600/1000
4620 Z1=710/100
4630 GOSUB 4930
4640 Z1=880/100
4650 GOSUB 4930
4660 Z0=5800/1000
4670 Z1=1650/100
4680 GOSUB 4930
4690 Z1=1720/100
4700 GOSUB 4930
4710 Z0=3800/1000
4720 Z1=2300/100
4730 GOSUB 4930
4740 Z1=2330/100
4750 GOSUB 4930
4760 Z0=3500/1000
4770 Z1=2350/100
4780 GOSUB 4930
4790 Z1=2450/100
4800 GOSUB 4930
4810 Z2=0
4820 Z1=0
4830 Z0=0
4840 GOSUB 5310
4850 END
4860 REM ------------------------------------------------------------------
4870 REM
4880 REM     SUBROUTINES
4890 REM     ===========
4900 GOSUB 5310
4910 GOSUB 5600
4920 RETURN
4930 REM ---------------
4940 Z2=0
4950 GOSUB 5310
4960 Z2=1
4970 Z0=Z0-0.2
4980 GOSUB 5310
4990 Z0=Z0+0.2
5000 GOSUB 5310
5010 RETURN
```

```
5020 REM ------------------------------------------------------------------
5030 REM
5040 REM     BENSON PLOTTER SUBROUTINES
5050 REM     ==========================
5060 REM
5070 REM     INITIALIZATION OF THE BENSON 1332 PLOTTER
5080 REM     -----------------------------------------
5090 DIM Z(25),Z$(80)
5100 ON SRQ THEN 5290
5110 PRINT "BENSON ADDRESS ? ",
5120 INPUT Z(25)
5130 PRINT @Z(25),"76 !",
5140 Z$=CHR(34)
5150 PRINT @Z(25),Z$;"#$%&'()*+,-./0123456789:;<=>?",
5160 PRINT @Z(25),"@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]↑←",
5170 PRINT @Z(25), USING "A/L",Z$
5180 PRINT @Z(25),"76                                    ",
5190 PRINT @Z(25),"                                      ",
5200 PRINT @Z(25), USING "A/L",Z$
5210 Z(1)=0
5220 Z(2)=0
5230 Z(3)=200
5240 Z(5)=60
5250 Z(6)=0
5260 Z(13)=0
5270 Z(20)=Z(3)
5280 Z(21)=Z(3)
5290 RETURN
5300 REM
5310 REM     MOVING THE PEN
5320 REM     --------------
5330 Z(7)=INT(Z0*Z(20))
5340 Z(8)=INT(Z1*Z(21))
5350 Z(9)=Z(7)-Z(1)
5360 Z(10)=Z(8)-Z(2)
5370 Z(1)=Z(7)
5380 Z(2)=Z(8)
5390 Z(7)=ABS(Z(9))
5400 Z(8)=ABS(Z(10))
5410 IF Z(7)>16383 THEN 5500
5420 IF Z(8)>16383 THEN 5500
5430 Z(7)=0
5440 Z(11)=Z(9)
5450 Z(12)=Z(10)
5460 PRINT @Z(25),"PL ";Z(11);Z(12);Z2;
5470 PRINT @Z(25), USING "A/L",Z$
5480 IF Z(7)>1 THEN 5390
5490 RETURN
5500 Z(7)=INT((Z(7)+16383)/16384)
5510 Z(8)=INT((Z(8)+16383)/16384)
5520 IF Z(7)>Z(8) THEN 5540
5530 Z(7)=Z(8)
5540 Z(11)=INT(Z(9)/Z(7))
5550 Z(12)=INT(Z(10)/Z(7))
5560 Z(9)=Z(9)-Z(11)
5570 Z(10)=Z(10)-Z(12)
5580 GO TO 5460
5590 REM
5600 REM     WRITING A STRING OF CHARACTERS
5610 REM     ------------------------------
```

```
5620 PRINT @Z(25),"TX ";Z3;" ";Z$;
5630 PRINT @Z(25), USING "A/L",";" "
5640 Z(1)=Z(1)+Z(5)*Z3
5650 Z(2)=Z(2)+Z(6)*Z3
5660 PRINT @Z(25),"NT ";Z(5)*Z3;" ";Z(6)*Z3;
5670 PRINT @Z(25), USING "A/L",";" "
5680 RETURN
5690 REM
5700 REM      INITIALIZATION OF CHARACTER TYPE AND ORIENTATION
5710 REM      ------------------------------------------------
5720 Z(5)=Z4*COS(Z9)
5730 Z(6)=Z4*SIN(Z9)
5740 Z(16)=INT(4*Z(5)*Z(20)/3)
5750 Z(17)=INT(-4*Z5*Z(20)*SIN(Z9)/3)
5760 Z(18)=INT(4*Z(6)*Z(21)/3)
5770 Z(19)=INT(4*Z5*Z(21)*COS(Z9)/3)
5780 PRINT @Z(25),"SZ ";Z(16);" ";Z(17);" ";Z(18);" ";Z(19);
5790 PRINT @Z(25), USING "A/L",";" "
5800 Z(5)=INT(Z(5)*Z(20))
5810 Z(6)=INT(Z(6)*Z(21))
5820 RETURN
5830 REM
5840 REM      INITIALIZATION OF A PEN
5850 REM      -----------------------
5860 Z(14)=INT(ABS(Z6))-3*INT(INT(ABS(Z6))/3)
5870 IF Z(14)=Z(13) THEN 5910
5880 PRINT @Z(25),"NP ";Z(14);
5890 PRINT @Z(25), USING "A/L",";" "
5900 Z(13)=Z(14)
5910 RETURN
5920 REM
5930 REM      WRITING OF NUMBERS
5940 REM      ------------------
5950 Z7=INT(Z7)
5960 Z(9)=ABS(INT(Z7))
5970 IF Z7>-2 THEN 6150
5980 IF Z(9)<10 THEN 6000
5990 Z(9)=9
6000 Z$=STR(Z(9))
6010 Z$=Z$&"E"
6020 Z(9)=Z(9)+7
6030 IF Z8>0 THEN 6070
6040 Z(9)=Z(9)+1
6050 PRINT @Z(25),"TX ";Z(9);" ";
6060 GO TO 6080
6070 PRINT @Z(25),"TX ";Z(9);
6080 PRINT @Z(25), USING Z$,Z8;
6090 PRINT @Z(25), USING "A/L",";" "
6100 Z(1)=Z(1)+Z(9)*Z(5)
6110 Z(2)=Z(2)+Z(9)*Z(6)
6120 PRINT @Z(25),"NT ";Z(9)*Z(5);" ";Z(9)*Z(6);
6130 PRINT @Z(25), USING "A/L",";" "
6140 RETURN
6150 Z(7)=INT(LGT(ABS(Z8)+0.5))+1
6160 IF Z(7)>0 THEN 6180
6170 Z(7)=1
6180 IF Z8=>0 THEN 6200
6190 Z(7)=Z(7)+1
6200 Z$=STR(Z(7))
6210 Z$=Z$&"D"
6220 IF Z7>-1 THEN 6250
6230 Z(9)=Z(7)
6240 GO TO 6050
6250 Z(8)=LEN(Z$)+2
6260 Z$=Z$&". D"
6270 IF Z(9)<10 THEN 6290
6280 Z(9)=9
6290 GO TO Z(9)+1 OF 6300,6320,6340,6360,6380,6400,6420,6440,6460,6480
6300 Z$=REP(" ",Z(8)+1,1)
6310 GO TO 6490
6320 Z$=REP("1",Z(8),1)
6330 GO TO 6490
6340 Z$=REP("2",Z(8),1)
6350 GO TO 6490
6360 Z$=REP("3",Z(8),1)
6370 GO TO 6490
6380 Z$=REP("4",Z(8),1)
6390 GO TO 6490
6400 Z$=REP("5",Z(8),1)
6410 GO TO 6490
6420 Z$=REP("6",Z(8),1)
6430 GO TO 6490
6440 Z$=REP("7",Z(8),1)
6450 GO TO 6490
6460 Z$=REP("8",Z(8),1)
6470 GO TO 6490
6480 Z$=REP("9",Z(8),1)
6490 Z(9)=Z(9)+Z(7)+1
6500 GO TO 6050
6510 REM
6520 REM      AXIS DRAWING
6530 REM      ------------
6540 REM
6550 Z0=X0
6560 Z1=Y0
6570 Z2=0
6580 GOSUB 5310
6590 Z(15)=INT(1.5/G0)+1
6600 J=INT(L/G0)
6610 FOR K=1 TO J+1
6620 Z0=X0+(K-1)*(1-I)*G0
6630 Z1=Y0+(K-1)*I*G0
6640 Z2=1
6650 GOSUB 5310
6660 Z0=Z0-I*0.2
6670 Z1=Z1-(1-I)*0.2
6680 GOSUB 5310
6690 IF K-1-INT((K-1)/Z(15))*Z(15)<>0 THEN 6770
6700 Z8=V0+V1*(K-1)
6710 Y$=STR(Z8)
6720 Z0=X0+(K-1)*(1-I)*G0-I*2*Z4
6730 Z1=Y0+(K-1)*I*G0-(1-I)*(LEN(Y$)+2)*Z4-I*(LEN(Y$)/2)*Z4
6740 Z2=0
6750 GOSUB 5310
6760 GOSUB 5930
6770 Z0=X0+(K-1)*(1-I)*G0
6780 Z1=Y0+(K-1)*I*G0
6790 GOSUB 5310
6800 NEXT K
6810 Z2=1
```

```
6820 Z0=X0+(1-I)*L
6830 Z1=Y0+I*L
6840 GOSUB 5310
6850 Z0=Z0-I*LEN(T$)*0.1
6860 Z1=Z1-I*LEN(T$)*Z4-(1-I)*(LEN(T$)+LEN(Y$)+3)*Z4
6870 Z2=0
6880 GOSUB 5310
6890 Z$=T$
6900 Z3=LEN(Z$)
6910 GOSUB 5600
6920 Z0=X0
6930 Z1=Y0
6940 Z2=0
6950 RETURN
```

B
★

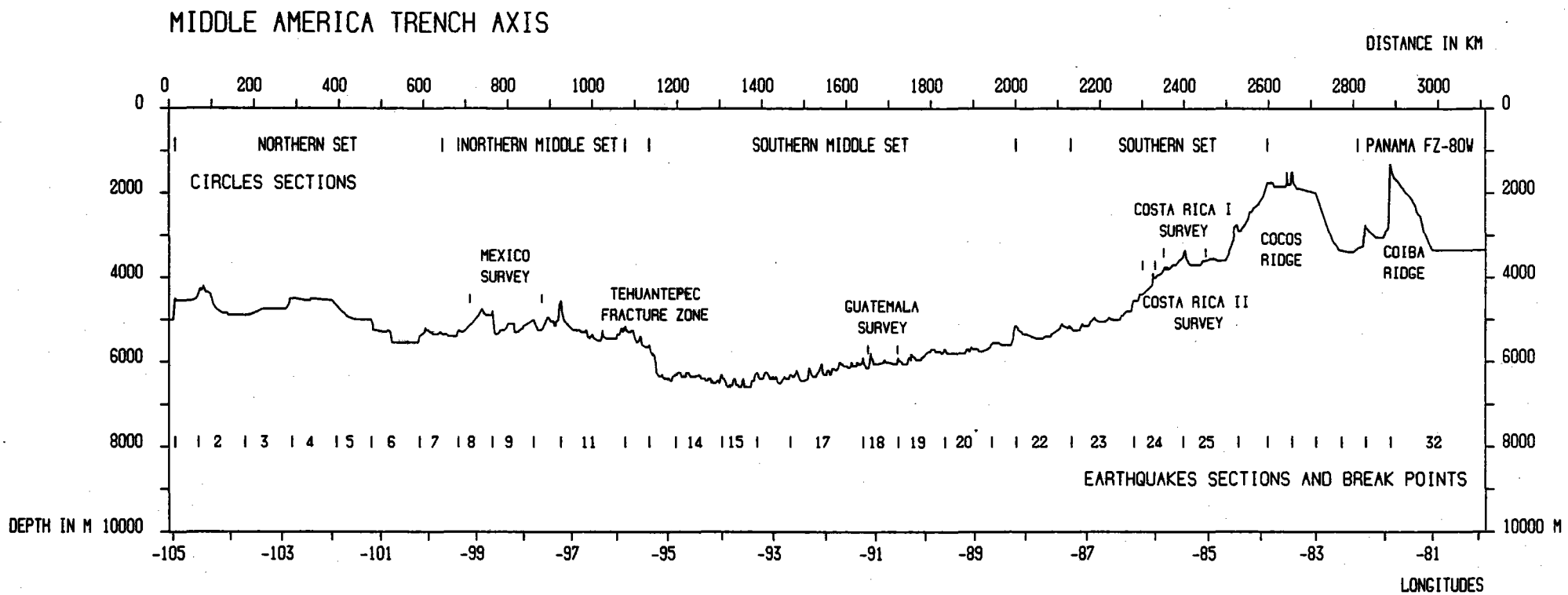
MAT PROFILE

PROGRAM TO DRAW THE MIDDLE AMERICA TRENCH AXIS PROFILE FROM
DATA PROCESSED ON A ZENITH 100 COMPUTER
THE PROFILE WILL BE DRAWN ON A BENSON 1332 PLOTTER, USING
COLORED OR BLACK PENS : IF BLACK PUT BLACK PENS IN THE 3
PENHOLDERS. IF IN COLOR, YOU MUST KNOW THAT :
  PEN 0 IS USED FOR THE AXES
  PEN 1 FOR THE TITLE, THE BREAKPOINTS AND THE CIRCLE LIMITS
  PEN 2 FOR THE PROFILE ITSELF
  PEN 2. AFTER CHANGED TO 4TH COLOR IS USED FOR THE COMMENTS
        ALONG THE PROFILE (MAIN FEATURES AND SURVEYS)

PLOTTER READY (Y/N) ? Y
BENSON ADDRESS ? 41

CHANGE PEN IN PENHOLDER #2 INTO THE 4TH COLOR

C


MIDDLE AMERICA TRENCH AXIS
DISTANCE IN KM
0
200
400
600
800
1000
1200
1400
1600
1800
2000
2200
2400
2600
2800
3000
NORTHERN SET
INORTHERN MIDDLE SET
SOUTHERN MIDDLE SET
SOUTHERN SET
PANAMA FZ-80W
CIRCLES SECTIONS
MEXICO SURVEY
TEHUANTEPEC FRACTURE ZONE
GUATEMALA SURVEY
COSTA RICA I SURVEY
COSTA RICA II SURVEY
COCOS RIDGE
COIBA RIDGE
2000
4000
6000
8000
10000
1 2 3 4 5 6 7 8 9 11 14 15 17 18 19 20 22 23 24 25 32
EARTHQUAKES SECTIONS AND BREAK POINTS
DEPTH IN M
10000 M
-105
-103
-101
-99
-97
-95
-93
-91
-89
-87
-85
-83
-81
LONGITUDES

RECHERCHE DE CENTRES DE CERCLES SUR LA SPHERE

5.1 - RECHERCHE EN UTILISANT LA DISTANCE

A - Exemple de fichier de données

B - Listing du programme 'DISSMCRC'

C - Copie d'écran d'un exemple d'exécution du programme 'DISSMCRC'*

D - Exemple de sortie sur imprimante*

5.2 - RECHERCHE EN UTILISANT LA DIRECTION DES STRUCTURES

A - Exemple de fichier de données

B - Listing du programme 'DIRSMCRC'

C - Copie d'écran d'un exemple d'exécution du programme 'DIRSMCRC'+

D - Exemple de sortie sur imprimante+

* et + : les lettres a à c de la partie C correspondent à des sorties sur imprimante (partie D) portant le même nom.

5.1

A

```
500 ' ============================================================
510 '
520 '                          DATA `DSSCMAT1'
530 '
540 '                                        Jacqueline ROUMP, Oct. 1984
550 ' ============================================================
560 '
570 '    These data are to be used by the program DISSMCRC.
580 '    They are the latitude and longitude of the first set of points along
590 ' the Middle America Trench : from off Manzanillo to SE of Acapulco
600 '
610 N=14                                    'number of points
620 DATA 18.67,-104.97
630 DATA 18.45,-104.67
640 DATA 18.21,-104.45
650 DATA 18.07,-104.17
660 DATA 17.83,-103.75
670 DATA 17.62,-103.33
680 DATA 17.37,-102.83
690 DATA 17.17,-102.33
700 DATA 16.94,-101.83
710 DATA 16.83,-101.5
720 DATA 16.67,-101.17
730 DATA 16.5,-100.67
740 DATA 16.33,-100.17
750 DATA 16.17,-99.67
```

B

```
10 ' ============================================================
20 '
30 '                          PROGRAM `DISSMCRC'
40 '
50 '                                        Jacqueline ROUMP, Oct.1984
60 ' ============================================================
70 '
80 CLS : CLEAR : OPTION BASE 1
90 '
100 '     PRESENTATION
110 '     ============
120 PRINT " This program has been written to help you to find the pole"
130 PRINT "fitting best to points distributed on a small circle."
140 PRINT
150 PRINT " The program will read the data in a data file beginning with line"
160 PRINT "500, and having a variable named N giving the number of data"
170 PRINT "points. Then each data line must have first the latitude of the"
180 PRINT "point, then the longitude. The last line number must not exceed"
190 PRINT "999."
200 PRINT
210 PRINT " The program will ask you the latitude and longitude of the pole"
220 PRINT "you have chosen to test. Then it will compute the sum of the"
230 PRINT "squares of the differences between the real distance and the mean"
240 PRINT "distance from the pole to each point. This value must be as close"
250 PRINT "to zero as possible, so the program will loop and test points from"
260 PRINT "2 degrees to 0.01 degree to the point having the smallest"
270 PRINT "deviation. The results will be verified and a graph will be"
280 PRINT "on the screen, and you will be allowed to copy this graph on the"
290 PRINT "printer."
300 PRINT
310   INPUT "Disk ready ";R$ : R$=LEFT$(R$,1)
320       IF R$()"Y" AND R$()"y" THEN 310
330 PRINT "Name of the data files on the disk :"
340 FILES "*.DAT"                           'list of the disk data files
350 PRINT
360   INPUT "Name of your data file : ";X$
370 CHAIN MERGE X$,500,ALL                  'merging the data file
380 '  ---------------------------------------------------------------------
390 '    PLACE FOR THE DATA FILE
1000 ' ---------------------------------------------------------------------
1010 '
1020 '     READING OF THE DATA FILE AND QUESTIONS
1030 '     ======================================
1040 DIM T(N,4)                             'data table
1050 '---------------
1060 FOR I=1 TO N
1070 '  ---------------
1080    FOR J=1 TO 2
1090       READ T(I,J)
1100    NEXT J
1110 '  ---------------
1120 NEXT I
1130 '---------------
1140 '
1150 PRINT "Title of the table and of the plot : "
1160   INPUT T$
```

```
1170   INPUT "   Latitude of the pole you want to test ";LAT
1180   INPUT "   Longitude of the pole ";LONG
1190 ' ------------------------------------------------------------------------
1200 '
1210 '     COMPUTATION PART
1220 '     ================
1230 DIM S(N)
1240 DIM V1(N,9),V2(N,9),V3(N,9)
1250 FPI#=3.14159265359#/180                    'PI/180 degrees
1260 '
1270 '    Definition of functions to compute distances and directions
1280 '    using spheric trigonometry
1290 '    ------------------------------------------------------------
1300 DEF FNA(X,Y,Z,W)=SIN(X*FPI#)*SIN(Z*FPI#)+COS(X*FPI#)*COS(Z*FPI#)*COS((W-Y
)*FPI#)       'cos of the distance between two points : X and Y = lat and long
                of the first point, Z and W = lat and long of the second point
1310 DEF FNB(X)=(-ATN(X/SQR(-X*X+1))+1.5708)/FPI#       'arcosine in degrees
1320 DEF FNC(X,Y,Z)=(COS(X*FPI#)-COS(Y*FPI#)*COS(Z*FPI#))/(SIN(Y*FPI#)*SIN(Z*F
PI#))        'cos of the angle facing X, with other lengths equal to Y and Z
1330 '
1340 '    Distance from each point to the preceding one
1350 '    ---------------------------------------------
1360 '---------------
1370  FOR J=2 TO N
1380     K#=FNA(T(J-1,1),T(J-1,2),T(J,1),T(J,2))
1390          IF K#).9999999999999999# THEN T(J,3)=0 : GOTO 1410
1400     T(J,3)=FNB(K#)
1410  NEXT J
1420 '---------------
1430 '
1440 '    Table of the distances in degrees for each computation step
1450 '    -----------------------------------------------------------
1460 DIM D(14)
1470 D(1)=2 : D(2)=1.5 : D(3)=1
1480 D(4)=.7 : D(5)=.5 : D(6)=.35 : D(7)=.2 : D(8)=.15 : D(9)=.1
1490 D(10)=.07 : D(11)=.05 : D(12)=.03 : D(13)=.02 : D(14)=.01
1500 '
1510 '    Initialization of latitude, longitude and loop number
1520 '    -----------------------------------------------------
1530 LAT(1)=LAT : LONG(1)=LONG
1540 L=1              '1st loop (the second is to verify the first one results)
1550 '
1560 '    Main loop
1570 '    ---------
1580 '---------------
1590 FOR E=1 TO 14
1600    PRINT
1610    PRINT "LOOP ";L;"    Step ";E;"  (D = ";D(E);" deg)"
                        'to indicate what the program is doing
1620    LAT(2)=CINT((LAT(1)+D(E))*100)/100 : LONG(2)=LONG(1)  'setting the lat
1630    LAT(3)=LAT(2) : LONG(3)=CINT((LONG(1)+D(E))*100)/100  'and long of the
1640    LAT(4)=LAT(1) : LONG(4)=LONG(3)                       'nine points of
1650    LAT(5)=CINT((LAT(1)-D(E))*100)/100 : LONG(5)=LONG(3)  'the basic grid
1660    LAT(6)=LAT(5) : LONG(6)=LONG(1)                       'of computation
1670    LAT(7)=LAT(5) : LONG(7)=CINT((LONG(1)-D(E))*100)/100
1680    LAT(8)=LAT(1) : LONG(8)=LONG(7)
1690    LAT(9)=LAT(2) : LONG(9)=LONG(7)
1700 '
1710 '     Computation of the parameters of the first basic grid of each step
1720 '     ------------------------------------------------------------------
1730 '---------------
1740  FOR I=1 TO 9
1750     GOSUB 1800
1760  NEXT I
1770 '---------------
1780  GOTO 2340
1790 '
1800 '     Computing the distance between each point and the pole
1810 '     -----------------------------------------------------
1820 '---------------
1830  FOR J=1 TO N
1840     K#=FNA(LAT(I),LONG(I),T(J,1),T(J,2))
1850          IF K#).9999999999999999# THEN V1(J,I)=0 : GOTO 1870
1860     V1(J,I)=FNB(K#)
1870  NEXT J
1880 '---------------
1890 '
1900 '     Computing the mean distance from the points to the pole
1910 '     ------------------------------------------------------
1920  SUM=0
1930 '---------------
1940  FOR J=1 TO N
1950     SUM=SUM+V1(J,I)
1960  NEXT J
1970 '---------------
1980  DIST(I)=SUM/N
1990 '
2000 '     Direction of the small circle in degrees from North toward East
2010 '     at each point
2020 '     ---------------------------------------------------------------
2030 '---------------
2040  FOR J=1 TO N
2050     SIGN=SGN(T(J,1)+LAT(I))                  'sign of the computation
2060     K#=FNC(SIGN*90-LAT(I),V1(J,I),SIGN*90-T(J,1))
2070          IF K#).9999999999999999# THEN K=0 : GOTO 2090
2080     K=FNB(K#)
2090     V2(J,I)=90+K*SGN(LONG(I)-T(J,2))
2100          IF V2(J,I)<0 THEN V2(J,I)=180+V2(J,I) : GOTO 2120
2110          IF V2(J,I)>180 THEN V2(J,I)=V2(J,I)-180
2120  NEXT J
2130 '---------------
2140 '
2150 '     Difference between the real distance to the pole
2160 '     and the mean distance
2170 '     -----------------------------------------------
2180 '---------------
2190  FOR J=1 TO N
2200     V3(J,I)=V1(J,I)-DIST(I)
2210  NEXT J
2220 '---------------
2230 '
2240 '     Computing the sum of the squares of the differences
2250 '     ---------------------------------------------------
2260  SUM(I)=0
2270 '---------------
```

```
2280    FOR J=1 TO N
2290       SUM(I)=SUM(I)+V3(J,I)*V3(J,I)
2300    NEXT J
2310 '  ---------------
2320    RETURN
2330 '
2340 '      Looking for the point having the smallest deviation
2350 '      ---------------------------------------------------
2360    SUM=SUM(1) : NB=1
2370 '  ---------------
2380    FOR I=2 TO 9
2390       IF SUM(I)<SUM THEN SUM=SUM(I) : NB=I
2400    NEXT I
2410 '  ---------------
2420         IF NB=1 THEN 4210                          'last point of a step
2430    ON NB-1 GOTO 2450,2670,2890,3110,3330,3550,3770,3990 'the pnt with the
  smallest dev. is not the central pnt of the basic grid so computation of the
  missing values of the new grid after having moved its center on the pnt with
  min dev.
2440 '
2450 '      NB=2                  'minimum sum in position 2 (N)
2460 '      ---------------------------------------------------
2470    LAT(5)=LAT(4) : LONG(5)=LONG(4) : DIST(5)=DIST(4) : SUM(5)=SUM(4)
2480    FOR J=1 TO N : V1(J,5)=V1(J,4):V2(J,5)=V2(J,4):V3(J,5)=V3(J,4): NEXT J
2490    LAT(6)=LAT(1) : LONG(6)=LONG(1) : DIST(6)=DIST(1) : SUM(6)=SUM(1)
2500    FOR J=1 TO N : V1(J,6)=V1(J,1):V2(J,6)=V2(J,1):V3(J,6)=V3(J,1): NEXT J
2510    LAT(7)=LAT(8) : LONG(7)=LONG(8) : DIST(7)=DIST(8) : SUM(7)=SUM(8)
2520    FOR J=1 TO N : V1(J,7)=V1(J,8):V2(J,7)=V2(J,8):V3(J,7)=V3(J,8): NEXT J
2530    LAT(8)=LAT(9) : LONG(8)=LONG(9) : DIST(8)=DIST(9) : SUM(8)=SUM(9)
2540    FOR J=1 TO N : V1(J,8)=V1(J,9):V2(J,8)=V2(J,9):V3(J,8)=V3(J,9): NEXT J
2550    LAT(1)=LAT(2) : LONG(1)=LONG(2) : DIST(1)=DIST(2) : SUM(1)=SUM(2)
2560    FOR J=1 TO N : V1(J,1)=V1(J,2):V2(J,1)=V2(J,2):V3(J,1)=V3(J,2): NEXT J
2570    LAT(4)=LAT(3) : LONG(4)=LONG(3) : DIST(4)=DIST(3) : SUM(4)=SUM(3)
2580    FOR J=1 TO N : V1(J,4)=V1(J,3):V2(J,4)=V2(J,3):V3(J,4)=V3(J,3): NEXT J
2590    LAT(2)=CINT((LAT(1)+D(E))*100)/100 : LONG(2)=LONG(1)
2600    LAT(3)=LAT(2) : LONG(3)=LONG(4)
2610    LAT(9)=LAT(2) : LONG(9)=LONG(7)
2620    I=2 : GOSUB 1800
2630    I=3 : GOSUB 1800
2640    I=9 : GOSUB 1800
2650    GOTO 2340
2660 '
2670 '      NB=3                  'minimum sum in position 3 (NE)
2680 '      ---------------------------------------------------
2690    LAT(6)=LAT(4) : LONG(6)=LONG(4) : DIST(6)=DIST(4) : SUM(6)=SUM(4)
2700    FOR J=1 TO N : V1(J,6)=V1(J,4):V2(J,6)=V2(J,4):V3(J,6)=V3(J,4): NEXT J
2710    LAT(7)=LAT(1) : LONG(7)=LONG(1) : DIST(7)=DIST(1) : SUM(7)=SUM(1)
2720    FOR J=1 TO N : V1(J,7)=V1(J,1):V2(J,7)=V2(J,1):V3(J,7)=V3(J,1): NEXT J
2730    LAT(8)=LAT(2) : LONG(8)=LONG(2) : DIST(8)=DIST(2) : SUM(8)=SUM(2)
2740    FOR J=1 TO N : V1(J,8)=V1(J,2):V2(J,8)=V2(J,2):V3(J,8)=V3(J,2): NEXT J
2750    LAT(1)=LAT(3) : LONG(1)=LONG(3) : DIST(1)=DIST(3) : SUM(1)=SUM(3)
2760    FOR J=1 TO N : V1(J,1)=V1(J,3):V2(J,1)=V2(J,3):V3(J,1)=V3(J,3): NEXT J
2770    LAT(2)=CINT((LAT(1)+D(E))*100)/100 : LONG(2)=LONG(1)
2780    LAT(3)=LAT(2) : LONG(3)=CINT((LONG(1)+D(E))*100)/100
2790    LAT(4)=LAT(1) : LONG(4)=LONG(3)
2800    LAT(5)=LAT(6) : LONG(5)=LONG(3)
2810    LAT(9)=LAT(2) : LONG(9)=LONG(8)
2820    I=2 : GOSUB 1800
2830    I=3 : GOSUB 1800
2840    I=4 : GOSUB 1800
2850    I=5 : GOSUB 1800
2860    I=9 : GOSUB 1800
2870    GOTO 2340
2880 '
2890 '      NB=4                  'minimum sum in position 4 (E)
2900 '      ---------------------------------------------------
2910    LAT(9)=LAT(2) : LONG(9)=LONG(2) : DIST(9)=DIST(2) : SUM(9)=SUM(2)
2920    FOR J=1 TO N : V1(J,9)=V1(J,2):V2(J,9)=V2(J,2):V3(J,9)=V3(J,2): NEXT J
2930    LAT(8)=LAT(1) : LONG(8)=LONG(1) : DIST(8)=DIST(1) : SUM(8)=SUM(1)
2940    FOR J=1 TO N : V1(J,8)=V1(J,1):V2(J,8)=V2(J,1):V3(J,8)=V3(J,1): NEXT J
2950    LAT(7)=LAT(6) : LONG(7)=LONG(6) : DIST(7)=DIST(6) : SUM(7)=SUM(6)
2960    FOR J=1 TO N : V1(J,7)=V1(J,6):V2(J,7)=V2(J,6):V3(J,7)=V3(J,6): NEXT J
2970    LAT(6)=LAT(5) : LONG(6)=LONG(5) : DIST(6)=DIST(5) : SUM(6)=SUM(5)
2980    FOR J=1 TO N : V1(J,6)=V1(J,5):V2(J,6)=V2(J,5):V3(J,6)=V3(J,5): NEXT J
2990    LAT(1)=LAT(4) : LONG(1)=LONG(4) : DIST(1)=DIST(4) : SUM(1)=SUM(4)
3000    FOR J=1 TO N : V1(J,1)=V1(J,4):V2(J,1)=V2(J,4):V3(J,1)=V3(J,4): NEXT J
3010    LAT(2)=LAT(3) : LONG(2)=LONG(3) : DIST(2)=DIST(3) : SUM(2)=SUM(3)
3020    FOR J=1 TO N : V1(J,2)=V1(J,3):V2(J,2)=V2(J,3):V3(J,2)=V3(J,3): NEXT J
3030    LAT(3)=LAT(2) : LONG(3)=CINT((LONG(1)+D(E))*100)/100
3040    LAT(4)=LAT(1) : LONG(4)=LONG(3)
3050    LAT(5)=LAT(6) : LONG(5)=LONG(3)
3060    I=3 : GOSUB 1800
3070    I=4 : GOSUB 1800
3080    I=5 : GOSUB 1800
3090    GOTO 2340
3100 '
3110 '      NB=5                  'minimum sum in position 5 (SE)
3120 '      ---------------------------------------------------
3130    LAT(2)=LAT(4) : LONG(2)=LONG(4) : DIST(2)=DIST(4) : SUM(2)=SUM(4)
3140    FOR J=1 TO N : V1(J,2)=V1(J,4):V2(J,2)=V2(J,4):V3(J,2)=V3(J,4): NEXT J
3150    LAT(9)=LAT(1) : LONG(9)=LONG(1) : DIST(9)=DIST(1) : SUM(9)=SUM(1)
3160    FOR J=1 TO N : V1(J,9)=V1(J,1):V2(J,9)=V2(J,1):V3(J,9)=V3(J,1): NEXT J
3170    LAT(8)=LAT(6) : LONG(8)=LONG(6) : DIST(8)=DIST(6) : SUM(8)=SUM(6)
3180    FOR J=1 TO N : V1(J,8)=V1(J,6):V2(J,8)=V2(J,6):V3(J,8)=V3(J,6): NEXT J
3190    LAT(1)=LAT(5) : LONG(1)=LONG(5) : DIST(1)=DIST(5) : SUM(1)=SUM(5)
3200    FOR J=1 TO N : V1(J,1)=V1(J,5):V2(J,1)=V2(J,5):V3(J,1)=V3(J,5): NEXT J
3210    LAT(3)=LAT(2) : LONG(3)=CINT((LONG(1)+D(E))*100)/100
3220    LAT(4)=LAT(1) : LONG(4)=LONG(3)
3230    LAT(5)=CINT((LAT(1)-D(E))*100)/100 : LONG(5)=LONG(3)
3240    LAT(6)=LAT(5) : LONG(6)=LONG(1)
3250    LAT(7)=LAT(5) : LONG(7)=LONG(8)
3260    I=3 : GOSUB 1800
3270    I=4 : GOSUB 1800
3280    I=5 : GOSUB 1800
3290    I=6 : GOSUB 1800
3300    I=7 : GOSUB 1800
3310    GOTO 2340
3320 '
3330 '      NB=6                  'minimum sum in position 6 (S)
3340 '      ---------------------------------------------------
3350    LAT(9)=LAT(8) : LONG(9)=LONG(8) : DIST(9)=DIST(8) : SUM(9)=SUM(8)
3360    FOR J=1 TO N : V1(J,9)=V1(J,8):V2(J,9)=V2(J,8):V3(J,9)=V3(J,8): NEXT J
3370    LAT(2)=LAT(1) : LONG(2)=LONG(1) : DIST(2)=DIST(1) : SUM(2)=SUM(1)
3380    FOR J=1 TO N : V1(J,2)=V1(J,1):V2(J,2)=V2(J,1):V3(J,2)=V3(J,1): NEXT J
3390    LAT(3)=LAT(4) : LONG(3)=LONG(4) : DIST(3)=DIST(4) : SUM(3)=SUM(4)
3400    FOR J=1 TO N : V1(J,3)=V1(J,4):V2(J,3)=V2(J,4):V3(J,3)=V3(J,4): NEXT J
```

```
3410    LAT(4)=LAT(5) : LONG(4)=LONG(5) : DIST(4)=DIST(5) : SUM(4)=SUM(5)
3420    FOR J=1 TO N : V1(J,4)=V1(J,5):V2(J,4)=V2(J,5):V3(J,4)=V3(J,5): NEXT J
3430    LAT(1)=LAT(6) : LONG(1)=LONG(6) : DIST(1)=DIST(6) : SUM(1)=SUM(6)
3440    FOR J=1 TO N : V1(J,1)=V1(J,6):V2(J,1)=V2(J,6):V3(J,1)=V3(J,6): NEXT J
3450    LAT(8)=LAT(7) : LONG(8)=LONG(7) : DIST(8)=DIST(7) : SUM(8)=SUM(7)
3460    FOR J=1 TO N : V1(J,8)=V1(J,7):V2(J,8)=V2(J,7):V3(J,8)=V3(J,7): NEXT J
3470    LAT(5)=CINT((LAT(1)-D(E))*100)/100 : LONG(5)=LONG(3)
3480    LAT(6)=LAT(5) : LONG(6)=LONG(1)
3490    LAT(7)=LAT(5) : LONG(7)=LONG(8)
3500    I=5 : GOSUB 1800
3510    I=6 : GOSUB 1800
3520    I=7 : GOSUB 1800
3530    GOTO 2340
3540 '
3550 '       NB=7                  'minimum sum in position 7 (SW)
3560 '       ------------------------------------------------------
3570    LAT(2)=LAT(8) : LONG(2)=LONG(8) : DIST(2)=DIST(8) : SUM(2)=SUM(8)
3580    FOR J=1 TO N : V1(J,2)=V1(J,8):V2(J,2)=V2(J,8):V3(J,2)=V3(J,8): NEXT J
3590    LAT(3)=LAT(1) : LONG(3)=LONG(1) : DIST(3)=DIST(1) : SUM(3)=SUM(1)
3600    FOR J=1 TO N : V1(J,3)=V1(J,1):V2(J,3)=V2(J,1):V3(J,3)=V3(J,1): NEXT J
3610    LAT(4)=LAT(6) : LONG(4)=LONG(6) : DIST(4)=DIST(6) : SUM(4)=SUM(6)
3620    FOR J=1 TO N : V1(J,4)=V1(J,6):V2(J,4)=V2(J,6):V3(J,4)=V3(J,6): NEXT J
3630    LAT(1)=LAT(7) : LONG(1)=LONG(7) : DIST(1)=DIST(7) : SUM(1)=SUM(7)
3640    FOR J=1 TO N : V1(J,1)=V1(J,7):V2(J,1)=V2(J,7):V3(J,1)=V3(J,7): NEXT J
3650    LAT(5)=CINT((LAT(1)-D(E))*100)/100 : LONG(5)=LONG(3)
3660    LAT(6)=LAT(5) : LONG(6)=LONG(1)
3670    LAT(7)=LAT(5) : LONG(7)=CINT((LONG(1)-D(E))*100)/100
3680    LAT(8)=LAT(1) : LONG(8)=LONG(7)
3690    LAT(9)=LAT(2) : LONG(9)=LONG(7)
3700    I=5 : GOSUB 1800
3710    I=6 : GOSUB 1800
3720    I=7 : GOSUB 1800
3730    I=8 : GOSUB 1800
3740    I=9 : GOSUB 1800
3750    GOTO 2340
3760 '
3770 '       NB=8                  'minimum sum in position 8 (W)
3780 '       ------------------------------------------------------
3790    LAT(3)=LAT(2) : LONG(3)=LONG(2) : DIST(3)=DIST(2) : SUM(3)=SUM(2)
3800    FOR J=1 TO N : V1(J,3)=V1(J,2):V2(J,3)=V2(J,2):V3(J,3)=V3(J,2): NEXT J
3810    LAT(4)=LAT(1) : LONG(4)=LONG(1) : DIST(4)=DIST(1): SUM(4)=SUM(1)
3820    FOR J=1 TO N : V1(J,4)=V1(J,1):V2(J,4)=V2(J,1):V3(J,4)=V3(J,1): NEXT J
3830    LAT(5)=LAT(6) : LONG(5)=LONG(6) : DIST(5)=DIST(6) : SUM(5)=SUM(6)
3840    FOR J=1 TO N : V1(J,5)=V1(J,6):V2(J,5)=V2(J,6):V3(J,5)=V3(J,6): NEXT J
3850    LAT(6)=LAT(7) : LONG(6)=LONG(7) : DIST(6)=DIST(7) : SUM(6)=SUM(7)
3860    FOR J=1 TO N : V1(J,6)=V1(J,7):V2(J,6)=V2(J,7):V3(J,6)=V3(J,7): NEXT J
3870    LAT(1)=LAT(8) : LONG(1)=LONG(8) : DIST(1)=DIST(8) : SUM(1)=SUM(8)
3880    FOR J=1 TO N : V1(J,1)=V1(J,8):V2(J,1)=V2(J,8):V3(J,1)=V3(J,8): NEXT J
3890    LAT(2)=LAT(9) : LONG(2)=LONG(9) : DIST(2)=DIST(9) : SUM(2)=SUM(9)
3900    FOR J=1 TO N : V1(J,2)=V1(J,9):V2(J,2)=V2(J,9):V3(J,2)=V3(J,9): NEXT J
3910    LAT(7)=LAT(5) : LONG(7)=CINT((LONG(1)-D(E))*100)/100
3920    LAT(8)=LAT(1) : LONG(8)=LONG(7)
3930    LAT(9)=LAT(2) : LONG(9)=LONG(7)
3940    I=7 : GOSUB 1800
3950    I=8 : GOSUB 1800
3960    I=9 : GOSUB 1800
3970    GOTO 2340
3980 '
3990 '       NB=9                  'minimum sum in position 9 (NW)
4000 '       ------------------------------------------------------
4010    LAT(4)=LAT(2) : LONG(4)=LONG(2) : DIST(4)=DIST(2) : SUM(4)=SUM(2)
4020    FOR J=1 TO N : V1(J,4)=V1(J,2):V2(J,4)=V2(J,2):V3(J,4)=V3(J,2): NEXT J
4030    LAT(5)=LAT(1) : LONG(5)=LONG(1) : DIST(5)=DIST(1) : SUM(5)=SUM(1)
4040    FOR J=1 TO N : V1(J,5)=V1(J,1):V2(J,5)=V2(J,1):V3(J,5)=V3(J,1): NEXT J
4050    LAT(6)=LAT(8) : LONG(6)=LONG(8) : DIST(6)=DIST(8) : SUM(6)=SUM(8)
4060    FOR J=1 TO N : V1(J,6)=V1(J,8):V2(J,6)=V2(J,8):V3(J,6)=V3(J,8): NEXT J
4070    LAT(1)=LAT(9) : LONG(1)=LONG(9) : DIST(1)=DIST(9) : SUM(1)=SUM(9)
4080    FOR J=1 TO N : V1(J,1)=V1(J,9):V2(J,1)=V2(J,9):V3(J,1)=V3(J,9): NEXT J
4090    LAT(2)=CINT((LAT(1)+D(E))*100)/100 : LONG(2)=LONG(1)
4100    LAT(3)=LAT(2) : LONG(3)=LONG(4)
4110    LAT(7)=LAT(5) : LONG(7)=CINT((LONG(1)-D(E))*100)/100
4120    LAT(8)=LAT(1) : LONG(8)=LONG(7)
4130    LAT(9)=LAT(2) : LONG(9)=LONG(7)
4140    I=2 : GOSUB 1800
4150    I=3 : GOSUB 1800
4160    I=7 : GOSUB 1800
4170    I=8 : GOSUB 1800
4180    I=9 : GOSUB 1800
4190    GOTO 2340
4200 '
4210 '       Standard deviation in degrees
4220 '       -----------------------------
4230    SDEV=SQR(SUM(1)/(N-1))
4240 '
4250 '       Printing the result of the subloop on the screen
4260 '       ------------------------------------------------
4270    PRINT USING "Lat = ###.##";LAT(1);
4280    PRINT USING "    Long = ####.##";LONG(1)
4290    PRINT USING " Mean distance = ####.##";DIST(1)*10000/90; : PRINT " km"
4300    PRINT USING " Standard deviation = ##.##";SDEV*10000/90; : PRINT " km"
4310 NEXT E
4320 '---------------
4330 ' ---------------------------------------------------------------------------
4340 '
4350 '     PRINTING PART
4360 '     =============
4370      IF L>1 THEN 4590                        'to print the title only once
4380 BEEP : PRINT
4390   INPUT "PRINTER READY (Y/N) ";R$ : R$=LEFT$(R$,1)
4400       IF R$<>"Y" AND R$<>"y" THEN 4380
4410 '
4420 '     Preparing the printer
4430 '     ---------------------
4440 LPRINT CHR$(27)"n" : LPRINT CHR$(27)"L017" : LPRINT CHR$(27)"!"
4450 '
4460 '     Printing the title and subtitle
4470 '     -------------------------------
4480 T=INT(72-17-10-LEN(T$))/2
4490 LPRINT TAB(T) CHR$(27)"X" T$                                  'title
4500 LPRINT CHR$(27)"E" : LPRINT CHR$(27)"Y"
4510 LPRINT : LPRINT
4520 LPRINT "Looking for the best fitting pole : criterion is distance"
4530 LPRINT CHR$(27)CHR$(34) : LPRINT CHR$(27)"N"
4540 LPRINT
4550 LPRINT USING "Position tested :  Lat = ###.##";LAT;
4560 LPRINT USING "  Long = ####.##";LONG
```

```
4570 LPRINT
4580 '
4590 '    Printing of the results of each loop on the line printer
4600 '    ---------------------------------------------------------
4610       IF L=1 THEN 4650
4620       IF CINT(LAT(1)*100)=CINT(LAT*100) AND CINT(LONG(1)*100)=CINT(LONG*1
00) THEN 4930               'are the results of the two last loops the same ?
4630 LPRINT
4640 LPRINT "Loop  ";L
4650 LPRINT USING "Center position :  Lat = ###.##";LAT(1);
4660 LPRINT USING "  Long = ####.##";LONG(1)
4670 LPRINT USING "Mean distance to the pole : ####.##";DIST(1)*10000/90;
4680 LPRINT " km"
4690 LPRINT : LPRINT
4700 LPRINT "  LAT      LONG     DPOLE    DEACH     CDIR     DDIST"
4710 LPRINT "  deg       deg       km       km       deg       km"
4720 LPRINT
4730 '---------------
4740 FOR J=1 TO N
4750    LPRINT USING "###.##  ";T(J,1);                  'latitude
4760    LPRINT USING "####.##  ";T(J,2);                 'longitude
4770    LPRINT USING "####.#  ";V1(J,1)*10000/90;        'distance to the pole
4780    LPRINT USING "####.#  ";T(J,3)*10000/90;         'dist. between points
4790    LPRINT USING "####.##  ";V2(J,1);                'computed direction
4800    LPRINT USING "###.##";V3(J,1)*10000/90           'distance deviation
4810 NEXT J
4820 '---------------
4830 LPRINT : LPRINT
4840 LPRINT USING "Standard deviation : ##.##";SDEV*10000/90; : LPRINT " Km"
4850 LPRINT
4860 '
4870 '    Preparing the next loop
4880 '    -----------------------
4890 L=L+1
4900 LAT=LAT(1) : LONG=LONG(1) : DIST=DIST(1)
4910 GOTO 1560
4920 LPRINT
4930 LPRINT "Results verified."      'the two last loops show the same results
4940 ' ------------------------------------------------------------------------
4950 '
4960 '     DRAWING PART
4970 '     ============
4980 '
4990 '    Sum of the distance between points, just for the plot
5000 '    ----------------------------------------------------
5010 S=0 : S(1)=0
5020 '---------------
5030 FOR I=2 TO N
5040    S=S+T(I,3) : S(I)=S
5050 NEXT I
5060 '---------------
5070 '
5080 '    Looking for the minimum and maximum distance to the pole
5090 '    --------------------------------------------------------
5100 MIN=DIST : MAX=MIN
5110 '---------------
5120 FOR I=1 TO N
5130       IF V1(I,1)<MIN THEN MIN=V1(I,1)
5140       IF V1(I,1)>MAX THEN MAX=V1(I,1)
5150 NEXT I
5160 '---------------
5170      IF DIST-2*SDEV<MIN THEN MIN=DIST-2*SDEV
5180      IF DIST+2*SDEV>MAX THEN MAX=DIST+2*SDEV
5190 '
5200 '    Asking questions for the graph
5210 '    ------------------------------
5220 PRINT : PRINT
5230 PRINT "Minimum = ";MIN*10000/90;"       Maximum = ";MAX*10000/90
5240   INPUT "Minimum value on the graph ";MIN : MIN=MIN*90/10000
5250   INPUT "Maximum value on the graph ";MAX : MAX=MAX*90/10000
5260   INPUT "   step for label printing ";ST : ST=ST*90/10000
5270 PRINT
5280 '
5290 '    Instructions to modify and copy the graph on the line printer
5300 '    -------------------------------------------------------------
5310 PRINT " Now the program will display a graph of the distance from the"
5320 PRINT "pole to the points and the mean distance."
5330 PRINT " By pressing the <G> key while the graph is being displayed you"
5340 PRINT "be able to change the parameters of the graph."
5350 PRINT " By pressing the <C> key a copy of the screen will be made on the"
5360 PRINT "printer, it will take about half an hour to complete, but before"
5370 PRINT "do not forget to change the paper."
5380 PRINT " The program will end by pressing any other key."
5390 PRINT : PRINT
5400   INPUT "READY (Y/N) ";R$ : R$=LEFT$(R$,1)
5410      IF R$<>"Y" AND R$<>"y" THEN 5400
5420 SCREEN 1 : CLS                  'setting the screen for graphics
5430 '
5440 '    Drawing the axis
5450 '    ----------------
5460 LINE (69,21)-(69,172) 'distance-between-points axis drawn twice to have
5470 LINE (70,21)-(70,172) 'a line as heavy as the distance-to-the-pole axis
5480 LINE -(560,172)       'draw the distance-to-the-pole axis
5490 '
5500 '    Drawing of the line linking the real data
5510 '    -----------------------------------------
5520 PSET ((V1(1,1)-MIN)*490/(MAX-MIN)+70,S(1)*151/S+21)
5530 '---------------
5540 FOR I=2 TO N
5550    LINE -((V1(I,1)-MIN)*490/(MAX-MIN)+70,S(I)*151/S+21)
5560 NEXT I
5570 '---------------
5580 '
5590 '    Drawing the line of the mean distance to the pole three times
5600 '    -------------------------------------------------------------
5610 LINE ((DIST-MIN)*490/(MAX-MIN)+69,21)-((DIST-MIN)*490/(MAX-MIN)+69,172)
5620 LINE ((DIST-MIN)*490/(MAX-MIN)+70,21)-((DIST-MIN)*490/(MAX-MIN)+70,172)
5630 LINE ((DIST-MIN)*490/(MAX-MIN)+71,21)-((DIST-MIN)*490/(MAX-MIN)+71,172)
5640 '
5650 '    Drawing the 70%-error-interval lines and printing the percentage
5660 '    ----------------------------------------------------------------
5670 LINE ((DIST-SDEV-MIN)*490/(MAX-MIN)+69,21)-((DIST-SDEV-MIN)*490/(MAX-MIN)
+69,172)
5680 LOCATE 2,INT(((DIST-SDEV-MIN)*490/(MAX-MIN)+69)/8) : PRINT "70%"
```

```
5690 LINE ((DIST+SDEV-MIN)*490/(MAX-MIN)+69,21)-((DIST+SDEV-MIN)*490/(MAX-MIN)
+69,172)
5700 LOCATE 2,INT(((DIST+SDEV-MIN)*490/(MAX-MIN)+69)/8) : PRINT "70%"
5710 '
5720 '    Drawing the 95%-error-interval lines and printing the percentage
5730 '    ------------------------------------------------------------
5740 LINE ((DIST-2*SDEV-MIN)*490/(MAX-MIN)+69,21)-((DIST-2*SDEV-MIN)*490/(MAX-
MIN)+69,172)
5750 LOCATE 2,INT(((DIST-2*SDEV-MIN)*490/(MAX-MIN)+69)/8) : PRINT "95%"
5760 LINE ((DIST+2*SDEV-MIN)*490/(MAX-MIN)+69,21)-((DIST+2*SDEV-MIN)*490/(MAX-
MIN)+69,172)
5770 LOCATE 2,INT(((DIST+2*SDEV-MIN)*490/(MAX-MIN)+69)/8) : PRINT "95%"
5780 '
5790 '    Drawing the tick marks along the-distance-between-points axis
5800 '    and printing the point numbers
5810 '    ------------------------------------------------------------
5820 '---------------
5830 FOR I=1 TO N
5840    LINE (60,S(I)*151/S+21)-(70,S(I)*151/S+21)
5850    LOCATE INT((S(I)*151/S+21)/9)+1,5 : PRINT USING "##";I
5860 NEXT I
5870 '---------------
5880 '
5890 '    Printing the distance-to-the-points axis title
5900 '    ----------------------------------------------
5910 LOCATE 1,6 : PRINT "POINT"
5920 LOCATE 2,6 : PRINT "NUMBER"
5930 '
5940 '    Printing the pole position and mean distance
5950 '    -------------------------------------------
5960 LOCATE 1,15
5970 PRINT USING "POLE POSITION :  LAT = ###.##";LAT(1);
5980 PRINT USING ", LONG = ####.##";LONG(1);
5990 PRINT USING ", DIST = ####.#";DIST*10000/90; : PRINT " KM"
6000 '
6010 '    Drawing the tick marks along the distance-to-the-pole axis twice
6020 '    and printing the labels
6030 '    ------------------------------------------------------------------
6040 '---------------
6050 FOR I=MIN TO MAX STEP ST
6060    LINE (69+(I-MIN)*490/(MAX-MIN),172)-(69+(I-MIN)*490/(MAX-MIN),178)
6070    LINE (70+(I-MIN)*490/(MAX-MIN),172)-(70+(I-MIN)*490/(MAX-MIN),178)
6080    LOCATE 21,(70+(I-MIN)*490/(MAX-MIN))/8-2
6090    PRINT USING "####";I*10000/90
6100 NEXT I
6110 '---------------
6120 '
6130 '    Printing the distance-to-the-pole axis title and unit
6140 '    ----------------------------------------------------
6150 LOCATE 19,72 : PRINT "DISTANCE"
6160 LOCATE 21,75 : PRINT "KM"
6170 '
6180 '    Printing the title and underlining it
6190 '    -------------------------------------
6200 LOCATE 23,(40-LEN(T$)/2) : PRINT T$
6210 LINE ((39-INT(LEN(T$)/2))*8,210)-STEP(LEN(T$)*8,0)
6220 '
6230 '    Waiting for and analysing the keyboard entry
6240 '    --------------------------------------------
6250 LOCATE 1,1
6260 Q$=INKEY$ : IF Q$="" THEN 6260          'waiting to change the graph
6270       IF Q$="G" OR Q$="g" THEN SCREEN 0 : LOCATE 24,1 : GOTO 5200
6280       IF Q$<>"C" AND Q$<>"c" THEN 6520
6290 '-------------------------------------------------------------------------
6300 '
6310 '      SCREEN COPY
6320 '      ===========
6330 '    Preparing the printer for graphics
6340 '    ----------------------------------
6350 LPRINT CHR$(27)"p" : LPRINT CHR$(27)"T02"
6360 '
6370 '    Reading the screen values point after point, line after line
6380 '    and printing the lines one after the other
6390 '    ------------------------------------------------------------
6400 '---------------
6410 FOR I=0 TO 224
6420    X1$="" : X2$="" : X3$="" : X4$=""
6430    FOR J=0 TO 159 : X1$=X1$+CHR$(POINT(J,I)/7) : NEXT J
6440    FOR J=160 TO 319 : X2$=X2$+CHR$(POINT(J,I)/7) : NEXT J
6450    FOR J=320 TO 479 : X3$=X3$+CHR$(POINT(J,I)/7) : NEXT J
6460    FOR J=480 TO 639 : X4$=X4$+CHR$(POINT(J,I)/7) : NEXT J
6470    LPRINT CHR$(27)"G0640"X1$X2$X3$X4$
6480 NEXT I
6490 '---------------
6500 LPRINT CHR$(27)"A" : LPRINT CHR$(27)"E"
                                  'setting the printer back to printing mode
6510 GOTO 6250                    'to wait for the next keyboard entry
6520 SCREEN 0                     'setting the screen back to printing mode
6530 END
```

C
★

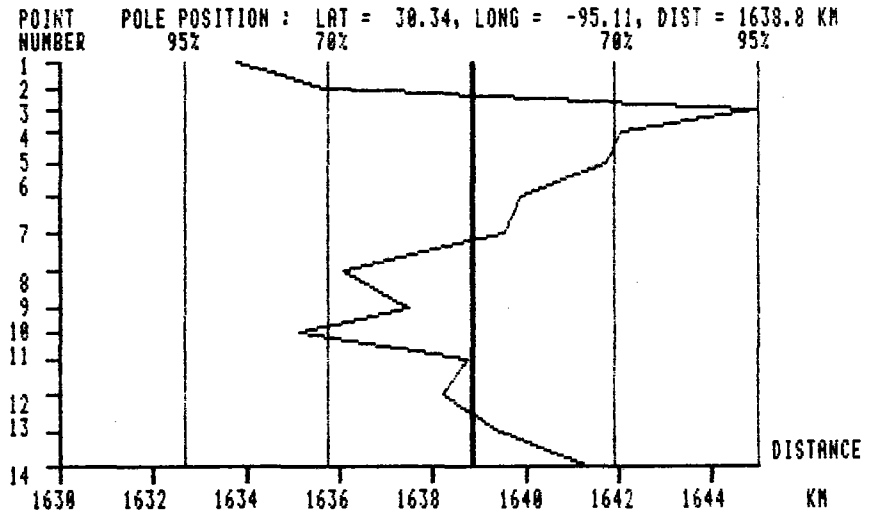
```
 This program has been written to help you to find the pole
fitting best to points distributed on a small circle.

 The program will read the data in a data file beginning with line
500, and having a variable named N giving the number of data
points. Then each data line must have first the latitude of the
point, then the longitude. The last line number must not exceed
999.

 The program will ask you the latitude and longitude of the pole
you have chosen to test. Then it will compute the sum of the
squares of the differences between the real distance and the mean
distance from the pole to each point. This value must be as close
to zero as possible, so the program will loop and test points from
2 degrees to 0.01 degree to the point having the smallest
deviation. The results will be verified and a graph will be
on the screen, and you will be allowed to copy this graph on the
printer.

Disk ready ? Y
Name of the data files on the disk :
MATRENCH.DAT DRSCMAT5.DAT DRSCMAT4.DAT DRSCMAT2.DAT DRSCMAT1.DAT DRSCMAT3.DAT
DRSCEPR1.DAT DRSCRIVE.DAT DSSCMAT2.DAT DSSCMAT1.DAT DSSCMAT3.DAT DSSCMAT5.DAT
DSSCMAT4.DAT DSSCRIVE.DAT BRKPROFL.DAT PNTPROFL.DAT DRSCRITA.DAT

Name of your data file : ? DSSCMAT1.DAT
Title of the table and of the plot :
? MIDDLE AMERICA TRENCH : NORTHERN SET
   Latitude of the pole you want to test ? 30
   Longitude of the pole ? -95

LOOP  1     Step  1   (D =  2  deg)
Lat =  30.00    Long =  -95.00
 Mean distance = 1611.46 km
 Standard deviation =  4.71 km

LOOP  1     Step  2   (D =  1.5  deg)
Lat =  30.00    Long =  -95.00
 Mean distance = 1611.46 km
 Standard deviation =  4.71 km

LOOP  1     Step  3   (D =  1  deg)
Lat =  29.00    Long =  -96.00
 Mean distance = 1468.41 km
 Standard deviation =  3.41 km

LOOP  1     Step  4   (D =  .7  deg)
Lat =  29.00    Long =  -96.00
 Mean distance = 1468.41 km
 Standard deviation =  3.41 km

LOOP  1     Step  5   (D =  .5  deg)
Lat =  29.00    Long =  -96.00
 Mean distance = 1468.41 km
 Standard deviation =  3.41 km

LOOP  1     Step  6   (D =  .35  deg)
Lat =  29.35    Long =  -95.65
 Mean distance = 1518.42 km
 Standard deviation =  3.37 km

LOOP  1     Step  7   (D =  .2  deg)
Lat =  29.75    Long =  -95.45
 Mean distance = 1566.24 km
 Standard deviation =  3.16 km

LOOP  1     Step  8   (D =  .15  deg)
Lat =  29.75    Long =  -95.45
 Mean distance = 1566.24 km
 Standard deviation =  3.16 km

LOOP  1     Step  9   (D =  .1  deg)
Lat =  29.95    Long =  -95.35
 Mean distance = 1590.15 km
 Standard deviation =  3.11 km

LOOP  1     Step  10   (D =  .07  deg)
Lat =  29.95    Long =  -95.35
 Mean distance = 1590.15 km
 Standard deviation =  3.11 km

LOOP  1     Step  11   (D =  .05  deg)
Lat =  29.95    Long =  -95.35
 Mean distance = 1590.15 km
 Standard deviation =  3.11 km

LOOP  1     Step  12   (D =  .03  deg)
Lat =  30.10    Long =  -95.26
 Mean distance = 1608.77 km
 Standard deviation =  3.09 km

LOOP  1     Step  13   (D =  .02  deg)
Lat =  30.32    Long =  -95.12
 Mean distance = 1636.45 km
 Standard deviation =  3.08 km

LOOP  1     Step  14   (D =  .01  deg)
Lat =  30.34    Long =  -95.11
 Mean distance = 1638.84 km
 Standard deviation =  3.08 km
```

a

```
PRINTER READY (Y/N) ? Y

LOOP  2     Step  1   (D =  2  deg)
Lat =  30.34    Long =  -95.11
 Mean distance = 1638.84 km
 Standard deviation =  3.08 km

LOOP  2     Step  2   (D =  1.5  deg)
Lat =  30.34    Long =  -95.11
 Mean distance = 1638.84 km
 Standard deviation =  3.08 km

LOOP  2     Step  3   (D =  1  deg)
Lat =  30.34    Long =  -95.11
 Mean distance = 1638.84 km
 Standard deviation =  3.08 km

LOOP  2     Step  4   (D =  .7  deg)
Lat =  30.34    Long =  -95.11
 Mean distance = 1638.84 km
 Standard deviation =  3.08 km

LOOP  2     Step  5   (D =  .5  deg)
Lat =  30.34    Long =  -95.11
 Mean distance = 1638.84 km
 Standard deviation =  3.08 km

LOOP  2     Step  6   (D =  .35  deg)
Lat =  30.34    Long =  -95.11
 Mean distance = 1638.84 km
 Standard deviation =  3.08 km

LOOP  2     Step  7   (D =  .2  deg)
Lat =  30.34    Long =  -95.11
 Mean distance = 1638.84 km
 Standard deviation =  3.08 km

LOOP  2     Step  8   (D =  .15  deg)
Lat =  30.34    Long =  -95.11
 Mean distance = 1638.84 km
 Standard deviation =  3.08 km

LOOP  2     Step  9   (D =  .1  deg)
Lat =  30.34    Long =  -95.11
 Mean distance = 1638.84 km
 Standard deviation =  3.08 km

LOOP  2     Step  10   (D =  .07  deg)
Lat =  30.34    Long =  -95.11
 Mean distance = 1638.84 km
 Standard deviation =  3.08 km

LOOP  2     Step  11   (D =  .05  deg)
Lat =  30.34    Long =  -95.11
 Mean distance = 1638.84 km
 Standard deviation =  3.08 km

LOOP  2     Step  12   (D =  .03  deg)
Lat =  30.34    Long =  -95.11
 Mean distance = 1638.84 km
 Standard deviation =  3.08 km

LOOP  2     Step  13   (D =  .02  deg)
Lat =  30.34    Long =  -95.11
 Mean distance = 1638.84 km
 Standard deviation =  3.08 km

LOOP  2     Step  14   (D =  .01  deg)
Lat =  30.34    Long =  -95.11
 Mean distance = 1638.84 km
 Standard deviation =  3.08 km
```

b

```
Minimum =  1632.685         Maximum =  1644.996
Minimum value on the graph ? 1630
Maximum value on the graph ? 1645
   step for label printing ? 2

 Now the program will display a graph of the distance from the
pole to the points and the mean distance.
 By pressing the (G) key while the graph is being displayed you
be able to change the parameters of the graph.
 By pressing the (C) key a copy of the screen will be made on the
printer, it will take about half an hour to complete, but before
do not forget to change the paper.
 The program will end by pressing any other key.


READY (Y/N) ? Y
```



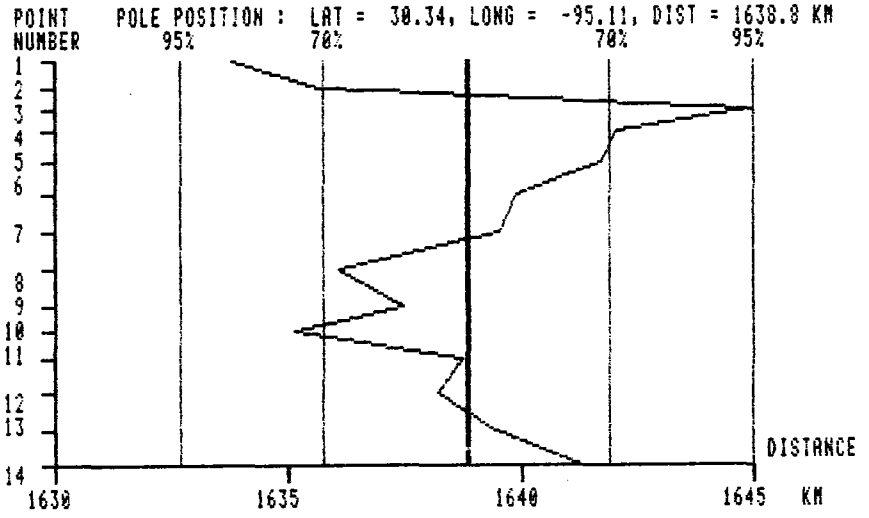
MIDDLE AMERICA TRENCH : NORTHERN SET

G

Minimum = 1630 Maximum = 1645
Minimum value on the graph ? 1630
Maximum value on the graph ? 1645
step for label printing ? 5

Now the program will display a graph of the distance from the pole to the points and the mean distance.
By pressing the (G) key while the graph is being displayed you be able to change the parameters of the graph.
By pressing the (C) key a copy of the screen will be made on the printer, it will take about half an hour to complete, but before do not forget to change the paper.
The program will end by pressing any other key.

READY (Y/N) ? Y



MIDDLE AMERICA TRENCH : NORTHERN SET

C C

D

a MIDDLE AMERICA TRENCH : NORTHERN SET

Looking for the best fitting pole : criterion is distance

Position tested : Lat = 30.00 Long = -95.00

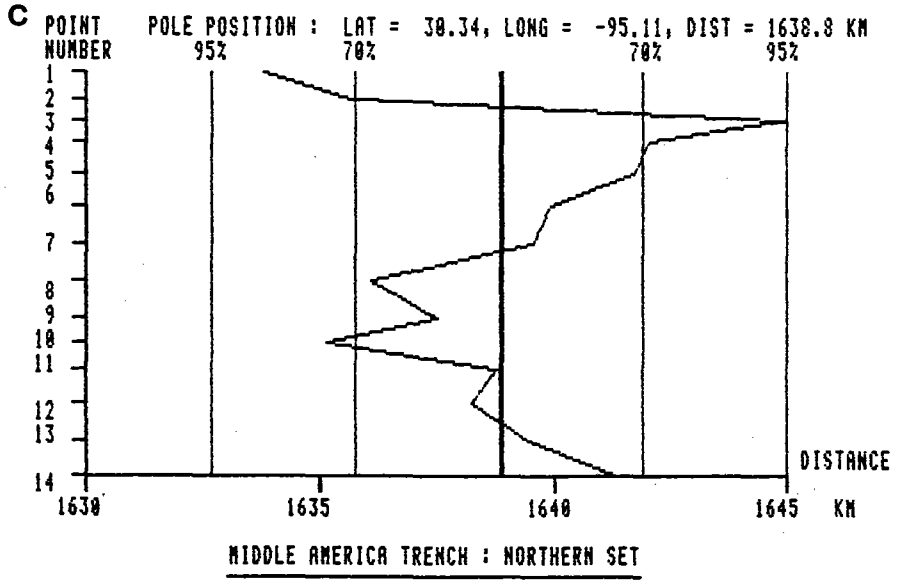
Center position : Lat = 30.34 Long = -95.11
Mean distance to the pole : 1638.84 km

LAT deg	LONG deg	DPOLE km	DEACH km	CDIR deg	DDIST km
18.67	-104.97	1633.8	0.0	125.61	-5.08
18.45	-104.67	1635.7	39.8	124.34	-3.17
18.21	-104.45	1644.9	35.2	123.24	6.09
18.07	-104.17	1642.0	33.2	122.20	3.20
17.83	-103.75	1641.7	51.7	120.56	2.85
17.62	-103.33	1639.9	50.1	118.97	1.03
17.37	-102.83	1639.5	59.7	117.08	0.67
17.17	-102.33	1636.1	57.5	115.26	-2.79
16.94	-101.83	1637.5	58.9	113.39	-1.38
16.83	-101.50	1635.1	37.2	112.22	-3.71
16.67	-101.17	1638.7	39.3	110.97	-0.11
16.50	-100.67	1638.2	56.4	109.18	-0.64
16.33	-100.17	1639.4	56.5	107.39	0.58
16.17	-99.67	1641.3	56.1	105.61	2.47

Standard deviation : 3.08 km

b Results verified.

c



MIDDLE AMERICA TRENCH : NORTHERN SET

5.2

A

```
500 ' ============================================================
510 '
520 '                          DATA `DRSCMAT1'
530 '
540 '                                        Jacqueline ROUMP, Oct. 1984
550 ' ============================================================
560 '
570 '   These data are to be used by the program DIRSMCRC.
580 '    They are the latitude and longitude of the first set of points along
590 ' the Middle America Trench : from off Manzanillo to SE of Acapulco, the
600 ' third number is the azimuth East from North of the trench axis
610 '
620 N=6                                      'number of points
630 DATA 18.45,-104.67,123
640 DATA 18.07,-104.17,120
650 DATA 17.62,-103.33,119
660 DATA 17.17,-102.33,118
670 DATA 16.83,-101.5,112
680 DATA 16.5,-100.67,108
```

B

```
10 ' ============================================================
20 '
30 '                          PROGRAM `DIRSMCRC'
40 '
50 '                                        Jacqueline ROUMP, Oct. 1984
60 ' ============================================================
70 '
80 CLS : CLEAR : OPTION BASE 1
90 '
100 '     PRESENTATION
110 '     ============
120 PRINT " This program has been written to help you to find the pole"
130 PRINT "fitting best to directions of features distributed on a small"
140 PRINT "circle."
150 PRINT
160 PRINT " The program will read the data in a data file beginning with line"
170 PRINT "500, and having a variable named N giving the number of data"
180 PRINT "points. Then each data line must have first the latitude of the"
190 PRINT "point, then the longitude, and at last the azimuth of the feature."
200 PRINT "The last line number must not exceed 999."
210 PRINT
220 PRINT " The program will ask you the latitude and longitude of the pole"
230 PRINT "you have chosen to test. Then it will compute the sum of the"
240 PRINT "squares of the differences between the real value and the azimuth"
250 PRINT "of the small circle at each point. This value must be as small as"
260 PRINT "possible so the program will  loop and test points within 2"
270 PRINT "degrees to 0.01 degree of latitude and longitude of the point"
280 PRINT "having the smallest deviation. The results will be verified and"
290 PRINT "a graph will displayed on the screen, and you will be allowed to"
300 PRINT "copy this graph on the printer."
310 PRINT
320   INPUT "Disk ready ";R$ : R$=LEFT$(R$,1)
330       IF R$()"Y" AND R$()"y" THEN 320
340 PRINT "Name of the data files on the disk :"
350 FILES "*.DAT"                            'list of the disk data files
360 PRINT
370   INPUT "Name of your data file : ";X$
380 CHAIN MERGE X$,500,ALL                   'merging the data file
390 '  ------------------------------------------------------------------------
400 '    PLACE FOR THE DATA FILE
1000 ' ------------------------------------------------------------------------
1010 '
1020 '     READING OF THE DATA FILE AND QUESTIONS
1030 '     ======================================
1040 DIM T(N,5)                              'data table
1050 '---------------
1060 FOR I=1 TO N
1070 '  ---------------
1080    FOR J=1 TO 3
1090       READ T(I,J)
1100    NEXT J
1110 '  ---------------
1120 NEXT I
1130 '---------------
1140 '
1150 PRINT "Title of the table and the plot : "
```

```
1160   INPUT T$
1170   INPUT "   Latitude of the pole you want to test ";LAT
1180   INPUT "   Longitude of the pole ";LONG
1190 ' ------------------------------------------------------------------------
1200 '
1210 '     COMPUTATION PART
1220 '     ================
1230 '
1240 DIM S(N)
1250 DIM V1(N,9),V2(N,9),V3(N,9)
1260 FPI#=3.1415926539#/180                        'PI/180 degrees
1270 '
1280 '    Definition of functions to compute distances and directions
1290 '    using spheric trigonometry
1300 '    ------------------------------------------------------------
1310 DEF FNA(X,Y,Z,W)=SIN(X*FPI#)*SIN(Z*FPI#)+COS(X*FPI#)*COS(Z*FPI#)*COS((W-Y
)*FPI#)       'cos of the distance between two points : X and Y = lat and long
                of the first point, Z and W = lat and long of the second point
1320 DEF FNB(X)=(-ATN(X/SQR(-X*X+1))+1.5708)/FPI#       'arcosine in degrees
1330 DEF FNC(X,Y,Z)=(COS(X*FPI#)-COS(Y*FPI#)*COS(Z*FPI#))/(SIN(Y*FPI#)*SIN(Z*F
PI#))          'cos of the angle facing X, with other lengths equal to Y and Z
1340 '
1350 '    Distance from each point to the preceding one
1360 '    ---------------------------------------------
1370 '---------------
1380 FOR J=2 TO N
1390     K#=FNA(T(J-1,1),T(J-1,2),T(J,1),T(J,2))
1400          IF K#).9999999999999999# THEN T(J,4)=0 : GOTO 1420
1410     T(J,4)=FNB(K#)
1420 NEXT J
1430 '---------------
1440 '
1450 '    Table of the distances in degrees for each computation step
1460 '    ----------------------------------------------------------
1470 DIM D(14)
1480 D(1)=2 : D(2)=1.5 : D(3)=1
1490 D(4)=.7 : D(5)=.5 : D(6)=.35 : D(7)=.2 : D(8)=.15 : D(9)=.1
1500 D(10)=.07 : D(11)=.05 : D(12)=.03 : D(13)=.02 : D(14)=.01
1510 '
1520 '    Initialization of latitude, longitude and loop number
1530 '    -----------------------------------------------------
1540 LAT(1)=LAT : LONG(1)=LONG
1550 L=1              '1st loop (the second is to verify the first one result)
1560 '
1570 '    Main loop
1580 '    ---------
1590 '---------------
1600 FOR E=1 TO 14
1610    PRINT
1620    PRINT "LOOP ";L;"    Step ";E;"  ( D = ";D(E);" deg)"
                        'to indicate what the program is doing
1630    LAT(2)=CINT((LAT(1)+D(E))*100)/100 : LONG(2)=LONG(1)  'setting the lat
1640    LAT(3)=LAT(2) : LONG(3)=CINT((LONG(1)+D(E))*100)/100  'and long of the
1650    LAT(4)=LAT(1) : LONG(4)=LONG(3)                       'nine points of
1660    LAT(5)=CINT((LAT(1)-D(E))*100)/100 : LONG(5)=LONG(3)  'the basic grid
1670    LAT(6)=LAT(5) : LONG(6)=LONG(1)                       'of computation
1680    LAT(7)=LAT(5) : LONG(7)=CINT((LONG(1)-D(E))*100)/100
1690    LAT(8)=LAT(1) : LONG(8)=LONG(7)
1700    LAT(9)=LAT(2) : LONG(9)=LONG(7)
1710 '
1720 '     Computation of the parameters of the first basic grid of each step
1730 '     ------------------------------------------------------------------
1740 '---------------
1750    FOR I=1 TO 9
1760       GOSUB 1810
1770    NEXT I
1780 '---------------
1790    GOTO 2340
1800 '
1810 '     Computing the distance between each point and the pole
1820 '     -----------------------------------------------------
1830 '---------------
1840    FOR J=1 TO N
1850       K#=FNA(LAT(I),LONG(I),T(J,1),T(J,2))
1860            IF K#).9999999999999999# THEN V1(J,I)=0 : GOTO 1880
1870       V1(J,I)=FNB(K#)
1880    NEXT J
1890 '---------------
1900 '
1910 '     Computing the mean distance from the points to the pole
1920 '     ------------------------------------------------------
1930    SUM=0
1940 '---------------
1950    FOR J=1 TO N
1960       SUM=SUM+V1(J,I)
1970    NEXT J
1980 '---------------
1990    DIST(I)=SUM/N
2000 '
2010 '     Direction in degrees from North toward East of the tangent
2020 '     to the small circle at each point
2030 '     ---------------------------------------------------------
2040 '---------------
2050    FOR J=1 TO N
2060       SIGN=SGN(T(J,1)+LAT(I))                  'sign of the computation
2070       K#=FNC(SIGN*90-LAT(I),V1(J,1),SIGN*90-T(J,1))
2080            IF K#).9999999999999999# THEN K=0 : GOTO 2100
2090       K=FNB(K#)
2100       V2(J,I)=90+K*SGN(LONG(I)-T(J,2))
2110            IF V2(J,I)<0 THEN V2(J,I)=180+V2(J,I) : GOTO 2130
2120            IF V2(J,I)>180 THEN V2(J,I)=V2(J,I)-180
2130    NEXT J
2140 '---------------
2150 '
2160 '     Difference between the real azimuth and the computed azimuth
2170 '     ------------------------------------------------------------
2180 '---------------
2190    FOR J=1 TO N
2200       V3(J,I)=T(J,3)-V2(J,I)
2210    NEXT J
2220 '---------------
2230 '
2240 '     Computing the sum of the squares of the differences
2250 '     ---------------------------------------------------
2260    SUM(I)=0
2270 '---------------
2280    FOR J=1 TO N
```

```
2290        SUM(I)=SUM(I)+V3(J,I)*V3(J,I)
2300      NEXT J
2310  '   ---------------
2320      RETURN
2330  '
2340  '       Looking for the point having the smallest deviation
2350  '       ---------------------------------------------------
2360      SUM=SUM(1) : NB=1
2370  '   ---------------
2380      FOR I=2 TO 9
2390         IF SUM(I)<SUM THEN SUM=SUM(I) : NB=I
2400      NEXT I
2410  '   ---------------
2420           IF NB=1 THEN 4200                          'last point of a step
2430      ON NB-1 GOTO 2450,2670,2880,3100,3320,3540,3760,3980 'the pnt with the
  smallest dev. is not the central pnt of the basic grid so computation of the
  missing values of the new grid after having moved its center on the pnt with
  min dev.
2440  '
2450  '       NB=2                  'minimum sum in position 2 (N)
2460  '       ---------------------------------------------------
2470      LAT(5)=LAT(4) : LONG(5)=LONG(4) : DIST(5)=DIST(4) : SUM(5)=SUM(4)
2480      FOR J=1 TO N : V1(J,5)=V1(J,4):V2(J,5)=V2(J,4):V3(J,5)=V3(J,4): NEXT J
2490      LAT(6)=LAT(1) : LONG(6)=LONG(1) : DIST(6)=DIST(1) : SUM(6)=SUM(1)
2500      FOR J=1 TO N : V1(J,6)=V1(J,1):V2(J,6)=V2(J,1):V3(J,6)=V3(J,1): NEXT J
2510      LAT(7)=LAT(8) : LONG(7)=LONG(8) : DIST(7)=DIST(8) : SUM(7)=SUM(8)
2520      FOR J=1 TO N : V1(J,7)=V1(J,8):V2(J,7)=V2(J,8):V3(J,7)=V3(J,8): NEXT J
2530      LAT(8)=LAT(9) : LONG(8)=LONG(9) : DIST(8)=DIST(9) : SUM(8)=SUM(9)
2540      FOR J=1 TO N : V1(J,8)=V1(J,9):V2(J,8)=V2(J,9):V3(J,8)=V3(J,9): NEXT J
2550      LAT(1)=LAT(2) : LONG(1)=LONG(2) : DIST(1)=DIST(2) : SUM(1)=SUM(2)
2560      FOR J=1 TO N : V1(J,1)=V1(J,2):V2(J,1)=V2(J,2):V3(J,1)=V3(J,2): NEXT J
2570      LAT(4)=LAT(3) : LONG(4)=LONG(3) : DIST(4)=DIST(3) : SUM(4)=SUM(3)
2580      FOR J=1 TO N : V1(J,4)=V1(J,3):V2(J,4)=V2(J,3):V3(J,4)=V3(J,3): NEXT J
2590      LAT(2)=CINT((LAT(1)+D(E))*100)/100 : LONG(2)=LONG(1)
2600      LAT(3)=LAT(2) : LONG(3)=LONG(4)
2610      LAT(9)=LAT(2) : LONG(9)=LONG(7)
2620      I=2 : GOSUB 1810
2630      I=3 : GOSUB 1810
2640      I=9 : GOSUB 1810
2650      GOTO 2340
2660  '
2670  '       NB=3                  'minimum sum in position 3 (NE)
2680  '       ---------------------------------------------------
2690      LAT(6)=LAT(4) : LONG(6)=LONG(4) : DIST(6)=DIST(4) : SUM(6)=SUM(4)
2700      FOR J=1 TO N : V1(J,6)=V1(J,4):V2(J,6)=V2(J,4):V3(J,6)=V3(J,4): NEXT J
2710      LAT(7)=LAT(1) : LONG(7)=LONG(1) : DIST(7)=DIST(1) : SUM(7)=SUM(1)
2720      FOR J=1 TO N : V1(J,7)=V1(J,1):V2(J,7)=V2(J,1):V3(J,7)=V3(J,1): NEXT J
2730      LAT(8)=LAT(2) : LONG(8)=LONG(2) : DIST(8)=DIST(2) : SUM(8)=SUM(2)
2740      FOR J=1 TO N : V1(J,8)=V1(J,2):V2(J,8)=V2(J,2):V3(J,8)=V3(J,2): NEXT J
2750      LAT(1)=LAT(3) : LONG(1)=LONG(3) : DIST(1)=DIST(3) : SUM(1)=SUM(3)
2760      FOR J=1 TO N : V1(J,1)=V1(J,3):V2(J,1)=V2(J,3):V3(J,1)=V3(J,3): NEXT J
2770      LAT(2)=CINT((LAT(1)+D(E))*100)/100 : LONG(2)=LONG(1)
2780      LAT(3)=LAT(2) : LONG(3)=CINT((LONG(1)+D(E))*100)/100
2790      LAT(4)=LAT(1) : LONG(4)=LONG(3)
2800      LAT(5)=LAT(6) : LONG(5)=LONG(3)
2810      LAT(9)=LAT(2) : LONG(9)=LONG(8)
2820      I=2 : GOSUB 1810
2830      I=3 : GOSUB 1810
2840      I=4 : GOSUB 1810
2850      I=5 : GOSUB 1810
2860      I=9 : GOSUB 1810
2870      GOTO 2340
2880  '
2890  '       NB=4                  'minimum sum in position 4 (E)
2900  '       ---------------------------------------------------
2910      LAT(9)=LAT(2) : LONG(9)=LONG(2) : DIST(9)=DIST(2) : SUM(9)=SUM(2)
2920      FOR J=1 TO N : V1(J,9)=V1(J,2):V2(J,9)=V2(J,2):V3(J,9)=V3(J,2): NEXT J
2930      LAT(8)=LAT(1) : LONG(8)=LONG(1) : DIST(8)=DIST(1) : SUM(8)=SUM(1)
2940      FOR J=1 TO N : V1(J,8)=V1(J,1):V2(J,8)=V2(J,1):V3(J,8)=V3(J,1): NEXT J
2950      LAT(7)=LAT(6) : LONG(7)=LONG(6) : DIST(7)=DIST(6) : SUM(7)=SUM(6)
2960      FOR J=1 TO N : V1(J,7)=V1(J,6):V2(J,7)=V2(J,6):V3(J,7)=V3(J,6): NEXT J
2970      LAT(6)=LAT(5) : LONG(6)=LONG(5) : DIST(6)=DIST(5) : SUM(6)=SUM(5)
2980      FOR J=1 TO N : V1(J,6)=V1(J,5):V2(J,6)=V2(J,5):V3(J,6)=V3(J,5): NEXT J
2990      LAT(1)=LAT(4) : LONG(1)=LONG(4) : DIST(1)=DIST(4) : SUM(1)=SUM(4)
3000      FOR J=1 TO N : V1(J,1)=V1(J,4):V2(J,1)=V2(J,4):V3(J,1)=V3(J,4): NEXT J
3010      LAT(2)=LAT(3) : LONG(2)=LONG(3) : DIST(2)=DIST(3) : SUM(2)=SUM(3)
3020      FOR J=1 TO N : V1(J,2)=V1(J,3):V2(J,2)=V2(J,3):V3(J,2)=V3(J,3): NEXT J
3030      LAT(3)=LAT(2) : LONG(3)=CINT((LONG(1)+D(E))*100)/100
3040      LAT(4)=LAT(1) : LONG(4)=LONG(3)
3050      LAT(5)=LAT(6) : LONG(5)=LONG(3)
3060      I=3 : GOSUB 1810
3070      I=4 : GOSUB 1810
3080      I=5 : GOSUB 1810
3090      GOTO 2340
3100  '
3110  '       NB=5                  'minimum sum in position 5 (SE)
3120  '       ---------------------------------------------------
3130      LAT(2)=LAT(4) : LONG(2)=LONG(4) : DIST(2)=DIST(4) : SUM(2)=SUM(4)
3140      FOR J=1 TO N : V1(J,2)=V1(J,4):V2(J,2)=V2(J,4):V3(J,2)=V3(J,4): NEXT J
3150      LAT(9)=LAT(1) : LONG(9)=LONG(1) : DIST(9)=DIST(1) : SUM(9)=SUM(1)
3160      FOR J=1 TO N : V1(J,9)=V1(J,1):V2(J,9)=V2(J,1):V3(J,9)=V3(J,1): NEXT J
3170      LAT(8)=LAT(6) : LONG(8)=LONG(6) : DIST(8)=DIST(6) : SUM(8)=SUM(6)
3180      FOR J=1 TO N : V1(J,8)=V1(J,6):V2(J,8)=V2(J,6):V3(J,8)=V3(J,6): NEXT J
3190      LAT(1)=LAT(5) : LONG(1)=LONG(5) : DIST(1)=DIST(5) : SUM(1)=SUM(5)
3200      FOR J=1 TO N : V1(J,1)=V1(J,5):V2(J,1)=V2(J,5):V3(J,1)=V3(J,5): NEXT J
3210      LAT(3)=LAT(2) : LONG(3)=CINT((LONG(1)+D(E))*100)/100
3220      LAT(4)=LAT(1) : LONG(4)=LONG(3)
3230      LAT(5)=CINT((LAT(1)-D(E))*100)/100 : LONG(5)=LONG(3)
3240      LAT(6)=LAT(5) : LONG(6)=LONG(1)
3250      LAT(7)=LAT(5) : LONG(7)=LONG(8)
3260      I=3 : GOSUB 1810
3270      I=4 : GOSUB 1810
3280      I=5 : GOSUB 1810
3290      I=6 : GOSUB 1810
3300      I=7 : GOSUB 1810
3310      GOTO 2340
3320  '
3330  '       NB=6                  'minimum sum in position 6 (S)
3340  '       ---------------------------------------------------
3350      LAT(9)=LAT(8) : LONG(9)=LONG(8) : DIST(9)=DIST(8) : SUM(9)=SUM(8)
3360      FOR J=1 TO N : V1(J,9)=V1(J,8):V2(J,9)=V2(J,8):V3(J,9)=V3(J,8): NEXT J
3370      LAT(2)=LAT(1) : LONG(2)=LONG(1) : DIST(2)=DIST(1) : SUM(2)=SUM(1)
3380      FOR J=1 TO N : V1(J,2)=V1(J,1):V2(J,2)=V2(J,1):V3(J,2)=V3(J,1): NEXT J
3390      LAT(3)=LAT(4) : LONG(3)=LONG(4) : DIST(3)=DIST(4) : SUM(3)=SUM(4)
3400      FOR J=1 TO N : V1(J,3)=V1(J,4):V2(J,3)=V2(J,4):V3(J,3)=V3(J,4): NEXT J
3410      LAT(4)=LAT(5) : LONG(4)=LONG(5) : DIST(4)=DIST(5) : SUM(4)=SUM(5)
```

```
3420    FOR J=1 TO N : V1(J,4)=V1(J,5):V2(J,4)=V2(J,5):V3(J,4)=V3(J,5): NEXT J
3430    LAT(1)=LAT(6) : LONG(1)=LONG(6) : DIST(1)=DIST(6) : SUM(1)=SUM(6)
3440    FOR J=1 TO N : V1(J,1)=V1(J,6):V2(J,1)=V2(J,6):V3(J,1)=V3(J,6): NEXT J
3450    LAT(8)=LAT(7) : LONG(8)=LONG(7) : DIST(8)=DIST(7) : SUM(8)=SUM(7)
3460    FOR J=1 TO N : V1(J,8)=V1(J,7):V2(J,8)=V2(J,7):V3(J,8)=V3(J,7): NEXT J
3470    LAT(5)=CINT((LAT(1)-D(E))*100)/100 : LONG(5)=LONG(3)
3480    LAT(6)=LAT(5) : LONG(6)=LONG(1)
3490    LAT(7)=LAT(5) : LONG(7)=LONG(8)
3500    I=5 : GOSUB 1810
3510    I=6 : GOSUB 1810
3520    I=7 : GOSUB 1810
3530    GOTO 2340
3540 '
3550 '      NB=7                'minimum sum in position 7 (SW)
3560 '      ---------------------------------------------------
3570    LAT(2)=LAT(8) : LONG(2)=LONG(8) : DIST(2)=DIST(8) : SUM(2)=SUM(8)
3580    FOR J=1 TO N : V1(J,2)=V1(J,8):V2(J,2)=V2(J,8):V3(J,2)=V3(J,8): NEXT J
3590    LAT(3)=LAT(1) : LONG(3)=LONG(1) : DIST(3)=DIST(1) : SUM(3)=SUM(1)
3600    FOR J=1 TO N : V1(J,3)=V1(J,1):V2(J,3)=V2(J,1):V3(J,3)=V3(J,1): NEXT J
3610    LAT(4)=LAT(6) : LONG(4)=LONG(6) : DIST(4)=DIST(6) : SUM(4)=SUM(6)
3620    FOR J=1 TO N : V1(J,4)=V1(J,6):V2(J,4)=V2(J,6):V3(J,4)=V3(J,6): NEXT J
3630    LAT(1)=LAT(7) : LONG(1)=LONG(7) : DIST(1)=DIST(7) : SUM(1)=SUM(7)
3640    FOR J=1 TO N : V1(J,1)=V1(J,7):V2(J,1)=V2(J,7):V3(J,1)=V3(J,7): NEXT J
3650    LAT(5)=CINT((LAT(1)-D(E))*100)/100 : LONG(5)=LONG(3)
3660    LAT(6)=LAT(5) : LONG(6)=LONG(1)
3670    LAT(7)=LAT(5) : LONG(7)=CINT((LONG(1)-D(E))*100)/100
3680    LAT(8)=LAT(1) : LONG(8)=LONG(7)
3690    LAT(9)=LAT(2) : LONG(9)=LONG(7)
3700    I=5 : GOSUB 1810
3710    I=6 : GOSUB 1810
3720    I=7 : GOSUB 1810
3730    I=8 : GOSUB 1810
3740    I=9 : GOSUB 1810
3750    GOTO 2340
3760 '
3770 '      NB=8                'minimum sum in position 8 (W)
3780 '      ---------------------------------------------------
3790    LAT(3)=LAT(2) : LONG(3)=LONG(2) : DIST(3)=DIST(2) : SUM(3)=SUM(2)
3800    FOR J=1 TO N : V1(J,3)=V1(J,2):V2(J,3)=V2(J,2):V3(J,3)=V3(J,2): NEXT J
3810    LAT(4)=LAT(1) : LONG(4)=LONG(1) : DIST(4)=DIST(1) : SUM(4)=SUM(1)
3820    FOR J=1 TO N : V1(J,4)=V1(J,1):V2(J,4)=V2(J,1):V3(J,4)=V3(J,1): NEXT J
3830    LAT(5)=LAT(6) : LONG(5)=LONG(6) : DIST(5)=DIST(6) : SUM(5)=SUM(6)
3840    FOR J=1 TO N : V1(J,5)=V1(J,6):V2(J,5)=V2(J,6):V3(J,5)=V3(J,6): NEXT J
3850    LAT(6)=LAT(7) : LONG(6)=LONG(7) : DIST(6)=DIST(7) : SUM(6)=SUM(7)
3860    FOR J=1 TO N : V1(J,6)=V1(J,7):V2(J,6)=V2(J,7):V3(J,6)=V3(J,7): NEXT J
3870    LAT(1)=LAT(8) : LONG(1)=LONG(8) : DIST(1)=DIST(8) : SUM(1)=SUM(8)
3880    FOR J=1 TO N : V1(J,1)=V1(J,8):V2(J,1)=V2(J,8):V3(J,1)=V3(J,8): NEXT J
3890    LAT(2)=LAT(9) : LONG(2)=LONG(9) : DIST(2)=DIST(9) : SUM(2)=SUM(9)
3900    FOR J=1 TO N : V1(J,2)=V1(J,9):V2(J,2)=V2(J,9):V3(J,2)=V3(J,9): NEXT J
3910    LAT(7)=LAT(5) : LONG(7)=CINT((LONG(1)-D(E))*100)/100
3920    LAT(8)=LAT(1) : LONG(8)=LONG(7)
3930    LAT(9)=LAT(2) : LONG(9)=LONG(7)
3940    I=7 : GOSUB 1810
3950    I=8 : GOSUB 1810
3960    I=9 : GOSUB 1810
3970    GOTO 2340
3980 '
3990 '      NB=9                'minimum sum in position 9 (NW)
4000 '      ---------------------------------------------------
4010    LAT(4)=LAT(2) : LONG(4)=LONG(2) : DIST(4)=DIST(2) : SUM(4)=SUM(2)
4020    FOR J=1 TO N : V1(J,4)=V1(J,2):V2(J,4)=V2(J,2):V3(J,4)=V3(J,2): NEXT J
4030    LAT(5)=LAT(1) : LONG(5)=LONG(1) : DIST(5)=DIST(1) : SUM(5)=SUM(1)
4040    FOR J=1 TO N : V1(J,5)=V1(J,1):V2(J,5)=V2(J,1):V3(J,5)=V3(J,1): NEXT J
4050    LAT(6)=LAT(8) : LONG(6)=LONG(8) : DIST(6)=DIST(8) : SUM(6)=SUM(8)
4060    FOR J=1 TO N : V1(J,6)=V1(J,8):V2(J,6)=V2(J,8):V3(J,6)=V3(J,8): NEXT J
4070    LAT(1)=LAT(9) : LONG(1)=LONG(9) : DIST(1)=DIST(9) : SUM(1)=SUM(9)
4080    FOR J=1 TO N : V1(J,1)=V1(J,9):V2(J,1)=V2(J,9):V3(J,1)=V3(J,9): NEXT J
4090    LAT(2)=CINT((LAT(1)+D(E))*100)/100 : LONG(2)=LONG(1)
4100    LAT(3)=LAT(2) : LONG(3)=LONG(4)
4110    LAT(7)=LAT(5) : LONG(7)=CINT((LONG(1)-D(E))*100)/100
4120    LAT(8)=LAT(1) : LONG(8)=LONG(7)
4130    LAT(9)=LAT(2) : LONG(9)=LONG(7)
4140    I=2 : GOSUB 1810
4150    I=3 : GOSUB 1810
4160    I=7 : GOSUB 1810
4170    I=8 : GOSUB 1810
4180    I=9 : GOSUB 1810
4190    GOTO 2340
4200 '
4210 '      Standard deviation in degrees
4220 '      -----------------------------
4230    SDEV=SQR(SUM(1)/(N-1))
4240 '
4250 '      Printing the results of the subloop on the screen
4260 '      ------------------------------------------------
4270    PRINT USING "Lat = ###.##";LAT(1);
4280    PRINT USING "    Long = ####.##";LONG(1);
4290    PRINT USING " Mean distance = ####.##";DIST(1)*10000/90; : PRINT " km"
4300    PRINT USING "Standard deviation = ##.##";SDEV; : PRINT " deg"
4310 NEXT E
4320 '---------------
4330 ' ---------------------------------------------------------------------------
4340 '
4350 '      PRINTING PART
4360 '      =============
4370     IF L>1 THEN 4590                    'to print the title only once
4380 BEEP : PRINT
4390   INPUT "PRINTER READY (Y/N) ";R$ : R$=LEFT$(R$,1)
4400      IF R$<>"Y" AND R$<>"y" THEN 4390
4410 '
4420 '    Preparing the printer
4430 '    ---------------------
4440 LPRINT CHR$(27)"n" : LPRINT CHR$(27)"L014" : LPRINT CHR$(27)"!"
4450 '
4460 '    Printing the title and subtitle
4470 '    -------------------------------
4480 T=INT(72-14-5-LEN(T$))/2
4490 LPRINT TAB(T) CHR$(27)"X" T$                          'title
4500 LPRINT CHR$(27)"E" : LPRINT CHR$(27)"Y"
4510 LPRINT : LPRINT
4520 LPRINT TAB(5) "Looking for the best fitting pole : criterion is azimuth"
4530 LPRINT CHR$(27)CHR$(34) : LPRINT CHR$(27)"N"
4540 LPRINT
4550 LPRINT USING "Position tested : Lat = ###.##";LAT;
4560 LPRINT USING "    Long = ####.##";LONG
4570 LPRINT
```

```
4580 '
4590 '    Printing of the results of each loop on the line printer
4600 '    ------------------------------------------------------
4610       IF L=1 THEN 4650
4620       IF CINT(LAT(1)*100)=CINT(LAT*100) AND CINT(LONG(1)*100)=CINT(LONG*1
00) THEN 4940               'are the results of the two last loops the same ?
4630 LPRINT
4640 LPRINT "Loop  ";L
4650 LPRINT USING "Center position :  Lat = ###.##";LAT(1);
4660 LPRINT USING "  Long = ####.##";LONG(1)
4670 LPRINT USING "Mean distance to the pole : ####.##";DIST(1)*10000/90;
4680 LPRINT " km"
4690 LPRINT : LPRINT
4700 LPRINT "  LAT      LONG     ODIR     DPOLE     DEACH     CDIR     DDIR"
4710 LPRINT "  deg       deg      deg       km        km       deg      deg"
4720 LPRINT
4730 '---------------
4740 FOR J=1 TO N
4750    LPRINT USING "###.##  ";T(J,1);                  'latitude
4760    LPRINT USING "####.##  ";T(J,2);                 'longitude
4770    LPRINT USING "####.##  ";T(J,3);                 'direction
4780    LPRINT USING "####.#  ";V1(J,1)*10000/90;        'distance to the pole
4790    LPRINT USING "####.#  ";T(J,4)*10000/90;         'dist. between points
4800    LPRINT USING "####.## ";V2(J,1);                 'computed direction
4810    LPRINT USING "###.##";V3(J,1)                    'direction deviation
4820 NEXT J
4830 '---------------
4840 LPRINT : LPRINT
4850 LPRINT USING "Standard deviation : ##.##";SDEV; : LPRINT " deg"
4860 LPRINT
4870 '
4880 '    Preparing the next loop
4890 '    -----------------------
4900 L=L+1
4910 LAT=LAT(1) : LONG=LONG(1) : DIST=DIST(1)
4920 GOTO 1570
4930 LPRINT
4940 LPRINT "Results verified."      'the two last loops show the same results
4950 '-------------------------------------------------------------------------
4960 '
4970 '     DRAWING PART
4980 '     ============
4990 '
5000 '    Sum of the distance between points, just for the plot
5010 '    -----------------------------------------------------
5020 S=0 : S(1)=0
5030 '---------------
5040 FOR I=2 TO N
5050    S=S+T(I,4) : S(I)=S
5060 NEXT I
5070 '---------------
5080 '
5090 '    Looking for the minimum and maximum direction
5100 '    ---------------------------------------------
5110 MIN=T(1,3) : MAX=MIN
5120 '---------------
5130 FOR I=1 TO N
5140       IF T(I,3)<MIN THEN MIN=T(I,3)
5150       IF T(I,3)>MAX THEN MAX=T(I,3)
5160 NEXT I
5170 '---------------
5180 '---------------
5190 FOR I=1 TO N
5200       IF V2(I,1)-2*SDEV<MIN THEN MIN=V2(I,1)-2*SDEV
5210       IF V2(I,1)+2*SDEV>MAX THEN MAX=V2(I,1)+2*SDEV
5220 NEXT I
5230 '---------------
5240 '
5250 '    Asking questions for the graph
5260 '    ------------------------------
5270 PRINT : PRINT
5280 PRINT "Minimum = ";MIN;"        Maximum = ";MAX
5290   INPUT "Minimum value on the graph ";MIN
5300   INPUT "Maximum value on the graph ";MAX
5310   INPUT "   step for label printing ";ST
5320 PRINT
5330 '
5340 '    Instructions to modify and copy the graph on the line printer
5350 '    -------------------------------------------------------------
5360 PRINT "  Now the program will display a graph of the original direction"
5370 PRINT "and the computed direction."
5380 PRINT "  By pressing the <G> key while the graph is being diplayed you"
5390 PRINT "will be able to change the parameters of the graph."
5400 PRINT "  By pressing the <C> key a copy of the screen will be made on"
5410 PRINT "the printer, it will take about half an hour to complete, but"
5420 PRINT "before do not forget to change the paper."
5430 PRINT "  The program will end by pressing any other key."
5440 PRINT : PRINT
5450   INPUT "READY (Y/N) ";R$ : R$=LEFT$(R$,1)
5460       IF R$<>"Y" AND R$<>"y" THEN 5450
5470 SCREEN 1 : CLS                  'setting the screen for graphics
5480 '
5490 '    Drawing the axis
5500 '    ----------------
5510 LINE (69,21)-(69,172)  'draw distance-between-points axis twice to have
5520 LINE (70,21)-(70,172)  'a line as heavy as the direction axis
5530 LINE -(560,172)        'draw the direction axis
5540 '
5550 '    Drawing of the line linking the real data
5560 '    -----------------------------------------
5570 PSET ((T(1,3)-MIN)*490/(MAX-MIN)+70,S(1)*151/S+21)
5580 '---------------
5590 FOR I=2 TO N
5600    LINE -((T(I,3)-MIN)*490/(MAX-MIN)+70,S(I)*151/S+21)
5610 NEXT I
5620 '---------------
5630 '
5640 '    Drawing of the line linking the computed direction three times
5650 '    --------------------------------------------------------------
5660 PSET ((V2(1,1)-MIN)*490/(MAX-MIN)+70,S(1)*151/S+21)
5670 '---------------
5680 FOR I=1 TO N
5690 LINE -((V2(I,1)-MIN)*490/(MAX-MIN)+70,S(I)*151/S+21)
5700 NEXT I
5710 '---------------
```

```
5720 PSET ((V2(1,1)-MIN)*490/(MAX-MIN)+69,S(1)*151/S+21)
5730 '---------------
5740 FOR I=1 TO N
5750 LINE -((V2(I,1)-MIN)*490/(MAX-MIN)+69,S(I)*151/S+21)
5760 NEXT I
5770 '---------------
5780 PSET ((V2(1,1)-MIN)*490/(MAX-MIN)+71,S(1)*151/S+21)
5790 '---------------
5800 FOR I=1 TO N
5810 LINE -((V2(I,1)-MIN)*490/(MAX-MIN)+71,S(I)*151/S+21)
5820 NEXT I
5830 '---------------
5840 '
5850 '    Drawing the 70%-error-interval lines and printing the percentage
5860 '    ------------------------------------------------------------
5870 LOCATE 2,INT(((V2(1,1)-SDEV-MIN)*490/(MAX-MIN)+71)/8) : PRINT "70%"
5880 PSET ((V2(1,1)-SDEV-MIN)*490/(MAX-MIN)+71,S(1)*151/S+21)
5890 '---------------
5900 FOR I=1 TO N
5910 LINE -((V2(I,1)-SDEV-MIN)*490/(MAX-MIN)+71,S(I)*151/S+21)
5920 NEXT I
5930 '---------------
5940 LOCATE 2,INT(((V2(1,1)+SDEV-MIN)*490/(MAX-MIN)+71)/8)
5950 PSET ((V2(1,1)+SDEV-MIN)*490/(MAX-MIN)+71,S(1)*151/S+21)
5960 '---------------
5970 FOR I=1 TO N
5980 LINE -((V2(I,1)+SDEV-MIN)*490/(MAX-MIN)+71,S(I)*151/S+21)
5990 NEXT I
6000 '---------------
6010 '
6020 '    Drawing the 95%-error-interval lines and printing the percentage
6030 '    ------------------------------------------------------------
6040 LOCATE 2,INT(((V2(1,1)-2*SDEV-MIN)*490/(MAX-MIN)+71)/8)
6050 PSET ((V2(1,1)-2*SDEV-MIN)*490/(MAX-MIN)+71,S(1)*151/S+21)
6060 '---------------
6070 FOR I=1 TO N
6080 LINE -((V2(I,1)-2*SDEV-MIN)*490/(MAX-MIN)+71,S(I)*151/S+21)
6090 NEXT I
6100 '---------------
6110 LOCATE 2,INT(((V2(1,1)+2*SDEV-MIN)*490/(MAX-MIN)+71)/8) : PRINT "95%"
6120 PSET ((V2(1,1)+2*SDEV-MIN)*490/(MAX-MIN)+71,S(1)*151/S+21)
6130 '---------------
6140 FOR I=1 TO N
6150 LINE -((V2(I,1)+2*SDEV-MIN)*490/(MAX-MIN)+71,S(I)*151/S+21)
6160 NEXT I
6170 '---------------
6180 '
6190 '    Drawing the tick marks along the distance-between-points axis
6200 '    and printing the point numbers
6210 '    ------------------------------------------------------------
6220 '---------------
6230 FOR I=1 TO N
6240    LINE (60,S(I)*151/S+21)-(70,S(I)*151/S+21)
6250    LOCATE INT((S(I)*151/S+21)/9)+1,5 : PRINT USING "##";I
6260 NEXT I
6270 '---------------
6280 '
6290 '    Printing the distance-between-points axis title
6300 '    -----------------------------------------------
6310 LOCATE 1,6 : PRINT "POINT"
6320 LOCATE 2,6 : PRINT "NUMBER"
6330 '
6340 '    Printing the pole position and mean distance
6350 '    --------------------------------------------
6360 LOCATE 1,15
6370 PRINT USING "POLE POSITION :  LAT = ###.##";LAT(1);
6380 PRINT USING ", LONG = ####.##";LONG(1);
6390 PRINT USING ", DIST = ####.#";DIST*10000/90; : PRINT " KM"
6400 '
6410 '    Drawing the tick marks along the direction axis twice
6420 '    and printing the labels
6430 '    -------------------------------------------------------
6440 '---------------
6450 FOR I=MIN TO MAX STEP ST
6460    LINE (69+(I-MIN)*490/(MAX-MIN),172)-(69+(I-MIN)*490/(MAX-MIN),178)
6470    LINE (70+(I-MIN)*490/(MAX-MIN),172)-(70+(I-MIN)*490/(MAX-MIN),178)
6480    LOCATE 21,(70+(I-MIN)*490/(MAX-MIN))/8-2
6490    PRINT USING "###.#";I
6500 NEXT I
6510 '---------------
6520 '
6530 '    Printing the direction axis title
6540 '    ---------------------------------
6550 LOCATE 19,72 : PRINT "DIRECTION"
6560 LOCATE 21,75 : PRINT "DEG"
6570 '
6580 '    Printing the title and underlining it
6590 '    -------------------------------------
6600 LOCATE 23,(40-LEN(T$)/2) : PRINT T$
6610 LINE ((39-INT(LEN(T$)/2))*8,210)-STEP(LEN(T$)*8,0)
6620 '
6630 '    Waiting for and analysing the keyboard entry
6640 '    --------------------------------------------
6650 LOCATE 1,1
6660 Q$=INKEY$ : IF Q$="" THEN 6660      'waiting to change the graph
6670       IF Q$="G" OR Q$="g" THEN SCREEN 0 : LOCATE 24,1 : GOTO 5250
6680       IF Q$<>"C" AND Q$<>"c" THEN 6920
6690 '-------------------------------------------------------------------------
6700 '
6710 '     SCREEN COPY
6720 '     ===========
6730 '    Preparing the printer for graphics
6740 '    ----------------------------------
6750 LPRINT CHR$(27)"p" : LPRINT CHR$(27)"T02"
6760 '
6770 '    Reading the screen values point after point, line after line
6780 '    and printing the lines one after the other
6790 '    ------------------------------------------------------------
6800 '---------------
6810 FOR I=0 TO 224
6820    X1$="" : X2$="" : X3$="" : X4$=""
6830    FOR J=0 TO 159 : X1$=X1$+CHR$(POINT(J,I)/7) : NEXT J
6840    FOR J=160 TO 319 : X2$=X2$+CHR$(POINT(J,I)/7) : NEXT J
6850    FOR J=320 TO 479 : X3$=X3$+CHR$(POINT(J,I)/7) : NEXT J
6860    FOR J=480 TO 639 : X4$=X4$+CHR$(POINT(J,I)/7) : NEXT J
```

```
6870     LPRINT CHR$(27)"G0640"X1$X2$X3$X4$
6880 NEXT I
6890 '---------------
6900 LPRINT CHR$(27)"A" : LPRINT CHR$(27)"E"
                                'setting the printer back to printing mode
6910 GOTO 6620                  'to wait for the next keyboard entry
6920 SCREEN 0                   'setting the screen back to printing mode
6930 END
```

C
★

This program has been written to help you to find the pole fitting best to directions of features distributed on a small circle.

The program will read the data in a data file beginning with line 500, and having a variable named N giving the number of data points. Then each data line must have first the latitude of the point, then the longitude, and at last the azimuth of the feature. The last line number must not exceed 999.

The program will ask you the latitude and longitude of the pole you have chosen to test. Then it will compute the sum of the squares of the differences between the real value and the azimuth of the small circle at each point. This value must be as small as possible so the program will loop and test points within 2 degrees to 0.01 degree of latitude and longitude of the point having the smallest deviation. The results will be verified and a graph will displayed on the screen, and you will be allowed to copy this graph on the printer.

```
Disk ready ? Y
Name of the data files on the disk :
MATRENCH.DAT DRSCMAT5.DAT DRSCMAT4.DAT DRSCMAT2.DAT DRSCMAT1.DAT DRSCMAT3.DAT
DRSCEPR1.DAT DRSCRIVE.DAT DSSCMAT2.DAT DSSCMAT1.DAT DSSCMAT3.DAT DSSCMAT5.DAT
DSSCMAT4.DAT DSSCRIVE.DAT BRKPROFL.DAT PNTPROFL.DAT DRSCRITA.DAT

Name of your data file : ? DRSCMAT1.DAT
Title of the table and the plot :
? MIDDLE AMERICA TRENCH : NORTHERN SET
   Latitude of the pole you want to test ? 30
   Longitude of the pole ? -95

LOOP  1     Step  1   ( D =  2  deg)
Lat =  30.00    Long =  -95.00 Mean distance = 1610.88 km
Standard deviation =  2.31 deg

LOOP  1     Step  2   ( D =  1.5  deg)
Lat =  33.00    Long =  -93.50 Mean distance = 1968.31 km
Standard deviation =  1.75 deg

LOOP  1     Step  3   ( D =  1  deg)
Lat =  33.00    Long =  -93.50 Mean distance = 1968.31 km
Standard deviation =  1.75 deg

LOOP  1     Step  4   ( D =  .7  deg)
Lat =  33.00    Long =  -93.50 Mean distance = 1968.31 km
Standard deviation =  1.75 deg

LOOP  1     Step  5   ( D =  .5  deg)
Lat =  33.00    Long =  -93.50 Mean distance = 1968.31 km
Standard deviation =  1.75 deg
```

```
LOOP  1     Step  6   ( D =  .35  deg)
Lat =  33.00    Long =  -93.50 Mean distance = 1968.31 km
Standard deviation =  1.75 deg

LOOP  1     Step  7   ( D =  .2  deg)
Lat =  31.80    Long =  -94.30 Mean distance = 1815.95 km
Standard deviation =  1.65 deg

LOOP  1     Step  8   ( D =  .15  deg)
Lat =  31.50    Long =  -94.45 Mean distance = 1780.16 km
Standard deviation =  1.64 deg

LOOP  1     Step  9   ( D =  .1  deg)
Lat =  31.50    Long =  -94.45 Mean distance = 1780.16 km
Standard deviation =  1.64 deg

LOOP  1     Step  10   ( D =  .07  deg)
Lat =  31.50    Long =  -94.45 Mean distance = 1780.16 km
Standard deviation =  1.64 deg

LOOP  1     Step  11   ( D =  .05  deg)
Lat =  31.50    Long =  -94.45 Mean distance = 1780.16 km
Standard deviation =  1.64 deg

LOOP  1     Step  12   ( D =  .03  deg)
Lat =  31.50    Long =  -94.45 Mean distance = 1780.16 km
Standard deviation =  1.64 deg

LOOP  1     Step  13   ( D =  .02  deg)
Lat =  31.50    Long =  -94.45 Mean distance = 1780.16 km
Standard deviation =  1.64 deg

LOOP  1     Step  14   ( D =  .01  deg)
Lat =  31.51    Long =  -94.45 Mean distance = 1781.12 km
Standard deviation =  1.64 deg

PRINTER READY (Y/N) ? Y
```

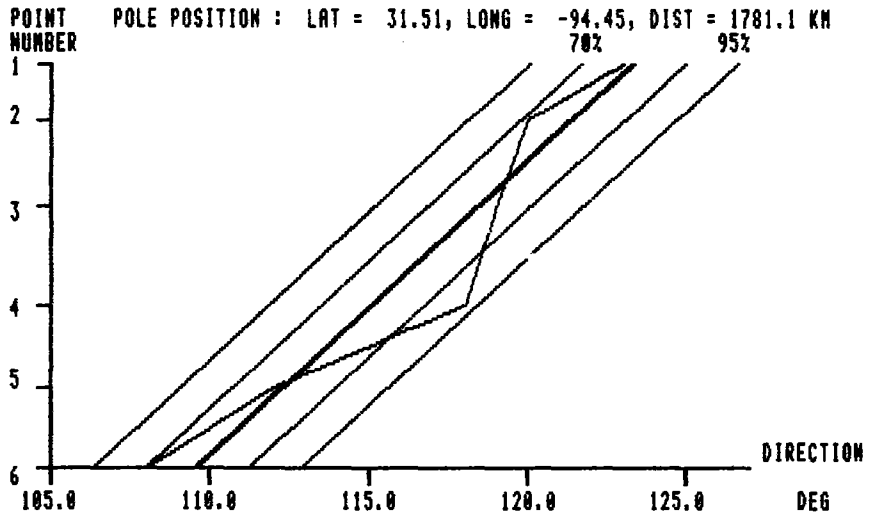
a

```
LOOP  2     Step  1   ( D =  2  deg)
Lat =  31.51    Long =  -94.45 Mean distance = 1781.12 km
Standard deviation =  1.64 deg

LOOP  2     Step  2   ( D =  1.5  deg)
Lat =  31.51    Long =  -94.45 Mean distance = 1781.12 km
Standard deviation =  1.64 deg

LOOP  2     Step  3   ( D =  1  deg)
Lat =  31.51    Long =  -94.45 Mean distance = 1781.12 km
Standard deviation =  1.64 deg

LOOP  2     Step  4   ( D =  .7  deg)
Lat =  31.51    Long =  -94.45 Mean distance = 1781.12 km
Standard deviation =  1.64 deg

LOOP  2     Step  5   ( D =  .5  deg)
Lat =  31.51    Long =  -94.45 Mean distance = 1781.12 km
Standard deviation =  1.64 deg

LOOP  2     Step  6   ( D =  .35  deg)
Lat =  31.51    Long =  -94.45 Mean distance = 1781.12 km
Standard deviation =  1.64 deg

LOOP  2     Step  7   ( D =  .2  deg)
Lat =  31.51    Long =  -94.45 Mean distance = 1781.12 km
Standard deviation =  1.64 deg

LOOP  2     Step  8   ( D =  .15  deg)
Lat =  31.51    Long =  -94.45 Mean distance = 1781.12 km
Standard deviation =  1.64 deg

LOOP  2     Step  9   ( D =  .1  deg)
Lat =  31.51    Long =  -94.45 Mean distance = 1781.12 km
Standard deviation =  1.64 deg

LOOP  2     Step  10   ( D =  .07  deg)
Lat =  31.51    Long =  -94.45 Mean distance = 1781.12 km
Standard deviation =  1.64 deg

LOOP  2     Step  11   ( D =  .05  deg)
Lat =  31.51    Long =  -94.45 Mean distance = 1781.12 km
Standard deviation =  1.64 deg

LOOP  2     Step  12   ( D =  .03  deg)
Lat =  31.51    Long =  -94.45 Mean distance = 1781.12 km
Standard deviation =  1.64 deg

LOOP  2     Step  13   ( D =  .02  deg)
Lat =  31.51    Long =  -94.45 Mean distance = 1781.12 km
Standard deviation =  1.64 deg

LOOP  2     Step  14   ( D =  .01  deg)
Lat =  31.51    Long =  -94.45 Mean distance = 1781.12 km
Standard deviation =  1.64 deg
```

b

```
Minimum =  106.2616         Maximum =  126.5746
Minimum value on the graph ? 105
Maximum value on the graph ? 127
   step for label printing ? 5

  Now the program will display a graph of the original direction
and the computed direction.
  By pressing the (G) key while the graph is being diplayed you
will be able to change the parameters of the graph.
  By pressing the (C) key a copy of the screen will be made on
the printer, it will take about half an hour to complete, but
before do not forget to change the paper.
  The program will end by pressing any other key.
```

READY (Y/N) ? Y



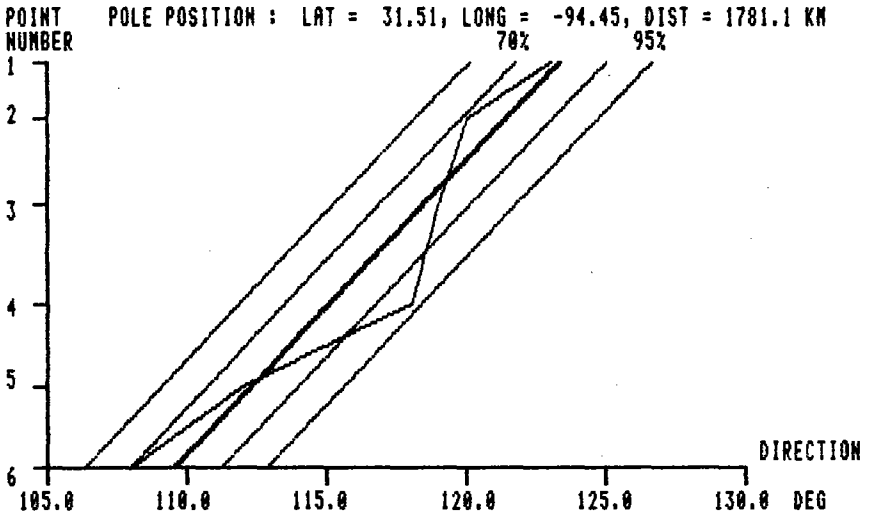
MIDDLE AMERICA TRENCH : NORTHERN SET

G

Minimum = 105 Maximum = 127
Minimum value on the graph ? 105
Maximum value on the graph ? 130
step for label printing ? 5

Now the program will display a graph of the original direction and the computed direction.
By pressing the (G) key while the graph is being diplayed you will be able to change the parameters of the graph.
By pressing the (C) key a copy of the screen will be made on the printer, it will take about half an hour to complete, but before do not forget to change the paper.
The program will end by pressing any other key.

READY (Y/N) ? Y



MIDDLE AMERICA TRENCH : NORTHERN SET

C C

D

a

MIDDLE AMERICA TRENCH : NORTHERN SET

Looking for the best fitting pole : criterion is azimuth

Position tested : Lat = 30.00 Long = -95.00

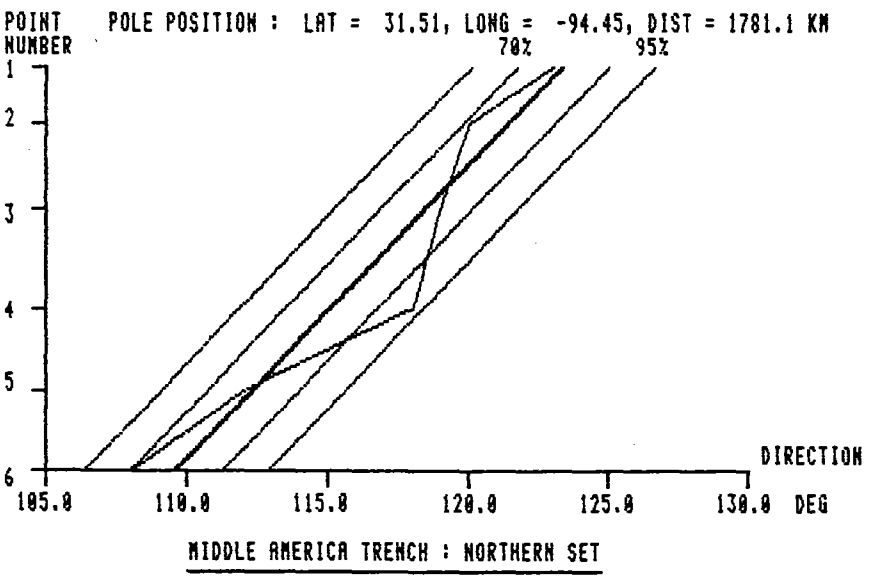
Center position : Lat = 31.51 Long = -94.45
Mean distance to the pole : 1781.12 km

LAT deg	LONG deg	ODIR deg	DPOLE km	DEACH km	CDIR deg	DDIR deg
18.45	-104.67	123.00	1776.9	0.0	123.30	-0.30
18.07	-104.17	120.00	1784.3	67.5	121.36	-1.36
17.62	-103.33	119.00	1783.4	101.9	118.43	0.57
17.17	-102.33	118.00	1780.3	117.2	115.05	2.95
16.83	-101.50	112.00	1779.5	95.9	112.29	-0.29
16.50	-100.67	108.00	1782.3	95.7	109.53	-1.53

Standard deviation : 1.64 deg

b Results verified.

c



MIDDLE AMERICA TRENCH : NORTHERN SET

RELATION PROFONDEUR-DU-FOSSE - RAYON-DU-CERCLE

6.1 - CALCUL DE LA PROFONDEUR MOYENNE DE SEGMENTS DE PROFILS

A - Listing du programme 'MEANDEPT'

B - Copie d'écran d'un exemple d'exécution du programme 'MEANDEPT'*

C - Exemple de sortie sur imprimante*

6.2 - DESSIN DU GRAPHIQUE MONTRANT LA RELATION PROFONDEUR-DU-FOSSE - RAYON-DU-CERCLE

A - Listing du programme 'RADEPTH'

B - Copie d'écran de l'exécution du programme 'RADEPTH'

C - Sortie sur traceur réduite de 67 %

* les lettres a et b de la partie B correspondent à des sorties sur imprimante (partie C) portant le même nom.

6.1

A

```
10 '  ================================================================
20 '
30 '                          PROGRAM 'MEANDEPT'
40 '
50 '                                          Jacqueline ROUMP, Sept. 1984
60 '  ================================================================
70 '
80 CLEAR : CLS : OPTION BASE 1
90
100 '    PRESENTATION OF THE PROGRAM
110 '    ===========================
120 PRINT : PRINT
130 PRINT "   This program computes the mean depth of sections of a profile"
140 PRINT "and its standard deviation."
150 PRINT
160 PRINT "   The first part of the computation is the same as in the program"
170 PRINT "'PROFILE' : longitude, latitude and depth of points along the"
180 PRINT "profile, and position of break points."
190 PRINT
200 PRINT "   The program first computes the distance along the profile, then"
210 PRINT "computes the mean depth of each sections, and at last the standard"
220 PRINT "deviation."
230 PRINT
240 PRINT "   Two tables are printed : first the distances of the sections"
250 PRINT "limits, then the mean depth and standard deviation of each section"
260 '
270 PRINT : PRINT
280 '
290 '    INPUT OF THE DATA INTO THE PROGRAM
300 '    ==================================
310 INPUT "DISK READY (Y/N) ";R$ : R$=LEFT$(R$,1)
320     IF R$<>"Y" AND R$<>"y" THEN 310
330 PRINT
340 PRINT "Names of the data files on the disk : "
350 FILES "*.DAT"                          'list of the files of the disk
360 PRINT
370   INPUT "Name of your breakpoints data file : ";X$
380   INPUT "Name of your profile points data file : ";Y$
390 CLS
400 '
410 '    INPUT OF THE DATA
420 '    =================
430 '    Input of the breakpoints data
440 '    -----------------------------
450 '  opening the file
460 '  ----------------
470 OPEN "R", £1, X$, 8
480 FIELD £1, 4 AS A$, 4 AS B$
490 '
500 '  reading the number of value records
510 '  -----------------------------------
520 GET £1,1
530 N1=CVS(A$)
540 DIM BRK(N1,7)
550 '
560 '  reading the value records
570 '  -------------------------
580 '---------------
590 FOR I%=2 TO N1+1
600    GET £1, I%
610    BRK(I%-1,1)=CVS(A$)
620    BRK(I%-1,2)=CVS(B$)
630 NEXT I%
640 '---------------
650 CLOSE £1                                          'closing the file
660 '
670 '    Input of the profile points data
680 '    --------------------------------
690 '  opening the file
700 '  ----------------
710 OPEN "R", £1, Y$, 10
720 FIELD £1, 4 AS A$, 4 AS B$, 2 AS C$
730 '
740 '  reading the number of value records
750 '  -----------------------------------
760 GET £1,1
770 N2=CVS(A$)
780 DIM PNTS(N2,7)
790 '
800 '  reading the value records
810 '  -------------------------
820 '---------------
830 FOR I%=2 TO N2+1
840    GET £1, I%
850    PNTS(I%-1,1)=CVS(A$)
860    PNTS(I%-1,2)=CVS(B$)
870    PNTS(I%-1,3)=CVI(C$)
880 NEXT I%
890 '---------------
900 CLOSE £1                                          'closing the file
910 ' ------------------------------------------------------------------------
920 '
930 '    ADDITIONAL QUESTIONS
940 '    ====================
950 PRINT : PRINT
960   INPUT "TITLE ";T$
970 PRINT
980 PRINT "Are the data continously increasing or decreasing along"
990   INPUT "              Latitudes (1) or along Longitudes (2) : ";LL
1000       IF LL<>1 AND LL<>2 THEN 970
1010 PRINT "Are the ";
1020       IF L=1 THEN PRINT "Latitudes "; ELSE PRINT "Longitudes ";
1030   INPUT "increasing (1) or decreasing (2) along the profile : ";DIR
1040       IF DIR <>1 AND DIR <>2 THEN 1010
1050       IF DIR=2 THEN DIR=-1
1060 '
1070 '    Table of the limit points
1080 '    -------------------------
1090 PRINT : PRINT
1100   INPUT "How many sections for which you want the mean depth ";S1
1110       IF S1<1 THEN 1090
1120 DIM S1(2*S1,6)                          'section table
1130 '---------------
1140 FOR J=1 TO S1
```

```
1150     PRINT
1160     PRINT "      Interval ";J;
1170       INPUT " First point  : Lat in degrees.minutes ";S1(J*2-1,1)
1180     PRINT SPC(25) ;
1190       INPUT "         Long in degrees.minutes ";S1(J*2-1,2)
1200     PRINT SPC(15) ;
1210       INPUT "   Second Point : Lat in degrees.minutes ";S1(J*2,1)
1220     PRINT SPC(25) ;
1230       INPUT "         Long in degrees.minutes ";S1(J*2,2)
1240 NEXT J
1250 '---------------
1260     CLS : LOCATE 12,5 : PRINT "PLEASE, WAIT ..."
1270 '
1280 '  ------------------------------------------------------------------------
1290 '
1300 '    COMPUTATION PART
1310 '    ================
1320 '
1330 '  Transformation of the latitudes and longitudes in degrees & minutes
1340 '  of the breakpoints table into decimal degrees
1350 '  ------------------------------------------------------------------------
1360 '---------------
1370 FOR I=1 TO N1
1380     B1=BRK(I,1) : B2=BRK(I,2)
1390     BRK(I,3)=FIX(B1)+(B1-FIX(B1))*100/60
1400      BRK(I,4)=FIX(B2)+(B2-FIX(B2))*100/60
1410 NEXT I
1420 '---------------
1430 '
1440 '  Transformation of the latitudes and longitudes in degrees & minutes
1450 '  of the profile points table into decimal degrees
1460 '  ------------------------------------------------------------------------
1470 '---------------
1480 FOR I=1 TO N2
1490     P1=PNTS(I,1) : P2=PNTS(I,2)
1500     PNTS(I,4)=FIX(P1)+(P1-FIX(P1))*100/60
1510     PNTS(I,5)=FIX(P2)+(P2-FIX(P2))*100/60
1520 NEXT I
1530 '---------------
1540 '
1550 FPI£=3.14159265358987£/180                         'PI/180 degrees
1560 '
1570 '   Definition of fonctions to compute the distance between two points
1580 '   on the Earth
1590 '   ------------------------------------------------------------------------
1600 DEF FNA(X,Y,Z,W)=SIN(X*FPI£)*SIN(Z*FPI£)+COS(X*FPI£)*COS(Z*FPI£)*COS((W-Y
)*FPI£)        'cosine of the angular distance between a point of X lat and Y
               long and a point of Z lat and W long
1610 DEF FNB(X)=(-ATN(X/SQR(1-X*X))+1.570796327£)/FPI£*10000/90
               'distance in km for an angular distance of X
1620 Y=6378.135*FPI£          'length of one degree of longitude at the Equator
1630 DEF FNC(X)=COS(ATN(.99333056822£*TAN(X*FPI£)))*Y
               'length of one degree of longitude at latitude X
1640 '
1650 '    Distance between the breakpoints
1660 '    --------------------------------
1670 C(1,5)=0
1680 '---------------
1690 FOR I=2 TO N1
1700    D£=FNA(BRK(I,3),BRK(I,4),BRK(I-1,3),BRK(I-1,4))
1710    BRK(I,5)=FNB(D£)
1720 NEXT I
1730 '---------------
1740 '
1750 '  Distance between the profile points, angular distances being usually
1760 '  too small for the precision of Z100 the Pythagoras theorem is used
1770 '  to compute the distance
1780 '  ------------------------------------------------------------------------
1790 '---------------
1800 FOR I=2 TO N2
1810    E=PNTS(I-1,4)-PNTS(I,4)
1820    H=E*Y
1830    L=(PNTS(I-1,5)-PNTS(I,5))*FNC(E/2+PNTS(I-1,4))
1840    PNTS(I,6)=SQR(H*H+L*L)
1850 NEXT I
1860 '---------------
1870 '
1880 '    Summing the distance along the profile of profile point distances
1890 '    between breakpoints
1900 '    ------------------------------------------------------------------
1910 J=2
1920 '---------------
1930 FOR I=2 TO N2
1940       IF PNTS(I,3+LL)*DIR>BRK(J,2+LL)*DIR THEN J=J+1
1950    E=PNTS(I,4)+(BRK(J-1,3)-PNTS(I,4))/2
1960       IF I=N2 THEN 2000
1970       IF BRK(J,6)=0 THEN H=(BRK(J-1,3)-PNTS(I,4))*Y ELSE 1990
1980    L=(BRK(J-1,4)-PNTS(I,5))*FNC(E) : D=SQR(H*H+L*L) : GOTO 2030
1990       IF PNTS(I+1,3+LL)*DIR<=BRK(J,2+LL)*DIR THEN 2020
2000    E=PNTS(I,4)+(PNTS(I,4)-BRK(J,3))/2 : H=(PNTS(I,4)-BRK(J,3))*Y
2010    L=(BRK(J,4)-PNTS(I,5))*FNC(E) : D=SQR(H*H+L*L)+PNTS(I,6) : GOTO 2030
2020    D=PNTS(I,6)
2030    BRK(J,6)=BRK(J,6)+D
2040 NEXT I
2050 '---------------
2060 '
2070 '    Summing up the breakpoint distances along the profile
2080 '    -----------------------------------------------------
2090 BRK(1,7)=0
2100 '---------------
2110 FOR I=2 TO N1
2120    BRK(I,7)=BRK(I-1,7)+BRK(I,5)
2130 NEXT I
2140 '---------------
2150 '
2160 '    Summing up the distances along the profile smoothed by using
2170 '    the distance between breakpoints
2180 '    -----------------------------------------------------
2190 J=2
2200 PNTS(1,6)=0 : PNTS(1,7)=0
2210 '---------------
2220 FOR I=2 TO N2
2230       IF PNTS(I,3+LL)*DIR>BRK(J,2+LL)*DIR THEN J=J+1
2240       IF PNTS(I,3+LL)=BRK(J,2+LL) THEN PNTS(I,7)=BRK(J,7) : GOTO 2260
2250    PNTS(I,7)=PNTS(I-1,7)+PNTS(I,6)*BRK(J,5)/BRK(J,6)
2260 NEXT I
2270 '---------------
```

```
2280 '
2290 '    Building up the limits table
2300 '    ----------------------------
2310 ' transformation of lat and long from degrees-minutes to decimal degrees
2320 ' ------------------------------------------------------------------------
2330 '---------------
2340 FOR J=1 TO S1*2
2350    T1=S1(J,1) : T2=S1(J,2)
2360    S1(J,3)=FIX(T1)+(T1-FIX(T1))*100/60
2370    S1(J,4)=FIX(T2)+(T2-FIX(T2))*100/60
2380 NEXT J
2390 '---------------
2400 K=1
2410 ' distance of each segment limits along the profile
2420 ' --------------------------------------------------
2430 '---------------
2440 FOR J=1 TO N2
2450       IF S1(K,2+LL)*DIR>PNTS(J,3+LL)*DIR THEN 2540
2460       IF S1(K,2+LL)=PNTS(J,3+LL) THEN S1(K,5)=PNTS(J,7) : GOTO 2520
2470    E=PNTS(J-1,4)+(S1(K,3)-PNTS(J-1,4))/2
2480    H=(PNTS(J-1,4)-S1(K,3))*Y
2490    L=(S1(K,4)-PNTS(J-1,5))*FNC(E)
2500    D=SQR(H*H+L*L)
2510    S1(K,5)=D*(PNTS(J,7)-PNTS(J-1,7))/PNTS(J,6)+PNTS(J-1,7)
2520    K=K+1
2530       IF K>S1*2 THEN 2570
2540 NEXT J
2550 '---------------
2560 '
2570 '    Computing the mean depth for each interval
2580 '    ------------------------------------------
2590 I=1
2600 '---------------
2610 FOR J=1 TO S1
2620       IF S1(J*2-1,2+LL)*DIR>PNTS(I,3+LL)*DIR THEN I=I+1 : GOTO 2620
                                  'looking for the first point
2630       IF S1(J*2-1,2+LL)=PNTS(I,3+LL) THEN AREA(J)=0 : GOTO 2670
2640       IF I=1 THEN D1=PNTS(1,3)
             ELSE D1=PNTS(I,3)-(PNTS(I,3)-PNTS(I-1,3))*(PNTS(I,7)-S1(J*2-1,
5))/(PNTS(I,7)-PNTS(I-1,7))                    'depth of the first point
2650    AREA(J)=(PNTS(I,7)-S1(J*2-1,5))*D1
2660          '---------------
2670          FOR I1=I+1 TO N2
2680                IF PNTS(I1,3+LL)*DIR<=S1(J*2,2+LL)*DIR THEN 2720
                                       'point of the section
2690             D2=PNTS(I1,3)-(PNTS(I1,3)-PNTS(I1-1,3))*(PNTS(I1,7)-S1(J*2,
5))/(PNTS(I1,7)-PNTS(I1-1,7))                  'depth of the last point
2700             AREA(J)=AREA(J)+(S1(J*2,5)-PNTS(I-1,7))*D2
2710             GOTO 2790
2720             AREA(J)=AREA(J)+(PNTS(I1,7)-PNTS(I1-1,7))*(PNTS(I1,3)+PNTS(
I1-1,3))/2
2730                IF PNTS(I1,3+LL)*DIR=S1(J*2,2+LL)*DIR THEN I1=I1+1
                       : GOTO 2790             'last point of a section
2740          NEXT I1
2750          '---------------
2760    '
2770    '. mean depth of a section
2780    ' ------------------------
2790    MEAND(J)=AREA(J)/(S1(J*2,5)-S1(J*2-1,5))
2800    '
2810    '  computation of the variance
2820    ' ----------------------------
2830       IF S1(J*2-1,2+LL)*DIR=PNTS(I,3+LL)  THEN ST(J)=0
             ELSE ST(J)=(D1-MEAND(J))^2
2840         FOR I2=I TO I1-1
2850            ST(J)=ST(J)+(PNTS(I2,3)-MEAND(J))^2
2860         NEXT I2
2870       IF S1(J*2,2+LL)*DIR=PNTS(I1-1,3+LL)*DIR
              THEN ST(J)=ST(J)+(PNTS(I2-1,3)-MEAND(J))^2
                ELSE ST(J)=ST(J)+(D2-MEAND(J))^2
2880    '
2890    '  standard deviation
2900    ' --------------------
2910    STDEV(J)=SQR(1/(I1-1-1)*ST(J))
2920    I=I1-1
2930 NEXT J
2940 '---------------
2950 '
2960 '  ------------------------------------------------------------------------
2970 '
2980 '    PRINTING PART
2990 '    =============
3000 '
3010 BEEP
3020 PRINT : PRINT
3030 '
3040 '    Limits table
3050 '    ------------
3060   INPUT "Ready to print the limits table (Y/N) ";R$
3070         R$=LEFT$(R$,1) : IF R$()"Y" AND R$()"y" THEN 3440
3080 '
3090 '  preparing the printer
3100 ' ----------------------
3110 LPRINT CHR$(27)"E"CHR$(27)"L015"CHR$(27)"!"CHR$(27)"n"
3120 LPRINT SPC(30-LEN(T$)/2)CHR$(27)"X"T$             'printing the title
3130 LPRINT CHR$(27)"Y" : LPRINT CHR$(27)"N"
3140 LPRINT : LPRINT : LPRINT
3150 LPRINT "Limits table"                          'printing the subtitle
3160 LPRINT CHR$(27)"E" : LPRINT CHR$(27)CHR$(34)
3170 LPRINT
3180 '
3190 '  printing the column headings
3200 ' -----------------------------
3210 LPRINT "Section   Latitude   Longitude   Latitude  Longitude     Distorg"
3220 LPRINT "  No.       in degrees-minutes     in decimal degrees      in km"
3230 '
3240 '  printing the point distances
3250 ' -----------------------------
3260 LPRINT
3270 '---------------
3280 FOR J=1 TO S1*2
3290    A=S1(J,1) : B=S1(J,2)
3300       IF J MOD 2=1 THEN LPRINT USING "  ££   ";INT(J/2)+1;
             ELSE LPRINT "       ";
3310    LPRINT USING "   ££";FIX(A);
3320    LPRINT USING "££.£'  ";(A-FIX(A))*100;
```

```
3330    LPRINT USING "£££££";FIX(B);
3340    LPRINT USING "££.£'  ";ABS(B-FIX(B))*100;
3350    LPRINT USING "  ££.£££  ";S1(J,3);
3360    LPRINT USING "££££.£££  ";S1(J,4);
3370    LPRINT USING "   ££££.££";S1(J,5)
3380 NEXT J
3390 '---------------
3400 '
3410 '    Printing the results
3420 '    --------------------
3430 PRINT : PRINT
3440   INPUT "Ready to print the results :
                    Mean Depth and Standard Deviation (Y/N) ";R$
3450           R$=LEFT$(R$,1) : IF R$()"Y" AND R$()"y" THEN 3440
3460 '
3470 '  preparing the printer
3480 '  ---------------------
3490 LPRINT CHR$(27)"E"CHR$(27)"L015"CHR$(27)"!"CHR$(27)"n"
3500 LPRINT SPC(30-LEN(T$)/2)CHR$(27)"X"T$              'printing the title
3510 LPRINT CHR$(27)"Y" : LPRINT CHR$(27)"N"
3520 LPRINT : LPRINT : LPRINT
3530 LPRINT "Results"
3540 LPRINT CHR$(27)"E" : LPRINT CHR$(27)CHR$(34)
3550 LPRINT
3560 '
3570 '  printing the column headings
3580 '  ----------------------------
3590 LPRINT "Section     Mean Depth     Standard Dev.     Min. Depth     Max.
Depth"
3600 LPRINT "  No.          in m            in m           in m          in
 m"
3610 '
3620 '  printing the point distances
3630 '  ----------------------------
3640 LPRINT
3650 '---------------
3660 FOR J=1 TO S1
3670    LPRINT USING "   ££";J;
3680    LPRINT USING "          ££££";MEAND(J);
3690    LPRINT USING "             ££££";STDEV(J);
3700    LPRINT USING"            ££££";MEAND(J)-STDEV(J);
3710    LPRINT USING "           ££££";MEAND(J)+STDEV(J)
3720 NEXT J
3730 '---------------
3740 END
```

B

★

This program computes the mean depth of sections of a profile and its standard deviation.

The first part of the computation is the same as in the program 'PROFILE' : longitude, latitude and depth of points along the profile, and position of break points.

The program first computes the distance along the profile, then computes the mean depth of each sections, and at last the standard deviation.

Two tables are printed : first the distances of the sections limits, then the mean depth and standard deviation of each section

```
DISK READY (Y/N) ? Y

Names of the data files on the disk :
MATRENCH.DAT BRKPROFL.DAT PNTPROFL.DAT

Name of your breakpoints data file : ? BRKPROFL.DAT
Name of your profile points data file : ? PNTPROFL.DAT
```

★

```
TITLE ? MIDDLE AMERICA TRENCH AXIS

Are the data.continously increasing or decreasing along
              Latitudes (1) or along Longitudes (2) : ? 2
Are the Longitudes increasing (1) or decreasing (2) along the profile : ? 1


How many sections for which you want the mean depth ? 5

     Interval  1  First point  : Lat in degrees.minutes ? 18.407
                                Long in degrees.minutes ? -104.578
                   Second Point : Lat in degrees.minutes ? 16.1
                                Long in degrees.minutes ? -99.418

     Interval  2  First point  : Lat in degrees.minutes ? 16.23
                                Long in degrees.minutes ? -99.215
                   Second Point : Lat in degrees.minutes ? 15.131
                                Long in degrees.minutes ? -95.488

     Interval  3  First point  : Lat in degrees.minutes ? 15.004
                                Long in degrees.minutes ? -95.7
                   Second Point : Lat in degrees.minutes ? 11.333
                                Long in degrees.minutes ? -88.1
```

```
Interval  4  First point  : Lat in degrees.minutes ? 10.503
                           Long in degrees.minutes ? -87.134
             Second Point : Lat in degrees.minutes ? 8.466
                           Long in degrees.minutes ? -84.366

Interval  5  First point  : Lat in degrees.minutes ? 7.134
                           Long in degrees.minutes ? -82.292
             Second Point : Lat in degrees.minutes ? 6.529
                           Long in degrees.minutes ? -79.503

★

   PLEASE, WAIT ...

Ready to print the limits table (Y/N) ? Y

Ready to print the results :
      Mean Depth and Standard Deviation (Y/N) ? Y
```

a

b

C
a

MIDDLE AMERICA TRENCH AXIS

Limits table

Section No.	Latitude in degrees-minutes	Longitude in degrees-minutes	Latitude in decimal degrees	Longitude in decimal degrees	Distorg in km
1	18°40.7′	-104°57.8′	18.678°	-104.963°	14.73
	16°10.0′	-99°41.8′	16.167°	-99.697°	644.89
2	16°23.0′	-99°21.5′	16.383°	-99.358°	683.26
	15°13.1′	-95°48.8′	15.218°	-95.813°	1077.44
3	15° 0.4′	-95°20.0′	15.007°	-95.333°	1133.91
	11°33.3′	-88°10.0′	11.555°	-88.167°	2000.39
4	10°50.3′	-87°13.4′	10.838°	-87.223°	2130.40
	8°46.6′	-84°36.6′	8.777°	-84.610°	2501.58
5	7°13.4′	-82°29.2′	7.223°	-82.487°	2810.27
	6°52.9′	-79°50.3′	6.882°	-79.838°	3110.73

b

MIDDLE AMERICA TRENCH AXIS

Results

Section No.	Mean Depth in m	Standard Dev. in m	Min. Depth in m	Max. Depth in m
1	4885	396	4489	5281
2	5181	212	4969	5392
3	6108	317	5791	6424
4	4301	630	3671	4931
5	2897	678	2219	3574

6.2

A

```
10 REM ================================================================
20 REM
30 REM                          PROGRAM  RADEPTH'
40 REM
50 REM                                        JACQUELINE ROUMP, SEPT. 1984
60 REM ================================================================
70 REM
80 PAGE
90 INIT
100 REM
110 REM     PRESENTATION
120 REM     ============
130 CHARSIZE 4
140 PRINT
150 PRINT "DEPTH VERSUS RADIUS"
160 PRINT
170 PRINT
180 CHARSIZE 3
190 PRINT "PROGRAM TO DRAW THE MEAN DEPTH OF SECTIONS THE MIDDLE"
200 PRINT "AMERICA TRENCH VERSUS THE RADIUS OF THESE CIRCLE SECTIONS"
210 PRINT
220 PRINT "THE DATA HAVE BEEN COMPUTED ON A Z100 COMPUTER IN PROGRAMS"
230 PRINT " DISSMCRC' AND  DIRSMCRC' FOR THE RADIUS VALUES, AND IN"
240 PRINT "PROGRAM  MEANDEPT' FOR THE MEAN DEPTHS AND THEIR STANDARD "
250 PRINT "DEVIATIONS."
260 PRINT
270 PRINT "IF YOU WANT A COLORED PLOT"
280 PRINT " PUT A BLACK PEN (FOR AXES, COMMENTS) INTO THE PEN HOLDER #0"
290 PRINT "     A RED PEN (FOR CROSSES) INTO THE PEN HOLDER #1"
300 PRINT "     YOU WILL HAVE TO CHANGE THE BLUE PEN OF PEN HOLDER #2"
310 PRINT "              BY A GREEN PEN LATER TO DRAW LINES"
320 PRINT
330 PRINT "IF YOU WANT A PLOT IN BLACK, PUT 3 BLACK PENS"
340 PRINT "                                 INTO THE 3 PEN HOLDERS"
350 REM ----------------------------------------------------------------
360 REM
370 REM     DATA
380 REM     ====
390 REM
400 REM FOR EACH POINT, MEAN DEPTH, MEAN DEPTH - STANDARD DEVIATION,
410 REM    MEAN DEPTH + STANDARD DEVIATION, RADIUS OF DIRECTION CIRCLE,
420 REM    RADIUS OF AZIMUTH CIRCLE
430 REM
440 DATA 4885,4489,5281,1639,1781
450 DATA 5181,4969,5392,784,1103
460 DATA 6108,5791,6424,3224,3177
470 DATA 4301,3671,4931,679,925
480 DATA 2897,2219,3574,277,396
490 DATA 4300,3800,4800,535,819
500 REM
510 REM   CREATING AND READING THE DATA TABLE
520 REM
530 DIM A(6,5)
540 READ A
550 REM
560 REM   CORRECTED MEAN DEPTH FOR THE PANAMA FZ TO 80W SECTION
570 REM
580 A5=3800
590 REM ----------------------------------------------------------------
600 REM
610 REM     DRAWING PART
620 REM     ============
630 REM
640 REM    PREPARING THE PLOTTER
650 REM    ---------------------
660 PRINT "PLACE THE PEN AT THE RIGHT END OF THE PLOTTER"
670 PRINT
680 PRINT "PLOTTER READY (Y/N) ? ";
690 INPUT R$
700 R$=SEG(R$,1,1)
710 IF R$<>"Y" THEN 680
720 PRINT
730 GOSUB 5310
740 REM
750 REM    DRAWING THE FRAME
760 REM    -----------------
770 Z0=21
780 Z1=0
790 Z2=1
800 GOSUB 5320
810 Z1=29.7
820 GOSUB 5320
830 Z0=0
840 GOSUB 5320
850 Z1=0
860 GOSUB 5320
870 REM
880 REM    TITLE
890 REM    -----
900 REM   SETTING CHARACTER TYPE AND ORIENTATON AND COLOR
910 REM
920 Z4=0.3
930 Z5=0.35
940 Z9=0
950 GOSUB 5340
960 REM
970 Z6=1
980 GOSUB 5350
990 REM
1000 REM   PRINTING THE TITLE TWICE
1010 REM
1020 Z$="MIDDLE AMERICA TRENCH"
1030 Z3=21
1040 Z2=0
1050 Z1=27
1060 Z0=11-Z3/2*Z4
1070 GOSUB 5320
1080 GOSUB 5330
1090 Z0=Z0+0.02
1100 GOSUB 5320
1110 GOSUB 5330
1120 REM
1130 REM   CHANGING THE COLOR
1140 REM
1150 Z6=0
1160 GOSUB 5350
1170 REM
1180 REM    VERTICAL AXIS
1190 REM    -------------
1200 REM   SETTING THE NEW CHARACTER TYPE
```

```
1210 REM
1220 Z4=0.2
1230 Z5=0.2
1240 GOSUB 5340
1250 REM
1260 Z7=-1
1270 REM
1280 REM   GOING TO THE ORIGIN
1290 REM
1300 Z0=4
1310 Z1=11.5
1320 GOSUB 5320
1330 REM
1340 REM   SETTING PARAMETERS
1350 REM
1360 L=12
1370 G0=(6500-2000)/500/L
1380 J1=(6500-2000)/500
1390 Z(15)=1
1400 REM
1410 REM   DRAWING THE AXIS AND TICK MARKS, AND PRINTING THE LABELS
1420 REM
1430 FOR K=1 TO J1+1
1440 Z1=11.5+(K-1)/G0
1450 Z2=1
1460 GOSUB 5320
1470 Z0=4-0.2
1480 GOSUB 5320
1490 IF K-1-INT((K-1)/Z(15))*Z(15)<>0 THEN 1560
1500 Z8=2000+(K-1)*500
1510 Z$=STR(Z8)
1520 Z0=Z0-0.2-(LEN(Z$)-0.5)*Z4
1530 Z2=0
1540 GOSUB 5320
1550 GOSUB 5360
1560 Z0=4
1570 GOSUB 5320
1580 NEXT K
1590 REM
1600 REM   DRAWING THE END OF THE AXIS
1610 REM
1620 Z2=1
1630 Z1=11.5+L
1640 GOSUB 5320
1650 REM
1660 REM   PRINTING THE TITLE AND UNIT
1670 REM
1680 Z2=0
1690 Z$="DEPTH"
1700 Z3=5
1710 Z1=11.5+L+1
1720 Z0=4-Z3/2*Z4
1730 GOSUB 5320
1740 GOSUB 5330
1750 Z$="M"
1760 Z3=1
1770 Z1=11.5+L+0.5
1780 Z0=4-Z3/2*Z4
1790 GOSUB 5320
1800 GOSUB 5330
1810 REM
1820 REM   HORIZONTAL AXIS
1830 REM   ---------------
1840 REM   GOING TO THE ORIGIN OF THE AXIS
1850 REM
1860 Z0=4
1870 Z1=11.5
1880 GOSUB 5320
1890 REM
1900 REM   SETTING PARAMETERS
1910 REM
1920 L=15
1930 G0=3500/500/15
1940 J1=3500/500
1950 Z(15)=1
1960 REM
1970 REM DRAWING THE AXIS AND TICK MARKS, AND PRINTING THE LABELS
1980 REM
1990 FOR K=1 TO J1+1
2000 Z2=1
2010 Z0=4+(K-1)/G0
2020 GOSUB 5320
2030 Z1=11.5-0.2
2040 GOSUB 5320
2050 IF K-1-INT((K-1)/Z(15))*Z(15)<>0 THEN 2130
2060 Z8=(K-1)*500
2070 Z$=STR(Z8)
2080 Z1=Z1-0.2-1.5*Z5
2090 Z0=Z0-(LEN(Z$)/2-0.5)*Z4
2100 Z2=0
2110 GOSUB 5320
2120 GOSUB 5360
2130 Z1=11.5
2140 Z0=4+(K-1)/G0
2150 GOSUB 5320
2160 NEXT K
2170 REM
2180 REM   DRAWING THE END OF THE AXIS
2190 REM
2200 Z2=1
2210 Z0=4+L
2220 GOSUB 5320
2230 REM
2240 REM   PRINTING THE TITLE AND UNIT
2250 REM
2260 Z2=0
2270 Z$="RADIUS"
2280 Z3=6
2290 Z1=11.5-0.2-5.5*Z5
2300 Z0=4+L-Z3*Z4
2310 GOSUB 5320
2320 GOSUB 5330
2330 Z$="KM"
2340 Z3=2
2350 Z1=11.5-0.2-7.5*Z5
2360 Z0=4+L-4*Z4
2370 GOSUB 5320
2380 GOSUB 5330
2390 REM
2400 REM   DATA DRAWING
```

```
2410 REM    ------------
2420 REM   SETTING NEW CHARACTER TYPE
2430 REM
2440 Z4=0.25
2450 Z5=0.25
2460 GOSUB 5340
2470 REM   SETTING PARAMETERS
2480 G0=3500/15
2490 G1=(6500-2000)/12
2500 REM
2510 REM   CHANGING THE PEN AND DRAWING THE CROSSES
2520 REM
2530 Z6=1
2540 GOSUB 5350
2550 FOR K=1 TO 6
2560 Z0=4+A(K,4)/G0
2570 Z1=11.5+(A(K,1)-2000)/G1
2580 GOSUB 5320
2590 Z2=1
2600 GOSUB 3270
2610 Z0=4+A(K,5)/G0
2620 GOSUB 5320
2630 GOSUB 3270
2640 Z2=0
2650 Z0=4+(A(K,4)+A(K,5))/2/G0
2660 Z1=11.5+(A(K,2)-2000)/G1
2670 GOSUB 5320
2680 Z2=1
2690 GOSUB 3340
2700 Z1=11.5+(A(K,3)-2000)/G1
2710 GOSUB 5320
2720 GOSUB 3340
2730 Z2=0
2740 NEXT K
2750 REM
2760 REM   CHANGING THE COLOR AND PRING THE LABELS
2770 REM
2780 Z6=0
2790 GOSUB 5350
2800 REM
2810 FOR K=1 TO 6
2820 Z0=4+(A(K,4)+A(K,5))/2/G0-0.7
2830 Z1=11.5+(A(K,1)-2000)/G1+0.3
2840 Z$=STR(K)
2850 Z3=LEN(Z$)
2860 GOSUB 5320
2870 GOSUB 5330
2880 NEXT K
2890 REM
2900 REM   DRAWING THE CORRECTED PANAMA FZ-80W CROSS
2910 REM
2920 REM   CHANGING THE PEN AND DRAWING THE CROSS
2930 REM
2940 Z6=1
2950 GOSUB 5350
2960 Z0=4+A(5,4)/G0
2970 Z1=11.5+(A5-2000)/G1
2980 GOSUB 5320
2990 Z2=1
3000 GOSUB 3270
3010 Z0=4+A(5,5)/G0
3020 GOSUB 5320
3030 GOSUB 3270
3040 Z2=0
3050 Z0=4+(A(5,4)+A(5,5))/2/G0
3060 Z1=11.5+(A5-2000)/G1-0.2
3070 GOSUB 5320
3080 Z2=1
3090 Z1=11.5+(A5-2000)/G1+0.2
3100 GOSUB 5320
3110 REM
3120 REM   CHANGING THE PEN AND PRINTING THE LABEL
3130 REM
3140 Z6=0
3150 GOSUB 5350
3160 Z2=0
3170 Z$="5"
3180 Z3=2
3190 Z0=Z0-0.4
3200 Z1=11.5+(A5-2000)/G1+0.3
3210 GOSUB 5320
3220 GOSUB 5330
3230 GO TO 3420
3240 REM
3250 REM   SUBROUTINES TO DRAW THE END OF THE CROSS ARMS
3260 REM
3270 Z1=Z1-0.12
3280 GOSUB 5320
3290 Z1=Z1+0.24
3300 GOSUB 5320
3310 Z1=Z1-0.12
3320 GOSUB 5320
3330 RETURN
3340 REM
3350 Z0=Z0-0.12
3360 GOSUB 5320
3370 Z0=Z0+0.24
3380 GOSUB 5320
3390 Z0=Z0-0.12
3400 GOSUB 5320
3410 RETURN
3420 REM
3430 REM    BEST FITTING LINES DRAWING AND LABELLING
3440 REM    ----------------------------------------
3450 REM   SETTING NEW CHARACTER TYPE
3460 REM
3470 Z4=0.3
3480 Z5=0.3
3490 GOSUB 5340
3500 REM
3510 REM   FIRST LINE
3520 REM
3530 Z6=2
3540 GOSUB 5350
3550 Z$="A"
3560 Z3=1
3570 Z0=4+0.1
3580 Z1=11.5+(3652-2000)/G1+0.2
3590 GOSUB 5320
3600 GOSUB 5330
```

```
3610 Z0=4
3620 Z1=11.5+(3652-2000)/G1
3630 GOSUB 5320
3640 Z2=1
3650 Z0=4+15
3660 Z1=11.5+(6324-2000)/G1
3670 GOSUB 5320
3680 Z2=0
3690 Z0=4+15-0.3
3700 Z1=11.5+(6324-2000)/G1-0.45
3710 GOSUB 5320
3720 GOSUB 5330
3730 REM
3740 REM   SECOND LINE
3750 REM
3760 A$=CHR(7)
3770 PRINT A$
3780 PRINT "PUT THE PEN WITH THE RIGHT COLOR IN PEN HOLDER #2"
3790 INPUT A$
3800 Z$="B"
3810 Z0=4+0.4
3820 Z1=11.5+(3651-2000)/G1+0.35
3830 GOSUB 5320
3840 GOSUB 5330
3850 Z0=4
3860 Z1=11.5+(3651-2000)/G1
3870 GOSUB 5320
3880 Z2=1
3890 Z0=4+15
3900 Z1=11.5+(6311-2000)/G1
3910 GOSUB 5320
3920 Z2=0
3930 Z0=4+15-0.6
3940 Z1=11.5+(6311-2000)/G1-0.75
3950 GOSUB 5320
3960 GOSUB 5330
3970 REM
3980 REM   THIRD LINE
3990 REM
4000 Z6=1
4010 GOSUB 5350
4020 Z$="C"
4030 Z0=4+0.01
4040 Z1=11.5+(3584.5-2000)/G1-0.35
4050 GOSUB 5320
4060 GOSUB 5330
4070 Z0=4
4080 Z1=11.5+(3584.5-2000)/G1
4090 GOSUB 5320
4100 Z2=1
4110 Z0=4+15
4120 Z1=11.5+(6336-2000)/G1
4130 GOSUB 5320
4140 Z2=0
4150 Z0=4+15-0.35
4160 Z1=11.5+(6336-2000)/G1+0.1
4170 GOSUB 5320
4180 GOSUB 5330
4190 REM
4200 REM    TEXT PRINTING
4210 REM    -------------
4220 REM   GOING TO THE FIRST LETTER POSITION
4230 REM
4240 Z0=4
4250 Z1=9.9
4260 GOSUB 5320
4270 REM
4280 REM   SETTING NEW CHARACTER TYPE AND COLOR
4290 REM
4300 Z4=0.22
4310 Z5=0.22
4320 GOSUB 5340
4330 REM
4340 Z6=0
4350 GOSUB 5350
4360 Z$="1  : NORTHERN SET"
4370 GOSUB 5190
4380 Z$="2  : NORTHERN MIDDLE SET"
4390 Z1=Z1-0.45
4400 GOSUB 5190
4410 Z$="3  : SOUTHERN MIDDLE SET"
4420 Z1=Z1-0.45
4430 GOSUB 5190
4440 Z$="4  : SOUTHERN SET"
4450 Z1=Z1-0.45
4460 GOSUB 5190
4470 Z$="5  : PANAMA FZ TO 80W"
4480 Z1=Z1-0.45
4490 GOSUB 5190
4500 Z$="5' : PANAMA FZ TO 80W (CORRECTED)"
4510 Z1=Z1-0.45
4520 GOSUB 5190
4530 Z$="6  : RIVERA FZ (DEPTH ESTIMATED)"
4540 Z1=Z1-0.45
4550 GOSUB 5190
4560 REM
4570 REM   PRINTING OF LEAST SQUARE LINES EQUATIONS AND CORRELATION
4580 REM
4590 REM   SETTING NEW CHARACTER TYPE AND COLOR
4600 REM
4610 Z4=0.26
4620 Z5=0.26
4630 GOSUB 5340
4640 Z6=2
4650 GOSUB 5350
4660 Z$="  LEAST SQUARE FITTING LINES"
4670 Z1=Z1-0.8
4680 GOSUB 5190
4690 REM
4700 REM   SETTING NEW CHARACTER TYPE AND COLOR
4710 REM
4720 Z6=0
4730 GOSUB 5350
4740 Z4=0.22
4750 Z5=0.22
4760 GOSUB 5340
4770 Z$="LINE AA  :  POINTS  1, 3, 4, 5',6"
4780 Z1=Z1-0.6
4790 GOSUB 5190
4800 Z$="    RADIUS   = 1.295 * DEPTH  - 4714"
```

```
4810 Z1=Z1-0.45
4820 GOSUB 5190
4830 Z$="          KM                M       KM"
4840 Z1=Z1-0.25
4850 GOSUB 5190
4860 Z$="    CORRELATION COEFFICIENT Rt2 = 0.9886"
4870 Z1=Z1-0.45
4880 GOSUB 5190
4890 REM
4900 Z$="LINE BB  :  POINTS 1, 3, 4"
4910 Z1=Z1-0.6
4920 GOSUB 5190
4930 Z$="    RADIUS   = 1.310 * DEPTH  - 4776"
4940 Z1=Z1-0.45
4950 GOSUB 5190
4960 Z$="          KM                M       KM"
4970 Z1=Z1-0.25
4980 GOSUB 5190
4990 Z$="    CORRELATION COEFFICIENT Rt2 = 0.9961"
5000 Z1=Z1-0.45
5010 GOSUB 5190
5020 REM
5030 Z$="LINE CC  :  POINTS  1, 3, 4, 5'"
5040 Z1=Z1-0.6
5050 GOSUB 5190
5060 Z$="    RADIUS   = 1.268 * DEPTH  - 4535"
5070 Z1=Z1-0.45
5080 GOSUB 5190
5090 Z$="          KM                M       KM"
5100 Z1=Z1-0.25
5110 GOSUB 5190
5120 Z$="    CORRELATION COEFFICIENT Rt2 = 0.9960"
5130 Z1=Z1-0.45
5140 GOSUB 5190
5150 Z0=24
5160 Z1=0
5170 GOSUB 5320
5180 END
5190 REM
5200 REM   PRINTING TEXT SUBROUTINE
5210 REM   ------------------------
5220 Z3=LEN(Z$)
5230 GOSUB 5320
5240 GOSUB 5330
5250 RETURN
5260 REM ------------------------------------------------------------------
5270 REM
5280 REM     BENSON PLOTTER SUBROUTINES
5290 REM     ==========================
5300 REM
5310 GO TO 5380
5320 GO TO 5620
5330 GO TO 5910
5340 GO TO 6010
5350 GO TO 6150
5360 GO TO 6240
5370 REM
5380 REM    INITIALIZATION OF THE BENSON PLOTTER
5390 REM    ------------------------------------
5400 DIM Z(25),Z$(80)
5410 ON SRQ THEN 5600
5420 PRINT "BENSON ADRESS ? ";
5430 INPUT Z(25)
5440 PRINT @Z(25),"76 ";
5450 Z$=CHR(34)
5460 PRINT @Z(25),Z$,"#$%&'()*+,-./0123456789:;<=>?";
5470 PRINT @Z(25),"@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]↑←";
5480 PRINT @Z(25), USING "A/L","  "
5490 PRINT @Z(25),"76                                                  ";
5500 PRINT @Z(25),"                                                    ";
5510 PRINT @Z(25), USING "A/L","  "
5520 Z(1)=0
5530 Z(2)=0
5540 Z(3)=200
5550 Z(5)=60
5560 Z(6)=0
5570 Z(13)=0
5580 Z(20)=Z(3)
5590 Z(21)=Z(3)
5600 RETURN
5610 REM
5620 REM    MOVING THE PEN
5630 REM    --------------
5640 Z(7)=INT(Z0*Z(20))
5650 Z(8)=INT(Z1*Z(21))
5660 Z(9)=Z(7)-Z(1)
5670 Z(10)=Z(8)-Z(2)
5680 Z(1)=Z(7)
5690 Z(2)=Z(8)
5700 Z(7)=ABS(Z(9))
5710 Z(8)=ABS(Z(10))
5720 IF Z(7)>16383 THEN 5810
5730 IF Z(8)>16383 THEN 5810
5740 Z(7)=0
5750 Z(11)=Z(9)
5760 Z(12)=Z(10)
5770 PRINT @Z(25),"PL ",Z(11),Z(12),Z2,
5780 PRINT @Z(25), USING "A/L","  "
5790 IF Z(7)>1 THEN 5700
5800 RETURN
5810 Z(7)=INT((Z(7)+16383)/16384)
5820 Z(8)=INT((Z(8)+16383)/16384)
5830 IF Z(7)>Z(8) THEN 5850
5840 Z(7)=Z(8)
5850 Z(11)=INT(Z(9)/Z(7))
5860 Z(12)=INT(Z(10)/Z(7))
5870 Z(9)=Z(9)-Z(11)
5880 Z(10)=Z(10)-Z(12)
5890 GO TO 5770
5900 REM
5910 REM    WRITING A STRING OF CHARACTERS
5920 REM    ------------------------------
5930 PRINT @Z(25),"TX ",Z3," ",Z$,
5940 PRINT @Z(25), USING "A/L","  "
5950 Z(1)=Z(1)+Z(5)*Z3
5960 Z(2)=Z(2)+Z(6)*Z3
5970 PRINT @Z(25),"NT ",Z(5)*Z3," ",Z(6)*Z3,
5980 PRINT @Z(25), USING "A/L","  "
5990 RETURN
6000 REM
```

```
6010 REM    INITIALIZATION OF CHARACTER TYPE AND ORIENTATION
6020 REM    ------------------------------------------------
6030 Z(5)=Z4*COS(Z9)
6040 Z(6)=Z4*SIN(Z9)
6050 Z(16)=INT(4*Z(5)*Z(20)/3)
6060 Z(17)=INT(-4*Z5*Z(20)*SIN(Z9)/3)
6070 Z(18)=INT(4*Z(6)*Z(21)/3)
6080 Z(19)=INT(4*Z5*Z(21)*COS(Z9)/3)
6090 PRINT @Z(25),"SZ ";Z(16);" ";Z(17);" ";Z(18);" ";Z(19);
6100 PRINT @Z(25), USING "A/L",";" "
6110 Z(5)=INT(Z(5)*Z(20))
6120 Z(6)=INT(Z(6)*Z(21))
6130 RETURN
6140 REM
6150 REM    CHANGING THE PEN
6160 REM    ----------------
6170 Z(14)=INT(ABS(Z6))-3*INT(INT(ABS(Z6))/3)
6180 IF Z(14)=Z(13) THEN 6220
6190 PRINT @Z(25),"NP ";Z(14),
6200 PRINT @Z(25), USING "A/L",";" "
6210 Z(13)=Z(14)
6220 RETURN
6230 REM
6240 REM    PRINTING OF NUMBERS
6250 REM    -------------------
6260 Z7=INT(Z7)
6270 Z(9)=ABS(INT(Z7))
6280 IF Z7>-2 THEN 6460
6290 IF Z(9)<10 THEN 6310
6300 Z(9)=9
6310 Z$=STR(Z(9))
6320 Z$=Z$&"E"
6330 Z(9)=Z(9)+7
6340 IF Z8>0 THEN 6380
6350 Z(9)=Z(9)+1
6360 PRINT @Z(25),"TX ";Z(9);" ",
6370 GO TO 6390
6380 PRINT @Z(25),"TX ";Z(9),
6390 PRINT @Z(25), USING Z$,Z8,
6400 PRINT @Z(25), USING "A/L",";" "
6410 Z(1)=Z(1)+Z(9)*Z(5)
6420 Z(2)=Z(2)+Z(9)*Z(6)
6430 PRINT @Z(25),"NT ";Z(9)*Z(5);" ";Z(9)*Z(6),
6440 PRINT @Z(25), USING "A/L",";" "
6450 RETURN
6460 Z(7)=INT(LGT(ABS(Z8)+0.5))+1
6470 IF Z(7)>0 THEN 6490
6480 Z(7)=1
6490 IF Z8=>0 THEN 6510
6500 Z(7)=Z(7)+1
6510 Z$=STR(Z(7))
6520 Z$=Z$&"D"
6530 IF Z7>-1 THEN 6560
6540 Z(9)=Z(7)
6550 GO TO 6360
6560 Z(8)=LEN(Z$)+2
6570 Z$=Z$&".D"
6580 IF Z(9)<10 THEN 6600
6590 Z(9)=9
6600 GO TO Z(9)+1 OF 6610,6630,6650,6670,6690,6710,6730,6750,6770,6790
6610 Z$=REP(" ",Z(8)+1,1)
6620 GO TO 6800
6630 Z$=REP("1",Z(8),1)
6640 GO TO 6800
6650 Z$=REP("2",Z(8),1)
6660 GO TO 6800
6670 Z$=REP("3",Z(8),1)
6680 GO TO 6800
6690 Z$=REP("4",Z(8),1)
6700 GO TO 6800
6710 Z$=REP("5",Z(8),1)
6720 GO TO 6800
6730 Z$=REP("6",Z(8),1)
6740 GO TO 6800
6750 Z$=REP("7",Z(8),1)
6760 GO TO 6800
6770 Z$=REP("8",Z(8),1)
6780 GO TO 6800
6790 Z$=REP("9",Z(8),1)
6800 Z(9)=Z(9)+Z(7)+1
6810 GO TO 6360
```

B

★

DEPTH VERSUS RADIUS

PROGRAM TO DRAW THE MEAN DEPTH OF SECTIONS THE MIDDLE AMERICA TRENCH VERSUS THE RADIUS OF THESE CIRCLE SECTIONS

THE DATA HAVE BEEN COMPUTED ON A Z100 COMPUTER IN PROGRAMS 'DISSMCRC' AND 'DIRSMCRC' FOR THE RADIUS VALUES, AND IN PROGRAM 'MEANDEPT' FOR THE MEAN DEPTHS AND THEIR STANDARD DEVIATIONS

IF YOU WANT A COLORED PLOT
PUT A BLACK PEN (FOR AXES, COMMENTS) INTO THE PEN HOLDER #0
A RED PEN (FOR CROSSES) INTO THE PEN HOLDER #1
YOU WILL HAVE TO CHANGE THE BLUE PEN OF PEN HOLDER #2
BY A GREEN PEN LATER TO DRAW LINES

IF YOU WANT A PLOT IN BLACK, PUT 3 BLACK PENS
INTO THE 3 PEN HOLDERS
PLACE THE PEN AT THE RIGHT END OF THE PLOTTER

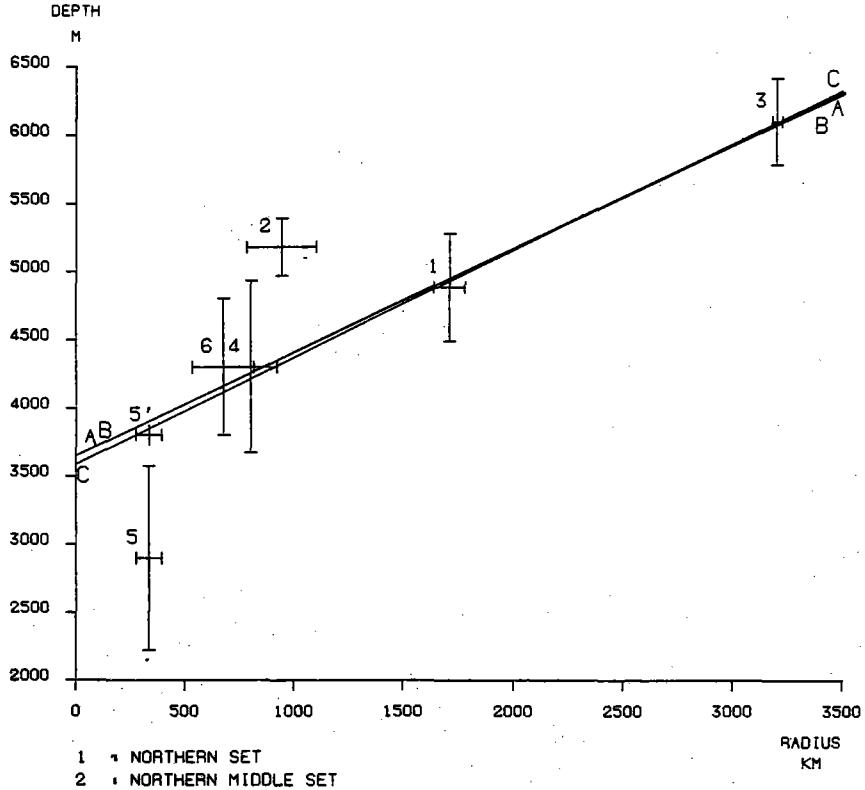
PLOTTER READY (Y/N) ? Y

BENSON ADRESS ? 41

PUT THE PEN WITH THE RIGHT COLOR IN PEN HOLDER #2

C

MIDDLE AMERICA TRENCH



1 : NORTHERN SET
2 : NORTHERN MIDDLE SET
3 : SOUTHERN MIDDLE SET
4 : SOUTHERN SET
5 : PANAMA FZ TO 80W
5' : PANAMA FZ TO 80W (CORRECTED)
6 : RIVERA FZ (DEPTH ESTIMATED)

LEAST SQUARE FITTING LINES

LINE AA : POINTS 1, 3, 4, 5', 6

$RADIUS_{KM} = 1.295 * DEPTH_{M} - 4714_{KM}$

CORRELATION COEFFICIENT R12 = 0.9886

LINE BB : POINTS 1, 3, 4

$RADIUS_{KM} = 1.310 * DEPTH_{M} - 4776_{KM}$

CORRELATION COEFFICIENT R12 = 0.9961

LINE CC : POINTS 1, 3, 4, 5'

$RADIUS_{KM} = 1.268 * DEPTH_{M} - 4535_{KM}$

CORRELATION COEFFICIENT R12 = 0.9960

POSITIONS DE POINTS EN FONCTION DE LA DISTANCE ET DE LA DIRECTION, SUR LA SPHERE, DISTANCE LE LONG DE DROITES, DE POINTS PROJETES

7.1 - CALCUL DE LA POSITION DE POINTS SUR LA SPHERE EN FONCTION DE LA DISTANCE ET DE LA DIRECTION

A - Exemple de fichier de données

B - Listing du programme 'POINTS'

C - Copie d'écran d'un exemple d'exécution du programme 'POINTS'

D - Exemple de sortie sur imprimante

7.2 - CALCUL DE LA DISTANCE LE LONG DE DROITES DE POINTS PROJETES

A - Listing du programmé 'PROPOINT'

B - Copie d'écran d'un exemple d'exécution du programme 'PROPOINT'

C - Exemple de sortie sur imprimante

7.1

A

```
500 ' ============================================================
510 '
520 '                          DATA `MATPNTSS.DAT'
530 '
540 '                                          Jacqueline ROUMP, Sept.1984
550 ' ============================================================
560 '
570 ' These data are for the program POINTS. The points are along the Middle
580 ' America Trench axis with directions parallel to mean oceanic fracture
590 ' zones direction, to be used for earthquake cross sections limits.
600 ' The data are the latitude and longitude of the point in decimal degrees,
610 ' then the distance to the west point, and then to the east point, the
620 ' last value is the direction of the line passing through the 3 points
630 ' clockwise from North.
640 '     This file must be saved in an ASCII format
650 '
660 N=28                                    'number of points
670 D$="D"                                  'distances are in degrees
680 '
690 DATA 18.68,-104.96,.5,5,43
700 DATA 18.37,-104.55,.8,5,43
710 DATA 17.77,-103.73,1.1,5.1,43
720 DATA 17.37,-102.78,1.4,5.2,43
730 DATA 16.96,-101.89,1.7,5.3,43
740 DATA 16.64,-101.19,1.7,5.4,43
750 DATA 16.34,-100.17,1.7,5.5,43
760 DATA 16.04,-99.36,1.7,5.6,43
770 DATA 15.77,-98.65,1.7,5.7,43
780 DATA 15.49,-97.79,1.7,5.8,43
790 DATA 15.29,-97.23,1.6,5.9,43
800 DATA 15.22,-95.81,1.5,6,43
810 DATA 15,-95.33,1.5,6,43
820 DATA 14.7,-94.84,1.6,5.8,43
830 DATA 14.17,-93.99,1.7,5.6,43
840 DATA 13.81,-93.32,1.8,5.4,43
850 DATA 13.48,-92.67,1.9,5.2,43
860 DATA 12.81,-91.24,2,5,43
870 DATA 12.5,-90.54,2,5,43
880 DATA 12.17,-89.58,2,5,43
890 DATA 11.78,-88.65,2,5,43
900 DATA 11.55,-88.17,2,5,43
910 DATA 10.83,-87.22,2,5,43
920 DATA 9.81,-86.33,2,5,43
930 DATA 9.15,-85.5,2,5,43
940 DATA 8.65,-84.42,2,5,43
950 DATA 8.35,-83.87,2,5,43
960 DATA 7.32,-82.83,2,5,43
```

B

```
10 ' ============================================================
20 '
30 '                          PROGRAM `POINTS'
40 '
50 '                                          Jacqueline ROUMP, Sept. 1984
60 ' ============================================================
70 '
80 CLS : CLEAR : OPTION BASE 1
90 '
100 '     PRESENTATION
110 '     ============
120 PRINT "Program to compute the positions of the points located on a line"
130 PRINT "passing through the given point : the first point will be W of the"
140 PRINT "given point and the second E of it, the line is given by its"
150 PRINT "direction clockwise from North ; the points on the line are"
160 PRINT "located by their distance from the given point."
170 PRINT
180 PRINT "Your data must have been written in another file, a data file"
190 PRINT "(extension of the filename =.DAT) that has been saved in an ASCII"
200 PRINT "format. The line number of the data file must go from 500 to less"
210 PRINT "than 1000. The first line of the file must read : `N=x : D$='"
220 PRINT "(x=number of given points) for D$ you must write K if the"
230 PRINT "distances are in km, M if they are in nautical milles or D if"
240 PRINT "they are in degrees"
250 PRINT
260 PRINT "Then the data must be organized this way :"
270 PRINT "    1st number : latitude of the point in decimal degrees"
280 PRINT "    2nd number : longitude of the point in decimal degrees"
290 PRINT "    3rd number : distance in km, nautical miles or degrees to the"
300 PRINT "                 W point, the unit is in variable D$"
310 PRINT "    4th number : distance in km, naut. miles or deg. to the E point"
320 PRINT "    5th number : angle of the line clockwise from the North in deg."
330 PRINT "For example : line 200 could be :   200 DATA 21.66,-107,.5,5,68"
340 PRINT
350 '
360 INPUT "DISK READY (Y/N) ";Y$
370     IF LEFT$(Y$,1)<>"Y" AND LEFT$(Y$,1)<>"y" THEN 360
380 PRINT
390 PRINT "Name of the data files on the disk :"
400 FILES"*.DAT"                            'list of the disk data files
410 PRINT
420   INPUT "Name of your data file : ";X$
430 CHAIN MERGE X$,500,ALL                  'merging the data file
440 '  ---------------------------------------------------------------------
450 '     PLACE FOR THE DATA FILE
1000 ' ---------------------------------------------------------------------
1010 '
1020 '     READING THE DATA  AND QUESTIONS
1030 '     ===============================
1040 DIM T(N,11)                            'creating the table T
1050 '
1060 '    Putting the data into the table
1070 '    -------------------------------
1080 '---------------
1090 FOR I=1 TO N
1100 '     ---------------
```

```
1110       FOR J=1 TO 5
1120          READ T(I,J)
1130       NEXT J
1140 '     ---------------
1150 NEXT I
1160 '---------------
1170 '
1180 '    Test to see if D$ is correct
1190 '    and to assign the unit label for the printing
1200 '    ---------------------------------------------
1210 D$=LEFT$(D$,1)
1220       IF D$="D" OR D$="d" THEN A$="deg" : GOTO 1520
1230       IF D$="K" OR D$="k" THEN A$=" km" : GOTO 1320
1240       IF D$="M" OR D$="m" THEN A$=" nm" : GOTO 1420
1250 '
1260 '  Asking for a new unit label
1270 '  ~~~~~~~~~~~~~~~~~~~~~~~~~~~
1280 PRINT "Unit of distance : K for Kilometers, M for nautical miles,"
1290   INPUT "or D for degrees (=60 nm)" ; D$
1300 GOTO 1210
1310 '
1320 '  Unit is kilometer
1330 '  ~~~~~~~~~~~~~~~~~
1340 '---------------
1350 FOR I=1 TO N
1360    T(I,10)=T(I,3)*90/10000
1370    T(I,11)=T(I,4)*90/10000
1380 NEXT I
1390 '---------------
1400 GOTO 1610
1410 '
1420 '  Unit is nautical mile
1430 '  ~~~~~~~~~~~~~~~~~~~~~
1440 '---------------
1450 FOR I=1 TO N
1460    T(I,10)=T(I,3)/60
1470    T(I,11)=T(I,4)/60
1480 NEXT I
1490 '---------------
1500 GOTO 1610
1510 '
1520 '  Unit is degree
1530 '  ~~~~~~~~~~~~~~
1540 '---------------
1550 FOR I=1 TO N
1560    T(I,10)=T(I,3)
1570    T(I,11)=T(I,4)
1580 NEXT I
1590 '---------------
1600 '
1610 PRINT
1620   INPUT "TITLE : ";T$
1630 '
1640 CLS                                        'clearing the screen
1650 LOCATE 15,5 : PRINT "COMPUTING ..."
1660 ' ------------------------------------------------------------------------
1670 '
```

```
1680 '     COMPUTATION PART
1690 '     ================
1700 '
1710 '  Definition of functions of spherical trigonometry of rectangle
1720 '  triangle to determine the latitude and longitude of the points
1730 '  ---------------------------------------------------------------
1740 FPI#=3.141592653589887#/180                    'PI/180
1750 DEF FNA(X,Y)=(SIN(X*FPI#)*SIN(Y*FPI#))         'sin of the delta latitude,
                              X = length of the side facing the right angle,
                                    Y = angle facing the side FNA
1760 DEF FNB(X#)=(ATN(X#/SQR(-X#*X#+1)))/FPI#       'arcsine
1770 DEF FNC(X,Y)=ATN(TAN(X*FPI#)*COS(Y*FPI#))/FPI#     'delta longitude,
                         X = length of the side facing the right angle,
                             Y = angle facing adjacent side to side FNA
1780 '
1790 '    Computing the positions of the points
1800 '    -------------------------------------
1810 '---------------
1820 FOR I=1 TO N
1830    K#=FNA(T(I,10),90-T(I,5))
1840       IF K#).9999999999999999# THEN T(I,6)=T(I,1)-90: GOTO 1860
1850    T(I,6)=T(I,1)-FNB(K#)
1860    K#=FNC(T(I,10),90-T(I,5))
1870    T(I,7)=T(I,2)-K#
1880    K#=FNA(T(I,11),90-T(I,5))
1890       IF K#).9999999999999999# THEN T(I,8)=T(I,1)+90: GOTO 1910
1900    T(I,8)=T(I,1)+FNB(K#)
1910    K#=FNC(T(I,11),90-T(I,5))
1920    T(I,9)=T(I,2)+K#
1930 NEXT I
1940 '---------------
1950 ' ------------------------------------------------------------------------
1960 '
1970 '     PRINTING PART
1980 '     =============
1990 BEEP : PRINT
2000   INPUT "PRINTER READY (Y/N)";R$
2010     . IF R$()"Y" AND R$()"y" THEN 2000
2020 '
2030 '    Preparing the printer
2040 '    ---------------------
2050 LPRINT CHR$(27)"n" : LPRINT CHR$(27)"L009" : LPRINT CHR$(27)"!"
2060 '
2070 '    Printing the title
2080 '    ------------------
2090 T=INT(72-9-LEN(T$))/2
2100 LPRINT TAB(T) CHR$(27)"X" T$
2110 '
2120 '    Printing the column headings
2130 '    ----------------------------
2140 LPRINT CHR$(27)"E" : LPRINT CHR$(27)"L012"
2150 LPRINT CHR$(27)"Y" : LPRINT CHR$(27)CHR$(34)
2160 LPRINT "No        POINT          D W      D E    Angle      Point  W
   Point  E"
2170 LPRINT"       lat     long       ";A$;"      ";A$;"     deg      lat     lo
ng       lat     long"
2180 LPRINT
2190 '
```

```
2200 '    Printing the data and results table
2210 '    ----------------------------------
2220 '---------------
2230 FOR I=1 TO N
2240    LPRINT USING "##  ";I;                 'point number
2250    LPRINT USING "###.### ";T(I,1);        'latitude of the original point
2260    LPRINT USING "####.### ";T(I,2);       'longitude of the original point
2270 '     ---------------
2280       FOR J=3 TO 4
2290          LPRINT USING "#####.## ";T(I,J);      'distance W, and E to the
2300       NEXT J                                   'searched points
2310 '     ---------------
2320    LPRINT USING "####.## ";T(I,5);        'angle of the 3 points line
2330 '     ---------------
2340       FOR J=6 TO 9
2350          LPRINT USING "####.### ";T(I,J);      'lat. and long. of the W
2360       NEXT J                                   'point, then the E point
2370 '     ---------------
2380    LPRINT
2390 NEXT I
2400 '---------------
2410 '
2420 END
```

C

★

Program to compute the positions of the points located on a line passing through the given point : the first point will be W of the given point and the second E of it, the line is given by its direction clockwise from North ; the points on the line are located by their distance from the given point.

Your data must have been written in another file, a data file (extension of the filename =.DAT) that has been saved in an ASCII format. The line number of the data file must go from 500 to less than 1000. The first line of the file must read : 'N=x : D$=' (x=number of given points) for D$ you must write K if the distances are in km, M if they are in nautical miles or D if they are in degrees

Then the data must be organized this way :

```
   1st number : latitude of the point in decimal degrees
   2nd number : longitude of the point in decimal degrees
   3rd number : distance in km, nautical miles or degrees to the
                W point, the unit is in variable D$
   4th number : distance in km, naut. miles or deg. to the E point
   5th number : angle of the line clockwise from the North in deg.
For example : line 600 could be :   600 DATA 21.66,-107,,5,5,68

DISK READY (Y/N) ? Y

Name of the data files on the disk :
ERSQUAKE.DAT MATPNTSS.DAT MATPNTFZ.DAT MATPNTTA.DAT MATPNTRM.DAT RIVPNTSS.DAT
RIVPNTPL.DAT DISCMAT3.DAT DISCMAT2.DAT MATPNTS1.DAT DISCMAT4.DAT DISCMAT1.DAT
MATPNTPL.DAT RIVPOINT.DAT

Name of your data file : ? MATPNTSS.DAT

TITLE : ? MIDDLE AMERICA TRENCH : EARTHQUAKES SECTIONS LIMITS
```

★

```
    COMPUTING ...

PRINTER READY (Y/N)? Y
```

D

MIDDLE AMERICA TRENCH : EARTHQUAKES SECTIONS LIMITS

No	POINT		D W	D E	Angle	Point W		Point E	
	lat	long	deg	deg	deg	lat	long	lat	long
1	18.680	-104.960	0.50	5.00	43.00	18.314	-105.301	22.335	-101.545
2	18.370	-104.550	0.80	5.00	43.00	17.785	-105.096	22.025	-101.135
3	17.770	-103.730	1.10	5.10	43.00	16.966	-104.480	21.498	-100.247
4	17.370	-102.780	1.40	5.20	43.00	16.346	-103.735	21.171	-99.228
5	16.960	-101.890	1.70	5.30	43.00	15.717	-103.050	20.834	-98.270
6	16.640	-101.190	1.70	5.40	43.00	15.397	-102.350	20.587	-97.501
7	16.340	-100.170	1.70	5.50	43.00	15.097	-101.330	20.360	-96.413
8	16.040	-99.360	1.70	5.60	43.00	14.797	-100.520	20.133	-95.534
9	15.770	-98.650	1.70	5.70	43.00	14.527	-99.810	19.936	-94.756
10	15.490	-97.790	1.70	5.80	43.00	14.247	-98.950	19.728	-93.827
11	15.290	-97.230	1.60	5.90	43.00	14.120	-98.321	19.601	-93.199
12	15.220	-95.810	1.50	6.00	43.00	14.123	-96.833	19.604	-91.710
13	15.000	-95.330	1.50	6.00	43.00	13.903	-96.353	19.384	-91.230
14	14.700	-94.840	1.60	5.80	43.00	13.530	-95.931	18.938	-90.877
15	14.170	-93.990	1.70	5.60	43.00	12.927	-95.150	18.263	-90.164
16	13.810	-93.320	1.80	5.40	43.00	12.494	-94.548	17.757	-89.631
17	13.480	-92.670	1.90	5.20	43.00	12.091	-93.966	17.281	-89.118
18	12.810	-91.240	2.00	5.00	43.00	11.347	-92.604	16.465	-87.825
19	12.500	-90.540	2.00	5.00	43.00	11.037	-91.904	16.155	-87.125
20	12.170	-89.580	2.00	5.00	43.00	10.707	-90.944	15.825	-86.165
21	11.780	-88.650	2.00	5.00	43.00	10.317	-90.014	15.435	-85.235
22	11.550	-88.170	2.00	5.00	43.00	10.087	-89.534	15.205	-84.755
23	10.830	-87.220	2.00	5.00	43.00	9.367	-88.584	14.485	-83.805
24	9.810	-86.330	2.00	5.00	43.00	8.347	-87.694	13.465	-82.915
25	9.150	-85.500	2.00	5.00	43.00	7.687	-86.864	12.805	-82.085
26	8.650	-84.420	2.00	5.00	43.00	7.187	-85.784	12.305	-81.005
27	8.350	-83.870	2.00	5.00	43.00	6.887	-85.234	12.005	-80.455
28	7.320	-82.830	2.00	5.00	43.00	5.857	-84.194	10.975	-79.415

7.2

A

```
10 ' ═══════════════════════════════════════════════════════════════
20 '
30 '                          PROGRAM 'PROPOINT'
40 '
50 '                                          Jacqueline ROUMP, Sept. 1984
60 ' ═══════════════════════════════════════════════════════════════
70 '
80 CLS : CLEAR : OPTION BASE 1
90 '
100 '     PRESENTATION AND QUESTIONS
110 '     ══════════════════════════
120 PRINT "This program computes the distances in Km along projection lines"
130 PRINT "(up to 5) of up to 10 points. These points may be the corners of"
140 PRINT "polygons in which earthquakes are sorted to draw cross-sections"
150 PRINT "(see SEISSORT)."
160 PRINT
170 '
180   INPUT "TITLE : ";TITLE$
190 PRINT
200   INPUT "NUMBER OF SIDES of the polygon (up to 10) : ";NS
210       IF NS>10 THEN PRINT "Only up to 10 sides" : GOTO 200
220 '
230 '    Latitude and longitude of each corner of the polygon
240 '    ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
250 '---------------
260 FOR I=1 TO NS
270    PRINT "       Corner ";I;" lat ";
280       INPUT B(I,1)
290       INPUT "                  long "; B(I,2)
300 NEXT I
310 '---------------
320 PRINT
330 '
340   INPUT "How many lines of projection (up to 5) : ";LP
350       IF LP>5 THEN PRINT "Only 5 different lines of projection" : GOTO 340
360 '
370 '    Latitude and longitude of the two points necessary for the definition
380 '    of the projection line
390 '    ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
400 PRINT
410 '---------------
420 FOR I=1 TO LP
430    PRINT "LINE OF PROJECTION ";I
440       INPUT "     Name (up to 4 letters)  ";LP$(I) : LP$(I)=LEFT$(LP$(I),4)
450       INPUT "     1st point (origin) : lat ";LAT(I)
460       INPUT "                          long ";LONG(I)
470       INPUT "     2nd point          : lat ";LAT(I+5)
480       INPUT "                          long ";LONG(I+5)
490    PRINT
500 NEXT I
510 '---------------
520 CLS : LOCATE 15,5 : PRINT "COMPUTING ..."
530 ' ------------------------------------------------------------------
540 '
550 '     COMPUTATION PART
560 '     ================
570 '    WGS72 ellipsoid parameters and definition of functions
580 '    for the transformation of latitudes into Y positions
590 '    ---------------------------------------------------
600 B#=SQR(.00666943178#)
610 FPI#=3.1415926539#/180
620 DEF FNX(X)=(LOG(TAN((45+X/2)*FPI#))-B#/2*LOG((1+B#*SIN(X*FPI#))/(1-B#*SIN(
X*FPI#))))/FPI#
630 '
640 '    Transformation of the latitude of the corners to be projected,
650 '    into Y position
660 '    ------------------------------------------------------------
670 '---------------
680 FOR I=1 TO NS
690    B(I,3)=FNX(B(I,1))
700 NEXT I
710 '---------------
720 '
730 '    Search for the maximum and minimum Y positions and longitudes
740 '    of the corners
750 '    ------------------------------------------------------------
760 TMAX=B(1,1) : TMIN=TMAX
770 GMAX=B(1,2) : GMIN=GMAX
780 LMAX=B(1,3) : LMIN=LMAX
790 '---------------
800 FOR I=2 TO NS
810       IF B(I,3)>LMAX THEN LMAX=B(I,3) : TMAX=B(I,1)
820       IF B(I,3)<LMIN THEN LMIN=B(I,3) : TMIN=B(I,1)
830       IF B(I,2)>GMAX THEN GMAX=B(I,2)
840       IF B(I,2)<GMIN THEN GMIN=B(I,2)
850 NEXT I
860 '---------------
870 '
880 '    Definition of the parameters of the lines bearing the sides
890 '    of the polygon
900 '    ------------------------------------------------------------
910 '  from the first to the last point
920 '  ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
930 '---------------
940 FOR I=1 TO NS-1
950       IF B(I,2)=B(I+1,2) THEN W(I)=1 : GOTO 1000
960       IF B(I,2)>B(I+1,2) THEN X1(I)=B(I,2) : X2(I)=B(I+1,2) : GOTO 980
970    X1(I)=B(I+1,2) : X2(I)=B(I,2)
980    C(I)=(B(I,3)-B(I+1,3))/(B(I,2)-B(I+1,2))     'a of y=a*x+b
990    D(I)=B(I,3)-B(I,2)*C(I)                      'b of y=a*x+b
1000 NEXT I
1010 '---------------
1020 '
1030 '  from the last to the first point
1040 '  ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
1050       IF B(1,2)=B(NS,2) THEN W(NS)=1 : GOTO 1110
1060       IF B(NS,2)>B(1,2) THEN X1(NS)=B(NS,2) : X2(NS)=B(1,2) : GOTO 1080
1070 X1(NS)=B(1,2) : X2(NS)=B(NS,2)
1080 C(NS)=(B(NS,3)-B(1,3))/(B(NS,2)-B(1,2))
1090 D(NS)=B(NS,3)-B(NS,2)*C(NS)
1100 '
1110 '    Definition of functions to compute the distance between 2 points
1120 '    ---------------------------------------------------------------
1130 CT#=6378.135#FPI#           'number of Km per degree of long
1140 F#=.99333056822#            'correction factor for lat (WGS 72 ellipsoid)
```

```
1150 DEF FNA(X)=ATN(F#*TAN(X*FPI#))/FPI#   'corrected lat in degrees
1160 DEF FNB(X,Y,Z,W)=SIN(X*FPI#)*SIN(Z*FPI#)+COS(X*FPI#)*COS(Z*FPI#)*COS((W-Y
)*FPI#)
1170 DEF FNC(X)=(-ATN(X/SQR(-X^2+1))+1.5708)/FPI#          'arc cosine
1180 '
1190 '    Building up the table containing the Y positions of the 2 projection
1200 '    points and the distance between these points
1210 '    ------------------------------------------------------------------
1220 DIM L(15)
1230 '---------------
1240 FOR I=1 TO LP
1250    L(I)=FNA(LAT(I))
1260    L(I+5)=FNA(LAT(I+5))
1270    L(I+10)=FNC(FNB(L(I),LONG(I),L(I+5),LONG(I+5)))
1280 NEXT I
1290 '---------------
1300 '
1310 '    Distance between the point from table A and the 2 points of the
1320 '    line of projection (L1 and L7), angle between the line and the
1330 '    point (L5), distance along the line from the origin to the
1340 '    projection in degrees (L2), and in km (B(I,5 to 10))
1350 '    ------------------------------------------------------------------
1360 '---------------
1370 FOR I=1 TO NS
1380    L3=FNA(B(I,1))
1390 '     ---------------
1400       FOR J=1 TO LP
1410          L4=FNB(L(J),LONG(J),L3,B(I,2))
1420               IF L4=1 THEN B(I,J+2)=0 : GOTO 1530
1430          L1=FNC(L4)
1440          L6=FNB(L(J+5),LONG(J+5),L3,B(I,2))
1450               IF L6=1 THEN L2=L(J+10) : GOTO 1520
1460          L7=FNC(L6)
1470          D#=(COS(L7*FPI#)-COS(L1*FPI#)*COS(L(J+10)*FPI#))/(SIN(L1*FPI#)*S
IN(L(J+10)*FPI#))
1480               IF ABS(D#)).9999999999999999# THEN L5=0 : GOTO 1500
1490          L5=FNC(D#)
1500          L2=ATN(TAN(L1*FPI#)*COS(L5*FPI#))/FPI#
1510               IF L5=0 AND D#<0 THEN L2=-L2
1520          B(I,J+2)=L2*CT#
1530       NEXT J
1540 '     ---------------
1550 NEXT I
1560 '---------------
1570 '-----------------------------------------------------------------------
1580 '
1590 '     PRINTING PART
1600 '     =============
1610 PRINT : BEEP
1620   INPUT "PRINTER READY (Y/N) ";R$
1630       IF LEFT$(R$,1)<>"Y" AND LEFT$(R$,1)<>"y" THEN 1610
1640 '
1650 '    Preparing the printer and printing the title
1660 '    --------------------------------------------
1670 LPRINT CHR$(27)"n"CHR$(27)"!"CHR$(27)"L009"
1680 LPRINT TAB((60-9-4-LEN(TITLE$))/2) CHR$(27)"X" TITLE$
1690 '
1700 '    Printing the projection line two points latitude and longitude
1710 '    --------------------------------------------------------------
1720 LPRINT CHR$(27)"q" : LPRINT CHR$(27)"Y"
1730 LPRINT CHR$(27)"L012"CHR$(27)CHR$(34)
1740 LPRINT "PROJECTION LINE";
1750 '---------------
1760 FOR I=1 TO LP
1770    LPRINT SPC(5+(LEN(LP$(I)))/2);LP$(I);SPC(6+LEN(LP$(I))-(LEN(LP$(I)))/2
);
1780 NEXT I
1790 '---------------
1800 LPRINT
1810 LPRINT "                  ";
1820    FOR I=1 TO LP : LPRINT "Lat      Long      "; : NEXT I
1830 LPRINT
1840 LPRINT "                  ";
1850    FOR I=1 TO LP : LPRINT "deg       deg      "; : NEXT I
1860 LPRINT
1870 LPRINT "  Origin     ";
1880 '---------------
1890 FOR I=1 TO LP
1900    LPRINT USING "  ###.### ";LAT(I);
1910    LPRINT USING " #####.###";LONG(I);
1920 NEXT I
1930 '---------------
1940 LPRINT
1950 LPRINT "  Other point";
1960 '---------------
1970 FOR I=1 TO LP
1980    LPRINT USING "  ###.### ";LAT(I+5);
1990    LPRINT USING " #####.###";LONG(I+5);
2000 NEXT I
2010 '---------------
2020 LPRINT
2030 '
2040 '    Printing the results
2050 '    --------------------
2060 LPRINT CHR$(27)"N"
2070 LPRINT "Corners                Distances" : LPRINT
2080 LPRINT "Pnt     Lat      Long ";
2090    FOR J=1 TO LP : LPRINT "  ";LP$(J); : NEXT J
2100 LPRINT
2110 LPRINT " No     deg       deg";
2120    FOR I=1 TO LP : LPRINT "    Km"; : NEXT I : LPRINT
2130 '---------------
2140 FOR I=1 TO NS
2150    LPRINT USING "###  ";I;                          'number
2160    LPRINT USING "###.###  ";B(I,1);                 'latitude
2170    LPRINT USING "#####.###  ";B(I,2);               'longitude
2180 '     ---------------
2190       FOR J=1 TO LP                        'distance along each projection
2200          LPRINT USING "#####  ";B(I,J+2);  'line from the origin to each
2210       NEXT J                               'projected point
2220 '     ---------------
2230    LPRINT
2240 NEXT I
2250 '---------------
2260 '
2270 END
```

B

★

This program computes the distances in km along projection lines (up to 5) of up to 10 points. These points may be the corners of polygons in which earthquakes are sorted to draw cross-sections (see SEISSORT).

TITLE : ? MAT 6

NUMBER OF SIDES of the polygon (up to 10) : ? 4

Corner 1 lat ? 15.397
long ? -102.35
Corner 2 lat ? 20.587
long ? -97.501
Corner 3 lat ? 20.36
long ? -96.413
Corner 4 lat ? 15.097
long ? -101.33

How many lines of projection (up to 5) : ? 1

LINE OF PROJECTION 1
Name (up to 4 letters) ? ___
1st point (origin) : lat ? 15.5
long ? -100.68
2nd point : lat ? 20.155
long ? -97.265

★

COMPUTING ...

PRINTER READY (Y/N) ? Y

C

MAT 6

PROJECTION LINE

	Lat deg	Long deg
Origin	16.500	-100.680
Other point	20.155	-97.265

Corners			Distances
Pnt No	Lat deg	Long deg	km
1	15.397	-102.350	-209
2	20.587	-97.501	561
3	20.360	-96.413	619
4	15.097	-101.330	-163

REPRESENTATION GRAPHIQUE
DES RESULTATS SUR DIAGRAMMES X-Y

8.1 - DESSIN DE POINTS SUR UN DIAGRAMME X-Y

A - Exemple de fichier de données

B - Listing du programme 'POINT PLOTTING'

C - Copie d'écran d'un exemple d'exécution du programme 'POINT PLOTTING'

D - Exemple de sortie sur traceur réduite de 63 %

8.2 - DESSIN DE LIGNES SUR UN DIAGRAMME X-Y

A - Exemple de fichier de données

B - Listing du programme 'LINEDRAW.BAS'

C - Copie d'écran d'un exemple d'exécution du programme 'LINEDRAW.BAS'

D - Exemple de sortie sur traceur réduite de 71%

8.1

A

```
10 REM ================================================================
20 REM
30 REM                    DATA  CRI MAGNETIC DEVIATION'
40 REM
50 REM                                        JACQUELINE ROUMP, JAN. 1985
60 REM ================================================================
70 REM
80 REM TABLE OF THE GREENWICH TIME OF THE DAY VERSUS MAGNETIC DEVIATION
90 REM (DIURNAL VARIATION) FOR THE COSTA RICA I SURVEY.
100 REM
110 REM AO = NUMBER OF POINTS
120 REM A1 = TIME IN DECIMAL HOUR + MAGNETIC DEVIATION
130 AO=66
140 A1=2
150 DIM A(AO,A1)
160 DATA 11.2833,0,11.3333,0,13.5167,-6,13.6833,-6,14.8,14,16.2,36
170 DATA 16.4167,46,17.6667,30,19,15,20.16667,13,20.25,13,20.4167,13
180 DATA 20.5,12,20.75,12,21.6167,10,22.25,7,23.6667,0,0.1667,-2,0.3667
190 DAT -2,0.5167,-3,0.6667,-3,0.8,-4,0.8833,-4,0.9667,-5,1.15,-5,1.2667
200 DATA -6,6.7667,-43,7.05,-45,8.3333,-33,9.4667,-3,9.8167,-14,10.1167
210 DATA -23,11.3667,-9,12.7167,7,13.3,0,13.9167,3,14.0667,5,14.1667,7
220 DATA 14.25,6,14.3333,5,16.3333,51,16.7167,60,17.3333,41,17.6,37
230 DATA 17.9167,33,19.7333,16,19.9,14,20.0833,9,20.3333,8,21.7,0,22,-2
240 DATA 1.35,-21,3.4167,-32,3.9167,-15,4.0167,-12,4.1333,-8,4.3167,-6
250 DATA 4.6667,-4,4.8333,-2,5.0833,0,5.2833,-5,6.5,10,6.6667,22
260 DATA 6.8667,20,7.0833,15,9.1667,-38
270 READ A
```

B

```
10 REM ================================================================
20 REM
30 REM                    PROGRAM  POINT PLOTTING'
40 REM
50 REM                                        JACQUELINE ROUMP, DEC. 1984
60 REM ================================================================
70 REM
80 INIT
90 PAGE
100 REM     PRESENTATION AND FIRST QUESTIONS
110 REM     ================================
120 CHARSIZE 4
130 PRINT
140 PRINT "            POINT PLOTTING"
150 PRINT "               X-Y GRAPH"
160 PRINT
170 PRINT
180 CHARSIZE 3
190 PRINT "   THIS PROGRAM PLOTS DATA POINTS. THE DATA MAY BE "
200 PRINT "REPRESENTED BY 8 DIFFERENT SYMBOLS AND 3 DIFFERENT COLORS."
210 PRINT "   THE DATA ARE LOCATED IN ANOTHER FILE. THIS FILE MUST BE"
220 PRINT "NAMED A, AND IT MUST HAVE TWO VARIABLES, AO AND A1 THAT MUST"
230 PRINT "READ THE NUMBER OF LINES (AO) AND THE NUMBER OF LINES (A1)"
240 PRINT "OF THE TABLE."
250 PRINT "   THE LAST LINE OF THE DATA FILE MUST BE  READ A'."
260 PRINT
270 PRINT "   EXAMPLE OF FILE : "
280 PRINT "10 AO=3"
290 PRINT "20 A1=2"
300 PRINT "30 DATA 1,2,3,4,5,6"
310 PRINT "40 DIM A(AO,A1)"
320 PRINT "50 READ A"
330 PRINT
340 PRINT "   THE PLOT WILL BE DRAWN ON A BENSON 1332 PLOTTER IN AN"
350 PRINT "21*29.7 CM FORMAT."
360 PRINT
370 PRINT "WHAT IS THE NUMBER OF YOUR DATA FILE ? (IF THE FILE HAS"
380 PRINT "ALREADY BEEN LOADED TYPE 0) ";
390 INPUT N
400 IF N=0 THEN 470
410 PRINT " IS THE TAPE READY (Y/N) ? ";
420 INPUT R$
430 R$=SEG(R$,1,1)
440 IF R$<>"Y" THEN 410
450 FIND N
460 APPEND 510
470 PRINT
480 REM ----------------------------------------------------------------
490 REM     PLACE FOR THE DATA FILE
500 REM
510 REM ----------------------------------------------------------------
520 REM
530 REM     ADDITIONAL QUESTIONS
540 REM     ====================
550 PRINT
560 PRINT " TITLE OF THE PLOT ? ";
570 INPUT T$
580 PRINT "SUBTITLE ? ";
590 INPUT S$
600 PRINT
```

```
610 PRINT "THIS TABLE HAS ";A0;" LINES AND ";A1;" COLUMNS."
620 PRINT
630 PRINT "NUMBER OF X-Y DATA SETS TO PLOT ? ";
640 INPUT N
650 PRINT "TITLE OF THE X AXIS ?, AND UNIT ? ";
660 INPUT X$,V$
670 PRINT "TITLE OF THE Y AXIS ?, AND UNIT ? ";
680 INPUT Y$,W$
690 PRINT
700 PRINT "YOU WILL NEED 3 PENS ON THE PLOTTER, ALL BLACK, OR IN COLOR"
710 PRINT "      PEN 0 IS FOR THE AXES"
720 PRINT "      PEN 1 IS FOR THE TITLE"
730 PRINT "      PEN 2 IS FOR THE SUBTITLE"
740 REM
750 REM     QUESTIONS ABOUT THE COLUMNS AND LINES TO PLOT,
760 REM     WHICH KIND OF SYMBOL TO USE AND WHICH COLOR
770 REM     ------------------------------------------
780 DIM B(N,6)
790 FOR I=1 TO N
800 CHARSIZE 3
810 PRINT "DATA SET ";I
820 CHARSIZE 2
830 PRINT "   COLUMN NUMBER FOR THE X VARIABLE ? ";
840 INPUT B(I,1)
850 PRINT "   COLUMN NUMBER FOR THE Y VARIABLE ? ";
860 INPUT B(I,2)
870 PRINT "   DO YOU WANT TO USE ALL THE LINES OF THE TABLE (Y/N) ? ";
880 INPUT R$
890 R$=SEG(R$,1,1)
900 IF R$<>"Y" THEN 940
910 B(I,3)=1
920 B(I,4)=A0
930 GO TO 980
940 PRINT "  NUMBER OF THE FIRST LINE TO PLOT ? ";
950 INPUT B(I,3)
960 PRINT "  NUMBER OF LINES TO BE USED ? ";
970 INPUT B(I,4)
980 PRINT "   CHOOSE AMONG THESE TYPES OF SYMBOL FOR THE POINTS : "
990 PRINT "1 : + ,    2 : * ,    3 : # ,    4 : $ ,    5 : ."
1000 PRINT "6 : O ,    7 : S ,    8 : X ,    9 : Z"
1010 INPUT B(I,5)
1020 PRINT "  IN WHICH COLOR (0 TO 2) ? ";
1030 INPUT B(I,6)
1040 PRINT
1050 NEXT I
1060 REM
1070 N0=0
1080 FOR I=1 TO N
1090 N0=N0+B(I,4)
1100 NEXT I
1110 REM
1120 REM     SEARCH FOR THE X AND Y MINIMUM AND MAXIMUM VALUES
1130 REM     ---------------------------------------------
1140 X1=A(B(1,3),B(1,1))
1150 X2=X1
1160 Y1=A(B(1,3),B(1,2))
1170 Y2=Y1
1180 FOR I=1 TO N
1190 FOR J=B(I,3) TO B(I,3)+B(I,4)-1
1200 X1=X1 MIN A(J,B(I,1))
1210 X2=X2 MAX A(J,B(I,1))
1220 Y1=Y1 MIN A(J,B(I,2))
1230 Y2=Y2 MAX A(J,B(I,2))
1240 NEXT J
1250 NEXT I
1260 REM
1270 REM     QUESTIONS FOR THE GRAPH
1280 REM     -----------------------
1290 REM  MAXIMUM AND MINIMUM VALUES
1300 REM
1310 PRINT
1320 PRINT "X MIN = ";X1;"     X MAX = ";X2
1330 PRINT "Y MIN = ";Y1;"     Y MAX = ";Y2
1340 PRINT
1350 PRINT "   FOR THE GRAPH : X MIN ?, X MAX ? ";
1360 INPUT X3,X4
1370 PRINT "                   Y MIN ?, Y MAX ? ";
1380 INPUT Y3,Y4
1390 PRINT
1400 IF X3<X4 THEN 1440
1410 X0=X4
1420 X4=X3
1430 X3=X0
1440 IF Y3<Y4 THEN 1510
1450 Y0=Y4
1460 Y4=Y3
1470 Y3=Y0
1480 REM
1490 REM  TICK MARKS AND LABELS
1500 REM
1510 PRINT " X AXIS : TICK MARKS AND LABELS EVERY ? UNITS ";
1520 INPUT X5,X6
1530 PRINT "          NUMBER OF DECIMAL CHARACTERS ? ";
1540 INPUT X7
1550 PRINT " Y AXIS : TICK MARKS AND LABELS EVERY ? UNITS ";
1560 INPUT Y5,Y6
1570 PRINT "          NUMBER OF DECIMAL CHARACTERS ? ";
1580 INPUT Y7
1590 REM
1600 REM  LINES OF COMMENTS
1610 REM
1620 PRINT "DO YOU WANT TO PRINT SOME LINES OF COMMENTS (UP TO 8) (Y/N)"
1630 INPUT R$
1640 R$=SEG(R$,1,1)
1650 IF R$="Y" THEN 1680
1660 D=0
1670 GO TO 1880
1680 PRINT "     1ST LINE : "
1690 INPUT A$
1700 D=1
1710 PRINT "     2ND LINE (IF NONE PRESS THE SPACE BAR AND RETURN) :"
1720 INPUT B$
1730 L$=" "
1740 IF B$=L$ THEN 2030
1750 D=2
1760 PRINT "     3RD LINE (IF NONE PRESS THE SPACE BAR AND RETURN) :"
1770 INPUT C$
1780 IF C$=L$ THEN 1880
1790 D=3
1800 PRINT "     4TH LINE (IF NONE PRESS THE SPACE BAR AND RETURN) :"
```

```
1810 INPUT D$
1820 IF D$=L$ THEN 2030
1830 D=4
1840 PRINT "     5TH LINE (IF NONE PRESS THE SPACE BAR AND RETURN) :"
1850 INPUT E$
1860 IF E$=L$ THEN 2030
1870 D=5
1880 PRINT "     6TH LINE (IF NONE PRESS THE SPACE BAR AND RETURN) :"
1890 INPUT F$
1900 IF F$=L$ THEN 2030
1910 D=6
1920 PRINT "     7TH LINE (IF NONE PRESS THE SPACE BAR AND RETURN) :"
1930 INPUT G$
1940 IF G$=L$ THEN 2030
1950 D=7
1960 PRINT "     8TH AND LAST LINE (IF NONE PRESS THE SPACE BAR,RETURN)"
1970 INPUT H$
1980 IF H$=L$ THEN 2030
1990 D=8
2000 REM
2010 REM    TABLE OFTHE POINTS CHARACTERISTICS
2020 REM    ---------------------------------
2030 DIM C(NO,4)
2040 K=1
2050 FOR I=1 TO N
2060 FOR J=B(I,3) TO B(I,3)+B(I,4)-1
2070 C(K,1)=A(J,B(I,1))
2080 C(K,2)=A(J,B(I,2))
2090 C(K,3)=B(I,5)
2100 C(K,4)=B(I,6)
2110 K=K+1
2120 NEXT J
2130 NEXT I
2140 REM ------------------------------------------------------------------
2150 REM
2160 REM     DRAWING PART
2170 REM     ============
2180 CHARSIZE 3
2190 PRINT
2200 PRINT "PLACE THE PEN AT THE RIGHT END OF THE PLOTTER"
2210 PRINT "READY TO PLOT (Y/N) ? ";
2220 INPUT R$
2230 R$=SEG(R$,1,1)
2240 IF R$<>"Y" THEN 2210
2250 REM
2260 REM    INITIALIZATION OF THE PLOTTER
2270 REM    ----------------------------
2280 GOSUB 5220
2290 REM
2300 REM    DRAWING THE FRAME
2310 REM    -----------------
2320 Z2=1
2330 Z0=21
2340 Z1=0
2350 GOSUB 5230
2360 Z1=29.7
2370 GOSUB 5230
2380 Z0=0
2390 GOSUB 5230
2400 Z1=0
2410 GOSUB 5230
2420 REM
2430 REM    TITLE
2440 REM    -----
2450 REM  SETTING CHARACTER TYPE AND ORIENTATION, AND COLOR
2460 REM
2470 Z4=0.3
2480 Z5=0.35
2490 Z9=0
2500 GOSUB 5250
2510 REM
2520 Z6=1
2530 GOSUB 5260
2540 REM
2550 REM  PRINTING THE TITLE TWICE
2560 Z$=T$
2570 Z3=LEN(Z$)
2580 Z2=0
2590 Z1=27
2600 Z0=11-Z3/2*Z4
2610 GOSUB 5230
2620 GOSUB 5240
2630 Z0=Z0+0.02
2640 GOSUB 5230
2650 GOSUB 5240
2660 REM
2670 REM  PRINTING THE SUBTITLE
2680 REM
2690 Z4=0.25
2700 Z5=0.2
2710 GOSUB 5250
2720 Z0=4.5
2730 Z1=25
2740 GOSUB 5230
2750 Z6=2
2760 GOSUB 5260
2770 Z$=S$
2780 Z3=LEN(Z$)
2790 GOSUB 5240
2800 REM
2810 REM  CHANGING THE COLOR
2820 REM
2830 Z6=0
2840 GOSUB 5260
2850 REM
2860 REM    VERTICAL AXIS
2870 REM    -------------
2880 REM  SETTING THE NEW CHARACTER TYPE
2890 REM
2900 Z4=0.2
2910 Z5=0.2
2920 GOSUB 5250
2930 REM
2940 Z7=Y7
2950 IF Z7<>0 THEN 2980
2960 Z7=-1
2970 REM
2980 REM  GOING TO THE ORIGIN
2990 REM
3000 Z0=4.5
```

```
3010 Z1=4+D*0.45
3020 GOSUB 5230
3030 REM
3040 REM  SETTING PARAMETERS
3050 REM
3060 L=17.5-D*0.45
3070 GO=(Y4-Y3)/Y5/L
3080 J1=(Y4-Y3)/Y5
3090 Z(15)=Y6/Y5
3100 REM
3110 REM  DRAWING THE AXIS AND TICK MARKS, AND PRINTING THE LABELS
3120 REM
3130 FOR K=1 TO J1+1
3140 Z2=1
3150 Z1=4+D*0.45+(K-1)/GO
3160 GOSUB 5230
3170 ZO=4.5-0.2
3180 GOSUB 5230
3190 IF K-1-INT((K-1)/Z(15))*Z(15)<>0 THEN 3260
3200 Z8=Y3+(K-1)*Y5
3210 GOSUB 4140
3220 ZO=ZO-0.2-(LEN(Z$)-0.5)*Z4
3230 Z2=0
3240 GOSUB 5230
3250 GOSUB 5240
3260 ZO=4.5
3270 GOSUB 5230
3280 NEXT K
3290 REM
3300 REM  DRAWING THE END OF THE AXIS
3310 REM
3320 Z2=1
3330 Z1=4+D*0.45+L
3340 GOSUB 5230
3350 REM
3360 REM  PRINTING THE TITLE AND UNIT
3370 REM
3380 Z2=0
3390 Z$=Y$
3400 Z3=LEN(Z$)
3410 Z1=4+D*0.45+L+1
3420 ZO=4.5-Z3/2*Z4
3430 GOSUB 5230
3440 GOSUB 5240
3450 Z$=W$
3460 Z3=LEN(Z$)
3470 Z1=4+D*0.45+L+0.5
3480 ZO=4.5-Z3/2*Z4
3490 GOSUB 5230
3500 GOSUB 5240
3510 REM
3520 REM    HORIZONTAL AXIS
3530 REM    ---------------
3540 REM  GOING TO THE ORIGIN OF THE AXIS
3550 REM
3560 Z7=X7
3570 IF Z7<>0 THEN 3590
3580 Z7=-1
3590 ZO=4.5
3600 Z1=4+D*0.45
3610 GOSUB 5230
3620 REM
3630 REM  SETTING PARAMETERS
3640 REM
3650 L=14.5
3660 GO=(X4-X3)/X5/L
3670 J1=(X4-X3)/X5
3680 Z(15)=X6/X5
3690 REM
3700 REM  DRAWING THE AXIS AND TICK MARKS, AND PRINTING THE LABELS
3710 REM
3720 FOR K=1 TO J1+1
3730 Z2=1
3740 ZO=4.5+(K-1)/GO
3750 GOSUB 5230
3760 Z1=4+D*0.45-0.2
3770 GOSUB 5230
3780 IF K-1-INT((K-1)/Z(15))*Z(15)<>0 THEN 3860
3790 Z8=X3+(K-1)*X5
3800 GOSUB 4140
3810 Z1=Z1-0.2-1.5*Z5
3820 ZO=ZO-(LEN(Z$)/2-0.5)*Z4
3830 Z2=0
3840 GOSUB 5230
3850 GOSUB 5270
3860 Z1=4+D*0.45
3870 ZO=4.5+(K-1)/GO
3880 GOSUB 5230
3890 NEXT K
3900 REM
3910 REM  DRAWING THE END OF THE AXIS
3920 REM
3930 Z2=1
3940 ZO=4.5+L
3950 GOSUB 5230
3960 REM
3970 REM  PRINTING THE TITLE AND UNIT
3980 REM
3990 Z2=0
4000 Z$=X$
4010 Z3=LEN(Z$)
4020 Z1=4+D*0.45-0.2-5.5*Z5
4030 ZO=4.5+L-Z3*Z4
4040 GOSUB 5230
4050 GOSUB 5240
4060 Z$=V$
4070 Z3=LEN(Z$)
4080 Z1=4+D*0.45-0.2-7.5*Z5
4090 ZO=4.5+L-(LEN(X$)/2+LEN(Z$)/2)*Z4
4100 GOSUB 5230
4110 GOSUB 5240
4120 GO TO 4340
4130 REM
4140 REM    SUBROUTINE TO PRINT LABELS ALONG THE AXES
4150 REM    ----------------------------------------
4160 Z8=INT(Z8+1.0E-5)+INT((Z8-INT(Z8+1.0E-5))*10↑Z7+1.0E-5)/10↑Z7
4170 Z$=STR(Z8)
4180 V8=Z8-INT(Z8)
4190 Q$=STR(V8)
4200 V7=LEN(Q$)
```

```
4210 IF V7=Z7+3 THEN 4310
4220 IF V8<>0 THEN 4260
4230 M$="."
4240 V7=3
4250 GO TO 4270
4260 M$=""
4270 FOR J2=1 TO Z7+3-V7
4280 M$=M$&"0"
4290 NEXT J2
4300 Z$=Z$&M$
4310 Z3=LEN(Z$)
4320 RETURN
4330 REM
4340 REM     DATA DRAWING
4350 REM     ------------
4360 REM
4370 G0=(X4-X3)/14.5
4380 G1=(Y4-Y3)/(17.5-D*0.45)
4390 Z4=0.2
4400 Z5=0.2
4410 GOSUB 5250
4420 FOR I=1 TO NO
4430 REM
4440 Z0=4.5+(C(I,1)-X3)/G0
4450 Z1=4+D*0.45+(C(I,2)-Y3)/G1
4460 GOSUB 5230
4470 Z6=C(I,4)
4480 GOSUB 5260
4490 GOSUB C(I,3) OF 4560,4590,4620,4650,4680,4710,4740,4770,4800
4500 Z1=Z1-0.1
4510 Z3=LEN(Z$)
4520 GOSUB 5230
4530 GOSUB 5240
4540 NEXT I
4550 GO TO 4840
4560 Z$="+"
4570 Z0=Z0+0.08
4580 RETURN
4590 Z$="*"
4600 Z0=Z0+0.08
4610 RETURN
4620 Z$="#"
4630 Z0=Z0+0.1
4640 RETURN
4650 Z$="$"
4660 Z0=Z0+0.1
4670 RETURN
4680 Z$="."
4690 Z0=Z0+0.04
4700 RETURN
4710 Z$="O"
4720 Z0=Z0+0.1
4730 RETURN
4740 Z$="S"
4750 Z0=Z0+0.1
4760 RETURN
4770 Z$="X"
4780 Z0=Z0+0.1
4790 RETURN
4800 Z$="Z"
4810 Z0=Z0+0.1
4820 RETURN
4830 REM
4840 REM     COMMENTS PRINTING
4850 REM     ----------------
4860 IF D=0 THEN 5140
4870 Z0=4.5
4880 Z6=0
4890 GOSUB 5260
4900 FOR I=1 TO D
4910 Z1=4+D*0.45-2-I*0.45
4920 GOSUB I OF 4980,5000,5020,5040,5060,5080,5100,5120
4930 Z3=LEN(Z$)
4940 GOSUB 5230
4950 GOSUB 5240
4960 NEXT I
4970 GO TO 5140
4980 Z$=A$
4990 RETURN
5000 Z$=B$
5010 RETURN
5020 Z$=C$
5030 RETURN
5040 Z$=D$
5050 RETURN
5060 Z$=E$
5070 RETURN
5080 Z$=F$
5090 RETURN
5100 Z$=G$
5110 RETURN
5120 Z$=H$
5130 RETURN
5140 Z0=24
5150 Z1=0
5160 GOSUB 5230
5170 END
5180 REM ------------------------------------------------------------------
5190 REM
5200 REM     BENSON PLOTTER SUBROUTINES
5210 REM     ==========================
5220 GO TO 5290
5230 GO TO 5530
5240 GO TO 5620
5250 GO TO 5920
5260 GO TO 6060
5270 GO TO 6150
5280 REM
5290 REM     INITIALIZATION OF THE BENSON 1332 PLOTTER
5300 REM     -----------------------------------------
5310 DIM Z(25),Z$(80)
5320 ON SRQ THEN 5510
5330 PRINT "BENSON ADDRESS ? ";
5340 INPUT Z(25)
5350 PRINT @Z(25),"76 !";
5360 Z$=CHR(34)
5370 PRINT @Z(25),Z$;"#$%&'()*+,-./0123456789:;<=>?";
5380 PRINT @Z(25),"@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]↑←";
5390 PRINT @Z(25), USING "A/L"," "
5400 PRINT @Z(25),"76                                    ";
```

```
5410 PRINT @Z(25),"                                ",
5420 PRINT @Z(25), USING "A/L",";" "
5430 Z(1)=0
5440 Z(2)=0
5450 Z(3)=200
5460 Z(5)=60
5470 Z(6)=0
5480 Z(13)=0
5490 Z(20)=Z(3)
5500 Z(21)=Z(3)
5510 RETURN
5520 REM
5530 REM     MOVING THE PEN
5540 REM     --------------
5550 Z(7)=INT(Z0*Z(20))
5560 Z(8)=INT(Z1*Z(21))
5570 Z(9)=Z(7)-Z(1)
5580 Z(10)=Z(8)-Z(2)
5590 Z(1)=Z(7)
5600 Z(2)=Z(8)
5610 Z(7)=ABS(Z(9))
5620 Z(8)=ABS(Z(10))
5630 IF Z(7)>16383 THEN 5720
5640 IF Z(8)>16383 THEN 5720
5650 Z(7)=0
5660 Z(11)=Z(9)
5670 Z(12)=Z(10)
5680 PRINT @Z(25),"PL ",Z(11),Z(12),Z2,
5690 PRINT @Z(25), USING "A/L",";" "
5700 IF Z(7)>1 THEN 5610
5710 RETURN
5720 Z(7)=INT((Z(7)+16383)/16384)
5730 Z(8)=INT((Z(8)+16383)/16384)
5740 IF Z(7)>Z(8) THEN 5760
5750 Z(7)=Z(8)
5760 Z(11)=INT(Z(9)/Z(7))
5770 Z(12)=INT(Z(10)/Z(7))
5780 Z(9)=Z(9)-Z(11)
5790 Z(10)=Z(10)-Z(12)
5800 GO TO 5680
5810 REM
5820 REM     WRITING A STRING OF CHARACTERS
5830 REM     ------------------------------
5840 PRINT @Z(25),"TX ",Z3," ",Z$,
5850 PRINT @Z(25), USING "A/L",";" "
5860 Z(1)=Z(1)+Z(5)*Z3
5870 Z(2)=Z(2)+Z(6)*Z3
5880 PRINT @Z(25),"NT ",Z(5)*Z3," ",Z(6)*Z3,
5890 PRINT @Z(25), USING "A/L",";" "
5900 RETURN
5910 REM
5920 REM     INITIALIZATION OF CHARACTER TYPE AND ORIENTATION
5930 REM     -----------------------------------------------
5940 Z(5)=Z4*COS(Z9)
5950 Z(6)=Z4*SIN(Z9)
5960 Z(16)=INT(4*Z(5)*Z(20)/3)
5970 Z(17)=INT(-4*Z5*Z(20)*SIN(Z9)/3)
5980 Z(18)=INT(4*Z(6)*Z(21)/3)
5990 Z(19)=INT(4*Z5*Z(21)*COS(Z9)/3)
6000 PRINT @Z(25),"SZ ",Z(16)," ",Z(17)," ",Z(18)," ",Z(19),
6010 PRINT @Z(25), USING "A/L",";" "
6020 Z(5)=INT(Z(5)*Z(20))
6030 Z(6)=INT(Z(6)*Z(21))
6040 RETURN
6050 REM
6060 REM     CHANGING THE PEN
6070 REM     ----------------
6080 Z(14)=INT(ABS(Z6))-3*INT(INT(ABS(Z6))/3)
6090 IF Z(14)=Z(13) THEN 6130
6100 PRINT @Z(25),"NP ",Z(14),
6110 PRINT @Z(25), USING "A/L",";" "
6120 Z(13)=Z(14)
6130 RETURN
6140 REM
6150 REM     PRINTING OF NUMBERS
6160 REM     -------------------
6170 Z7=INT(Z7)
6180 Z(9)=ABS(INT(Z7))
6190 IF Z7>-2 THEN 6370
6200 IF Z(9)<10 THEN 6220
6210 Z(9)=9
6220 Z$=STR(Z(9))
6230 Z$=Z$&"E"
6240 Z(9)=Z(9)+7
6250 IF Z8>0 THEN 6290
6260 Z(9)=Z(9)+1
6270 PRINT @Z(25),"TX ",Z(9)," ",
6280 GO TO 6300
6290 PRINT @Z(25),"TX ",Z(9),
6300 PRINT @Z(25), USING Z$,Z8,
6310 PRINT @Z(25), USING "A/L",";" "
6320 Z(1)=Z(1)+Z(9)*Z(5)
6330 Z(2)=Z(2)+Z(9)*Z(6)
6340 PRINT @Z(25),"NT ",Z(9)*Z(5)," ",Z(9)*Z(6),
6350 PRINT @Z(25), USING "A/L",";" "
6360 RETURN
6370 Z(7)=INT(LGT(ABS(Z8)+0.5))+1
6380 IF Z(7)>0 THEN 6400
6390 Z(7)=1
6400 IF Z8=>0 THEN 6420
6410 Z(7)=Z(7)+1
6420 Z$=STR(Z(7))
6430 Z$=Z$&"D"
6440 IF Z7>-1 THEN 6470
6450 Z(9)=Z(7)
6460 GO TO 6270
6470 Z(8)=LEN(Z$)+2
6480 Z$=Z$&". D"
6490 IF Z(9)<10 THEN 6510
6500 Z(9)=9
6510 GO TO Z(9)+1 OF 6520,6540,6560,6580,6600,6620,6640,6660,6680,6700
6520 Z$=REP(" ",Z(8)+1,1)
6530 GO TO 6710
6540 Z$=REP("1",Z(8),1)
6550 GO TO 6710
6560 Z$=REP("2",Z(8),1)
6570 GO TO 6710
6580 Z$=REP("3",Z(8),1)
6590 GO TO 6710
6600 Z$=REP("4",Z(8),1)
```

```
6610 GO TO 6710
6620 Z$=REP("5",Z(8),1)
6630 GO TO 6710
6640 Z$=REP("6",Z(8),1)
6650 GO TO 6710
6660 Z$=REP("7",Z(8),1)
6670 GO TO 6710
6680 Z$=REP("8",Z(8),1)
6690 GO TO 6710
6700 Z$=REP("9",Z(8),1)
6710 Z(9)=Z(9)+Z(7)+1
6720 GO TO 6270
```

C

★

POINT PLOTTING
X-Y GRAPH

THIS PROGRAM PLOTS DATA POINTS. THE DATA MAY BE REPRESENTED BY 8 DIFFERENT SYMBOLS AND 3 DIFFERENT COLORS.
THE DATA ARE LOCATED IN ANOTHER FILE. THIS FILE MUST BE NAMED A, AND IT MUST HAVE TWO VARIABLES, A0 AND A1 THAT MUST READ THE NUMBER OF LINES (A0) AND THE NUMBER OF LINES (A1) OF THE TABLE.
THE LAST LINE OF THE DATA FILE MUST BE 'READ A'.

EXAMPLE OF FILE

```
10 A0=3
20 A1=2
30 DATA 1,2,3,4,5,6
40 DIM A(A0,A1)
50 READ A
```

THE PLOT WILL BE DRAWN ON A BENSON 1332 PLOTTER IN AN 21*29.7 CM FORMAT.

```
WHAT IS THE NUMBER OF YOUR DATA FILE ? (IF THE FILE HAS
ALREADY BEEN LOADED TYPE 0) 13
 IS THE TAPE READY (Y/N) ? Y


 TITLE OF THE PLOT ? COSTA RICA I MAGNETIC DEVIATION
SUBTITLE ? GREENWICH TIME

THIS TABLE HAS 66 LINES AND 2 COLUMNS.

NUMBER OF X-Y DATA SETS TO PLOT ? 3
TITLE OF THE X AXIS ?, AND UNIT ? TIME
HOUR
TITLE OF THE Y AXIS ?, AND UNIT ? DEVIATION
NT

YOU WILL NEED 3 PENS ON THE PLOTTER, ALL BLACK, OR IN COLOR
      PEN 0 IS FOR THE AXES
      PEN 1 IS FOR THE TITLE
      PEN 2 IS FOR THE SUBTITLE
DATA SET 1
   COLUMN NUMBER FOR THE X VARIABLE ? 1
   COLUMN NUMBER FOR THE Y VARIABLE ? 2
   DO YOU WANT TO USE ALL THE LINES OF THE TABLE (Y/N) ? N
  NUMBER OF THE FIRST LINE TO PLOT ? 1
  NUMBER OF LINES TO BE USED ? 17
   CHOOSE AMONG THESE TYPES OF SYMBOL FOR THE POINTS :
1 : + ;    2 : X ;    3 : * ;    4 : 0 ;    5 : .
6 : O ;    7 : S ;    8 : X ;    9 : Z
1
  IN WHICH COLOR (0 TO 21) ? 0
DATA SET 2
   COLUMN NUMBER FOR THE X VARIABLE ? 1
   COLUMN NUMBER FOR THE Y VARIABLE ? 2
   DO YOU WANT TO USE ALL THE LINES OF THE TABLE (Y/N) ? N
  NUMBER OF THE FIRST LINE TO PLOT ? 18
  NUMBER OF LINES TO BE USED ? 34
   CHOOSE AMONG THESE TYPES OF SYMBOL FOR THE POINTS :
1 : + ;    2 : X ;    3 : * ;    4 : 0 ;    5 : .
6 : O ;    7 : S ;    8 : X ;    9 : Z
2
  IN WHICH COLOR (0 TO 21) ? 0
```

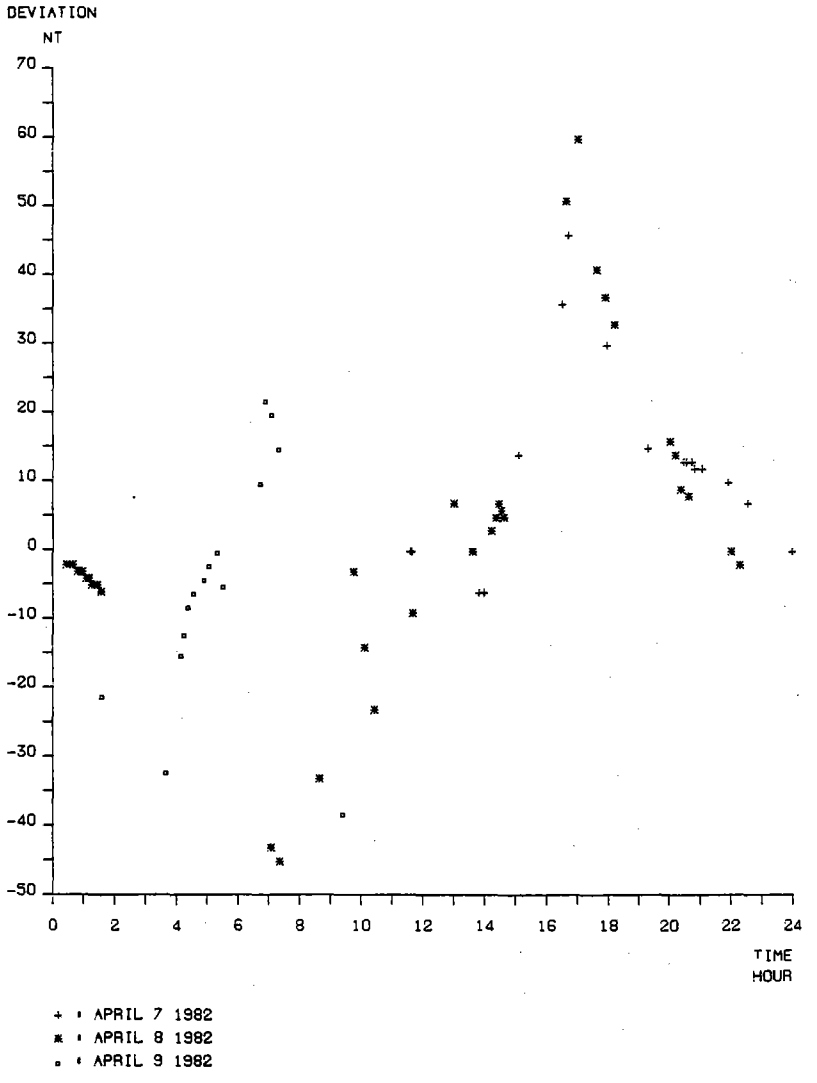
DATA SET 3

```
   COLUMN NUMBER FOR THE X VARIABLE ? 1
   COLUMN NUMBER FOR THE Y VARIABLE ? 2
   DO YOU WANT TO USE ALL THE LINES OF THE TABLE (Y/N) ? N
  NUMBER OF THE FIRST LINE TO PLOT ? 52
  NUMBER OF LINES TO BE USED ? 15
   CHOOSE AMONG THESE TYPES OF SYMBOL FOR THE POINTS :
1 : + ;    2 : # :    3 : * :    4 : $ :    5 : .
6 : O ;    7 : S ;    8 : X ;    9 : Z
5
  IN WHICH COLOR (0 TO 2) ? 0

X MIN = 0.1667      X MAX = 23.6667
Y MIN = -45      Y MAX = 60

   FOR THE GRAPH : X MIN ?, X MAX ? 0,24
                   Y MIN ?, Y MAX ? -50,70

 X AXIS : TICK MARKS AND LABELS EVERY ? UNITS 1,2
          NUMBER OF DECIMAL CHARACTERS ? 0
 Y AXIS : TICK MARKS AND LABELS EVERY ? UNITS 5,10
          NUMBER OF DECIMAL CHARACTERS ? 0
DO YOU WANT TO PRINT SOME LINES OF COMMENTS (UP TO 8) (Y/N)
Y
     1ST LINE
+ : APRIL 7 1982
     2ND LINE (IF NONE PRESS THE SPACE BAR AND RETURN) :
# : APRIL 8 1982
     3RD LINE (IF NONE PRESS THE SPACE BAR AND RETURN) :
* : APRIL 9 1982
     4TH LINE (IF NONE PRESS THE SPACE BAR AND RETURN) :

PLACE THE PEN AT THE RIGHT END OF THE PLOTTER
READY TO PLOT (Y/N) ? Y
BENSON ADDRESS ? 41
```

D

COSTA RICA I MAGNETIC DEVIATION

GREENWICH TIME



8.2

A

```
 30          39
 18.68 -104.96 0 5.26 33.155 4.773 35.396 5.744 31.041 29.303 5.508 37.363 4.983 4.284 3
7.78 6.225 29.028 25.748 5.723 42.003
 4.679 3.28 24.49 2.62 24.939 3.934 23.907 18.042 3.216 31.152 3.124 1.956 25.132 4.583
23.184 11.075 2.888 38.678 2.78
 18.37 -104.55 55 5.396 33.117 4.91 35.315 5.879 31.035 29.35 5.645 37.224 5.119 4.422 3
7.641 6.358 29.046 25.869 5.865 41.74
 4.817 3.423 24.93 2.764 25.457 4.077 24.282 18.766 3.376 31.328 3.264 2.102 25.792 4.72
4 23.514 12.211 3.076 38.524 2.925
 17.77 -103.73 165 5.663 32.988 5.18 35.114 6.143 30.961 29.38 5.916 36.908 5.386 4.694
37.342 6.619 29.03 26.036 6.141
 41.196 5.086 3.708 25.608 3.051 26.268 4.359 24.857 19.932 3.689 31.538 3.541 2.39 26.7
71 5.004 24.045 14.042 3.437 38.154
 3.215
 17.37 -102.78 272 5.921 32.204 5.438 34.219 6.399 30.266 28.764 6.174 35.927 5.644 4.95
3 36.353 6.872 28.406 25.567 6.402
 39.982 5.345 3.995 25.155 3.341 25.764 4.644 24.443 19.896 3.985 30.638 3.834 2.682 26.
253 5.285 23.661 14.492 3.762 36.682
 3.527
 16.96 -101.89 377 6.167 31.587 5.686 33.528 6.643 29.725 28.291 6.422 35.144 5.891 5.20
2 35.544 7.114 27.912 25.219 6.653
 39.003 5.595 4.269 24.888 3.617 25.487 4.914 24.186 19.974 4.269 30.009 4.111 2.96 25.9
68 5.552 23.413 14.969 4.07 35.604
 3.818
 16.64 -101.19 459 6.36 31.121 5.881 32.996 6.834 29.304 27.929 6.615 34.557 6.084 5.398
 34.952 7.303 27.542 24.95 6.848
 38.275 5.79 4.483 24.679 3.833 25.275 5.126 23.995 20.006 4.49 29.546 4.326 3.178 25.75
7 5.761 23.212 15.273 4.311 34.835
 4.043
 16.34 -100.17 570 6.612 30.152 6.135 31.944 7.084 28.418 27.102 6.868 33.425 6.338 5.65
2 33.798 7.551 26.726 24.249 7.102
 36.955 6.046 4.771 23.868 4.124 24.39 5.41 23.242 19.488 4.78 28.415 4.622 3.472 24.773
 6.041 22.507 15.065 4.614 33.314
 4.354
 16.04 -99.36 662 6.822 29.541 6.346 31.273 7.292 27.857 26.595 7.078 32.696 6.549 5.866
 33.053 7.756 26.207 23.834 7.315
 36.089 6.257 5.008 23.462 4.364 23.959 5.644 22.857 19.298 5.022 27.78 4.861 3.714 24.3
5 6.272 22.141 15.107 4.869 32.41
 4.604
 15.77 -98.65 744 7.007 29.041 6.533 30.72 7.475 27.394 26.181 7.263 32.1 6.734 6.054 32
.453 7.936 25.789 23.496 7.502 35.381
 6.444 5.216 23.146 4.575 23.631 5.849 22.552 19.156 5.234 27.282 5.071 3.927 24.01 6.47
4 21.847 15.152 5.095 31.7 4.821
 .
 .
 .
 .
 .
 .
 .
 .
 7.83 -82.99 2660 11.08 23.602 10.686 24.813 11.468 22.375 21.832 11.346 25.448 10.818 1
0.278 26.02 11.842 21.141 20.136
 11.608 27.377 10.552 9.82 21.446 9.265 22.075 10.36 20.746 19.447 9.978 23.535 9.625 8.
696 22.636 10.884 19.984
 17.525 10.086 25.727 9.398
 7.32 -82.83 2703 11.17 24.067 10.781 25.288 11.558 22.837 22.301 11.438 25.91 10.91 10.
377 26.495 11.929 21.591 20.607
 11.701 27.833 10.644 9.917 22.065 9.365 22.722 10.455 21.342 20.079 10.084 24.144 9.716
 8.798 23.314 10.977 20.56
 18.172 10.204 26.325 9.485
```

B

```
10 ' ************************************************************************
20 ' *                                                                      *
30 ' *                          PROGRAM 'LINEDRAW.BAS'                      *
40 ' *                                                                      *
50 ' * adaptation of 'LINE DRAWING' on Tektronix for a Benson 1332 plotter, *
60 ' * made in Sept. 1984                          Jacqueline ROUMP, June 1987 *
70 ' ************************************************************************
80 '
90 KEY OFF : WIDTH 80 : SCREEN 0,0,0
100 '
110 '     PRESENTATION AND FIRST QUESTIONS
120 '     ================================
130 '
140 PRINT : PRINT "                                LINE DRAWING"
150 PRINT "                                   X-Y PLOT"
160 PRINT : PRINT
170 PRINT "    This program draws lines on an X-Y plot from data input directl
y"
180 PRINT "to this program (and saved in a file) or from an old data file."
190 PRINT
200 PRINT "     The data file can be created and must be corrected using a"
210 PRINT "text editor. The first line has the number of lines, followed by"
220 PRINT "the number of columns. The subsequent lines have the X and Y"
230 PRINT "coordonates : you may have several Y for one X value ...."
240 PRINT
250 PRINT "    The plot will be drawn on a HP 7475A compatible plotter in the"
260 PRINT "21*29.7 cm format."
270 PRINT : PRINT
280 INPUT "Do you want to create a new data set (1) or to use an old data file
 (2) : ",ANS
290   IF ANS<>1 AND ANS<>2 THEN BEEP : GOTO 280
300   IF ANS=2 THEN 720
310 '
320 '     Data input from the keyboard
330 '     ----------------------------
340 '
350 PRINT : PRINT : PRINT "DATA INPUT" : PRINT
360 '
370 '     Data input from the keyboard
380 '     ----------------------------
390 '
400 INPUT "Number of lines : ",LN
410 INPUT "Number of columns : ",CN
420 DIM DATASET(LN,CN)                'data table
430 FOR I=1 TO LN
440     PRINT
450     PRINT "Column ";I
460     FOR J=1 TO CN
470         PRINT "       line ";J;"  : ";
480         INPUT ,DATASET(J,I)
490     NEXT J
500 NEXT I
510 PRINT
520 PRINT "Name of the data files on the default disk : " : PRINT
530 FILES "*.DAT"
```

```
540 PRINT : INPUT "Name of your data file : ",FILNAME$
550 OPEN "O",£1,FILNAME$
560 PRINT £1,LN,CN
570 FOR I=1 TO LN
580     A$=""
590     FOR J=1 TO CN
600         LENA=LEN(A$)
610         PLU$=STR$(DATASET(I,J))
620         LENP=LEN(PLU$)
630           IF LENA+LENP(125 THEN A$=A$+PLU$ : GOTO 660
640         PRINT £1,A$
650         A$=PLU$
660     NEXT J
670       IF LEN(A$)()0 THEN PRINT £1,A$
680 NEXT I
690 CLOSE
700 GOTO 890
710 '
720 '     Data input from a file
730 '     ----------------------
740 '
750 PRINT
760 PRINT "Name of the data files on the default disk : " : PRINT
770 FILES "*.DAT"
780 PRINT : INPUT "Name of your data file : ",FILNAME$
790 OPEN "I",£1,FILNAME$
800 INPUT £1,LN,CN
810 DIM DATASET(LN,CN)              'data table
820 FOR I=1 TO LN
830     FOR J=1 TO CN
840         INPUT £1,DATASET(I,J)
850     NEXT J
860 NEXT I
870 CLOSE
880 '
890 '     ADDITIONAL QUESTIONS
900 '     ====================
910 '
920 PRINT
930 INPUT "Title of the plot : ",TIT$
940 INPUT "Subtitle : ",SUBTIT$
950 PRINT
960 PRINT "This table has ";LN;" lines and ";CN;" columns."
970 PRINT
980 INPUT "Number of X-Y data sets to plot : ",NSET
990 INPUT "Title of the X axis : ",XTIT$
1000 INPUT "           and unit : ",XUNIT$
1010 INPUT "Title of the Y axis : ",YTIT$
1020 INPUT "           and unit : ",YUNIT$
1030 PRINT : PRINT
1040 PRINT "You will need 3 pens on the plotter, all black or in color :"
1050 PRINT "      Pen 1 is for the axes and the comments"
1060 PRINT "      Pen 2 is for the title"
1070 PRINT "      Pen 3 is for the subtitle"
1080 '
1090 '     Questions about the columns and the lines to plot,
1100 '     which kind of line to use and which color
1110 '     -----------------------------------------
1120 '
1130 DIM LINPAR(NSET,6)       'parameter table
1140 FOR I=1 TO NSET
1150     PRINT : PRINT "DATA SET ";I
1160     INPUT "     Column number for the X variable : ",LINPAR(I,1)
1170     INPUT "     Column number for the Y variable : ",LINPAR(I,2)
1180     INPUT "     Do you want to use all the lines of the table (Y/N) : ",A
N$
1190     AN$=LEFT$(AN$,1)
1200       IF AN$()"Y" AND AN$()"y" AND AN$()"N" AND AN$()"n" THEN 1180
1210       IF AN$="Y" OR AN$="y" THEN LINPAR(I,3)=1 : LINPAR(I,4)=LN : GOTO 12
40
1220     INPUT "     Number of the first line to plot : ",LINPAR(I,3)
1230     INPUT "     Number of lines to be used : ",LINPAR(I,4)
1240     PRINT "     Choose among these 6 types of line drawing : "
1250     PRINT "             1 : full line"
1260     PRINT "             2 : very short dashes (0.8 mm)"
1270     PRINT "             3 : short dashes (1.2 mm)"
1280     PRINT "             4 : medium dashes (1.6 mm)"
1290     PRINT "             5 : long dashes (2.1 mm)"
1300     PRINT "             6 : very long dashes (2.8 mm)"
1310     INPUT  LINPAR(I,5)
1320     INPUT "     In which color (1 to 3) : ",LINPAR(I,6)
1330 NEXT I
1340 '
1350 '     Search for the X and Y minimum and maximum values
1360 '     ------------------------------------------------
1370 '
1380 XMIN=DATASET(LINPAR(1,3),LINPAR(1,1)) : XMAX=XMIN
1390 YMIN=DATASET(LINPAR(1,3),LINPAR(1,2)) : YMAX=YMIN
1400 FOR I=1 TO NSET
1410     FOR J=LINPAR(I,3) TO LINPAR(I,3)+LINPAR(I,4)-1
1420           IF DATASET(J,LINPAR(I,1))(XMIN THEN XMIN=DATASET(J,LINPAR(I,1))
 : GOTO 1440.
1430           IF DATASET(J,LINPAR(I,1)))XMAX THEN XMAX=DATASET(J,LINPAR(I,1))
1440           IF DATASET(J,LINPAR(I,2))(YMIN THEN YMIN=DATASET(J,LINPAR(I,2))
 : GOTO 1460
1450           IF DATASET(J,LINPAR(I,2)))YMAX THEN YMAX=DATASET(J,LINPAR(I,2))
1460     NEXT J
1470 NEXT I
1480 '
1490 '     Questions for the graph
1500 '     -----------------------
1510 '
1520 '     maximum and minimum values
1530 '     --------------------------
1540 PRINT
1550 PRINT "X min = ";XMIN;"     X max = ";XMAX
1560 PRINT "Y min = ";YMIN;"     Y max = ";YMAX
1570 PRINT
1580 INPUT "     For the graph      X min : ",XMIN
1590 INPUT "                        X max : ",XMAX
1600 INPUT "                        Y min : ",YMIN
1610 INPUT "                        Y max : ",YMAX
1620 PRINT
1630   IF XMAX(XMIN THEN SWAP XMIN,XMAX
1640   IF YMAX(YMIN THEN SWAP YMIN,YMAX
1650 '
```

```
1660 '     tick marks and labels
1670 '     ---------------------
1680 INPUT "X axis : tick marks every ? units : ",XTICK
1690 INPUT "         labels every ? units : ",XLABL
1700 INPUT "         number of decimal characters for the labels : ",XDEC
1710 PRINT
1720 INPUT "Y axis : tick marks every ? units : ",YTICK
1730 INPUT "         labels every ? units : ",YLABL
1740 INPUT "         number of decimal characters for the labels : ",YDEC
1750 '
1760 '     lines of comments
1770 '     -----------------
1780 PRINT : PRINT
1790 INPUT "Do you want to print some lines of comments (up to 10 ) (Y/N) : ",
AN$
1800 AN$=LEFT$(AN$,1)
1810   IF AN$()"Y" AND AN$()"y" AND AN$()"N" AND AN$()"n" THEN 1790
1820   IF AN$="N" OR AN$="n" THEN NCOM=0 : GOTO 1930
1830 PRINT "     1st line : " : INPUT COMM$(1) : NCOM=1
1840 PRINT "     2nd line (if no more line press only RETURN)"
1850 INPUT COMM$(2) : IF COMM$(2)="" THEN 1930 ELSE NCOM=2
1860 PRINT "     3rd line (if no more line press only RETURN)"
1870 INPUT COMM$(3) : IF COMM$(3)="" THEN 1930 ELSE NCOM=3
1880 FOR I=4 TO 10
1890     PRINT "    ";I;"th line (if no more line press only RETURN)"
1900     INPUT COMM$(I) : IF COMM$(I)="" THEN 1930 ELSE NCOM=I
1910 NEXT I
1920 '
1930 '-------------------------------------------------------------------------
1940 '
1950 '     DRAWING PART
1960 '     ============
1970 '
1980 PRINT : PRINT
1990 INPUT "PLOTTER READY (Y/N) : ",AN$
2000 AN$=LEFT$(AN$,1)
2010   IF AN$()"Y" AND AN$()"y" AND AN$()"N" AND AN$()"n" THEN 1990
2020   IF AN$="N" OR AN$="n" THEN 1990
2030 '
2040 '     Initialization of the plotter
2050 '     -----------------------------
2060 '
2070 OPEN "COM1:9600,S,7,1,RS,CS65535,DS,CD" AS 2
       'opening communication line with the plotter (serial interface)
2080 PRINT £2,"IN RO90 IP IW";          'initialisation + rotation
2090 CEND$=CHR$(3)                      'character marking end of label
2100 PRINT £2,"DT" CEND$;
2110 '
2120 '     Title
2130 '     -----
2140 '
2150 '     setting character type and orientation, and color
2160 '     -------------------------------------------------
2170 SIZEH=.3 : SIZEV=.35 : GOSUB 4840          'character width and height
2180 PENN$="2" : GOSUB 4910                     'choosing the pen
2190 '
2200 '     printing the title twice
2210 '     ------------------------
2220 CHAINN$=TIT$ : LTIT=LEN(TIT$)
2230 PENM=0
2240 X=11-LTIT/2*SIZEH : Y=27 : GOSUB 4720      'moving pen to X,Y
2250 GOSUB 4790                                 'printing the title
2260 X=X+.02 : GOSUB 4720                       'moving pen to X,Y
2270 GOSUB 4790                                 'printing the title
2280 '
2290 '     printing the subtitle
2300 '     ---------------------
2310 SIZEH=.25 : SIZEV=.2 : GOSUB 4840          'character width and height
2320 X=4.5 : Y=24.5 : GOSUB 4720                'moving pen to X,Y
2330 PENN$="3" : GOSUB 4910                     'choosing the pen
2340 CHAINN$=SUBTIT$
2350 GOSUB 4790                                 'printing the subtitle
2360 '
2370 '     changing the color
2380 '     ------------------
2390 PENN$="1" : GOSUB 4910                     'choosing the pen
2400 '
2410 '     Vertical axis
2420 '     -------------
2430 '
2440 '     setting the new character type
2450 '     ------------------------------
2460 SIZEH=.2 : SIZEV=.2 : GOSUB 4840           'character width and height
2470 '
2480 '     setting the number of decimal characters for numbers
2490 '     ---------------------------------------------------
2500 NPUIS=YDEC
2510   IF NPUIS()0 THEN 2540
2520 NPUIS=-1
2530 '
2540 '     going to the origin
2550 '     -------------------
2560 X=4.5 : Y=4+NCOM*.45 : GOSUB 4720          'moving pen to X,Y
2570 '
2580 '     setting parameters
2590 '     ------------------
2600 LENG=17.5-NCOM*.45
2610 STPT=(YMAX-YMIN)/YTICK/LENG
2620 NSTPT=(YMAX-YMIN)/YTICK
2630 NSTP=YLABL/YTICK
2640 '
2650 '     drawing the axis and tick marks, and printing the labels
2660 '     --------------------------------------------------------
2670 FOR K=1 TO NSTPT+1
2680     PENM=1
2690     Y=4+NCOM*.45+(K-1)/STPT : GOSUB 4720   'moving pen to X,Y
2700     X=4.5-.2 : GOSUB 4720                  'drawing tick mark
2710       IF K-1-INT((K-1)/NSTP+.00001)*NSTP()0 THEN 2770
2720     NUMB=YMIN+(K-1)*YTICK : GOSUB 3410     'label preparation
2730     X=X-.2-(LEN(CHAINN$))*SIZEH
2740     PENM=0
2750     GOSUB 4720                             'moving pen to X,Y
2760     GOSUB 4790                             'writing label
2770     X=4.5 : GOSUB 4720                     'moving pen to X,Y
2780 NEXT K
2790 '
```

```
2800 '     drawing the end of the axis
2810 '     ---------------------------
2820 PENM=1
2830 Y=4+NCOM*.45+LENG : GOSUB 4720                'moving pen to X,Y
2840 '
2850 '     printing the title and unit
2860 '     ---------------------------
2870 PENM=0
2880 CHAINN$=YTIT$
2890 Y=4+NCOM*.45+LENG+1 : X=4.5-LEN(CHAINN$)/2*SIZEH
2900 GOSUB 4720 : GOSUB 4790                       'writing title of axis
2910 CHAINN$=YUNIT$
2920 Y=4+NCOM*.45+LENG+.5 : X=4.5-LEN(CHAINN$)/2*SIZEH
2930 GOSUB 4720 : GOSUB 4790                       'writing unit of axis
2940 '
2950 '     Horizontal axis
2960 '     ---------------
2970 '
2980 '     going to the origin of the axis
2990 '     -------------------------------
3000 NPUIS=XDEC
3010   IF NPUIS<>0 THEN 3030
3020 NPUIS=-1
3030 X=4.5 : Y=4+NCOM*.45 : GOSUB 4720             'character width and height
3040 '
3050 '     setting parameters
3060 '     ------------------
3070 LENG=14.5
3080 STPT=(XMAX-XMIN)/XTICK/LENG
3090 NSTPT=(XMAX-XMIN)/XTICK
3100 NSTP=XLABL/XTICK
3110 '
3120 '     drawing the axis and tick marks, and printing the labels
3130 '     --------------------------------------------------------
3140 FOR K=1 TO NSTPT+1
3150     PENM=1
3160     X=4.5+(K-1)/STPT : GOSUB 4720             'moving pen to X,Y
3170     Y=4+NCOM*.45-.2 : GOSUB 4720              'drawing tick mark
3180       IF K-1-INT((K-1)/NSTP+.00001)*NSTP<>0 THEN 3220
3190     NUMB=XMIN+(K-1)*XTICK : GOSUB 3410        'label preparation
3200     Y=Y-.2-1.5*SIZEV : X=X-LEN(CHAINN$)/2*SIZEH
3210     PENM=0 : GOSUB 4720 : GOSUB 4790          'writing label
3220     Y=4+NCOM*.45 : X=4.5+(K-1)/STPT : GOSUB 4720     'moving pen to X,Y
3230 NEXT K
3240 '
3250 '     drawing the end of the axis
3260 '     ---------------------------
3270 PENM=1
3280 X=4.5+LENG : GOSUB 4720                       'moving pen to X,Y
3290 '
3300 '     printing the title and unit
3310 '     ---------------------------
3320 PENM=0
3330 CHAINN$=XTIT$
3340 Y=4+NCOM*.45-.2-5.5*SIZEV : X=4.5+LENG-LEN(CHAINN$)*SIZEH
3350 GOSUB 4720 : GOSUB 4790                       'writing title of axis
3360 CHAINN$=XUNIT$
3370 Y=4+NCOM*.45-.2-7.5*SIZEV : X=4.5+LENG-(LEN(XTIT$)/2-LEN(CHAINN$)/2)*SIZE
H
```

```
3380 GOSUB 4720 : GOSUB 4790                       'writing unit of axis
3390 GOTO 3600
3400 '
3410 '     Subroutine to print labels along the axis
3420 '     -----------------------------------------
3430 '
3440 NPUI=-NPUIS-3
3450 NUMB=NUMB+10^NPUI
3460   IF NPUIS<0 THEN CHAINN$=STR$(INT(NUMB)) : IF NUMB>=0 THEN CHAINN$=RIGH
T$(CHAINN$,LEN(CHAINN$)-1) : GOTO 3580 ELSE GOTO 3580
3470 CHN$=STR$(NUMB) : IF NUMB>=0 THEN CHN$=RIGHT$(CHN$,LEN(CHN$)-1)
3480 POINTPOS=INSTR(CHN$,".")
3490 CHAININ$=STR$(INT(NUMB))+"."
3500   IF POINTPOS=0 THEN CHAININ$=CHN$+"." : CHAINDEC$="" : GOTO 3540
3510 CHAINDEC$=MID$(CHN$,POINTPOS+1)
3520   IF LEN(CHAINDEC$)<NPUIS THEN 3540
3530 CHAINDEC$=LEFT$(CHAINDEC$,NPUIS) : GOTO 3570
3540 FOR J2=1 TO NPUIS-LEN(CHAINDEC$)
3550     CHAINDEC$=CHAINDEC$+"0"
3560 NEXT J2
3570 CHAINN$=CHAININ$+CHAINDEC$
3580 RETURN
3590 '
3600 '     Data drawing
3610 '     ------------
3620 '
3630 XSCAL=(XMAX-XMIN)/14.5
3640 YSCAL=(YMAX-YMIN)/(17.5-NCOM*.45)
3650 '
3660 '     lines drawing loop
3670 '     ------------------
3680 FOR I=1 TO NSET
3690     '
3700     'going to the first point of the line
3710     '
3720     X=4.5+(DATASET(LINPAR(I,3),LINPAR(I,1))-XMIN)/XSCAL
3730     Y=4+NCOM*.45+(DATASET(LINPAR(I,3),LINPAR(I,2))-YMIN)/YSCAL
3740     GOSUB 4720                                'moving pen to X,Y
3750     PENN$=RIGHT$(STR$(LINPAR(I,6)),1) : GOSUB 4910   'choosing pen
3760     PENM=1
3770     L1=0 : L2=0
3780     '
3790     'drawing the line segments
3800     '
3810     FOR J=LINPAR(I,3)+1 TO LINPAR(I,3)+LINPAR(I,4)-1
3820         ON LINPAR(I,5) GOSUB 3880,3970,3990,4010,4030,4050
3830     NEXT J
3840     PENM=0
3850 NEXT I
3860 GOTO 4520
3870 '
3880 '     subroutine to draw full lines
3890 '     -----------------------------
3900 X=4.5+(DATASET(J,LINPAR(I,1))-XMIN)/XSCAL
3910 Y=4+NCOM*.45+(DATASET(J,LINPAR(I,2))-YMIN)/YSCAL
3920 GOSUB 4720                                    'moving pen to X,Y
3930 RETURN
```

```
3940 '
3950 '     to set the length of the dashes for interrupted lines
3960 '     -----------------------------------------------------
3970 LLEN=.08 : GOSUB 4080
3980 RETURN
3990 LLEN=.12 : GOSUB 4080
4000 RETURN
4010 LLEN=.16 : GOSUB 4080
4020 RETURN
4030 LLEN=.21 : GOSUB 4080
4040 RETURN
4050 LLEN=.28 : GOSUB 4080
4060 RETURN
4070 '
4080 '     subroutine to draw dashes on segments forming one line
4090 '     ------------------------------------------------------
4100 ' L = length of a segment
4110 ' L2 = left piece of dash from preceding segment, to be drawn in the
        segment
4120 ' K8 and K9 = length of a dash along the X and Y axes
4130 ' L3 = length left to be drawn after the drawing of L2 length
4140 ' NDASH = number of dashes to be drawn plus spaces between them
4150 '
4160 G5=LINPAR(I,1) : G6=LINPAR(I,2)
4170 L=SQR(((DATASET(J,G5)-DATASET(J-1,G5))/XSCAL)^2+((DATASET(J,G6)-DATASET(J
-1,G6))/YSCAL)^2)
4180 K8=(DATASET(J,G5)-DATASET(J-1,G5))/XSCAL/L
4190 K9=(DATASET(J,G6)-DATASET(J-1,G6))/YSCAL/L
4200   IF L2=0 THEN 4320
4210   IF L>=L2 THEN 4270
4220 X=4.5+(DATASET(J,G5)-XMIN)/XSCAL
4230 Y=4+NCOM*.45+(DATASET(J,G6)-YMIN)/YSCAL
4240 GOSUB 4720                                    'moving pen to X,Y
4250 L2=L2-L
4260 GOTO 4470
4270 X=4.5+(DATASET(J-1,G5)-XMIN)/XSCAL+L2*K8
4280 Y=4+NCOM*.45+(DATASET(J-1,G6)-YMIN)/YSCAL+L2*K9
4290 GOSUB 4720                                    'moving pen to X,Y
4300   IF PENM=0 THEN PENM=1 : GOTO 4320
4310 PENM=0
4320 L3=L-L2
4330 NDASH=INT(L3/LLEN+.00001)
4340   IF L3<LLEN THEN 4420
4350 FOR K=1 TO NDASH
4360     X=4.5+(DATASET(J-1,G5)-XMIN)/XSCAL+L2*K8+K*LLEN*K8
4370     Y=4+NCOM*.45+(DATASET(J-1,G6)-YMIN)/YSCAL+L2*K9+K*LLEN*K9
4380     GOSUB 4720                                'moving pen to X,Y
4390       IF PENM=0 THEN PENM=1 : GOTO 4410
4400     PENM=0
4410 NEXT K
4420 L2=LLEN-(L3-NDASH*LLEN)
4430   IF L2=0 THEN 4470
4440 X=4.5+(DATASET(J,G5)-XMIN)/XSCAL
4450 Y=4+NCOM*.45+(DATASET(J,G6)-YMIN)/YSCAL
4460 GOSUB 4720                                    'moving pen to X,Y
4470 RETURN
4480 '
4490 '     Comment printing
4500 '     ----------------
4510 '
4520   IF NCOM=0 THEN 4620
4530 PENN$="1" : GOSUB 4910                        'choosing pen
4540 X=4.5
4550 FOR I=1 TO NCOM
4560     Y=4+NCOM*.45-2-I*.45
4570     CHAINN$=COMM$(I)
4580     GOSUB 4720                                'moving pen to X,Y
4590     GOSUB 4790                                'writing the comment line
4600 NEXT I
4610 '
4620 '     END
4630 '     ---
4640 '
4650 END
4660 '
4670 ' ---------------------------------------------------------------------
4680 '
4690 '          PLOTTER SUBROUTINES
4700 '          ===================
4710 '
4720 '     Moving the pen
4730 '     --------------
4740 '
4750 XX=(X-1)*1016/2.54 : YY=(Y-1)*1016/2.54
4760   IF PENM=0 THEN PRINT £2,"PU";XX;YY; ELSE PRINT £2,"PD";XX;YY;
4770 RETURN
4780 '
4790 '     Writing a string of characters
4800 '     ------------------------------
4810 '
4820 PRINT £2,"LB";CHAINN$;CEND$
4830 '
4840 '     Initialization of character type and orientation
4850 '     -----------------------------------------------
4860 '
4870 SIZEHH=SIZEH*2/3
4880 PRINT £2,"SI";SIZEHH;SIZEV;
4890 RETURN
4900 '
4910 '     Changing the pen
4920 '     ----------------
4930 '
4940 PRINT £2,"SP";PENN$;
4950 RETURN
```

C★

LINE DRAWING
X-Y PLOT

This program draws lines on an X-Y plot from data input directly to this program (and saved in a file) or from an old data file.

The data file can be created and must be corrected using a text editor. The first line has the number of lines, followed by the number of columns. The subsequent lines have the X and Y coordonates : you may have several Y for one X value ....

The plot will be drawn on a HP 7475A compatible plotter in the 21*29.7 cm format.

```
Do you want to create a new data set (1) or to use an old data file (2) : 2

Name of the data files on the default disk :

MIDAMEPL.DAT      MATPLATE.DAT

Name of your data file : MATPLATE.DAT

Title of the plot : MIDDLE AMERICA TRENCH
Subtitle : SPEED : COCOS-NORTH AMERICA & COCOS-CARIBBEAN COUPLES

This table has  30  lines and  39  columns.

Number of X-Y data sets to plot : 10
Title of the X axis : DISTANCE
          and unit : KM
Title of the Y axis : SPEED
          and unit : CM/YR


You will need 3 pens on the plotter, all black or in color :
      Pen 1 is for the axes and the comments
      Pen 2 is for the title
      Pen 3 is for the subtitle

DATA SET  1
     Column number for the X variable : 3
     Column number for the Y variable : 4
     Do you want to use all the lines of the table (Y/N) : N
     Number of the first line to plot : 1
     Number of lines to be used : 13
     Choose among these 6 types of line drawing :
           1 : full line
           2 : very short dashes (0.8 mm)
           3 : short dashes (1.2 mm)
           4 : medium dashes (1.6 mm)
           5 : long dashes (2.1 mm)
           6 : very long dashes (2.8 mm)
? 1
     In which color (1 to 3) : 1

DATA SET  2
     Column number for the X variable : 3
     Column number for the Y variable : 6
     Do you want to use all the lines of the table (Y/N) : N
     Number of the first line to plot : 1
     Number of lines to be used : 13
     Choose among these 6 types of line drawing :
           1 : full line
           2 : very short dashes (0.8 mm)
           3 : short dashes (1.2 mm)
           4 : medium dashes (1.6 mm)
           5 : long dashes (2.1 mm)
           6 : very long dashes (2.8 mm)
? 3
     In which color (1 to 3) : 1

DATA SET  3
     Column number for the X variable : 3
     Column number for the Y variable : 8
     Do you want to use all the lines of the table (Y/N) : N
     Number of the first line to plot : 1
     Number of lines to be used : 13
     Choose among these 6 types of line drawing :
           1 : full line
           2 : very short dashes (0.8 mm)
           3 : short dashes (1.2 mm)
           4 : medium dashes (1.6 mm)
           5 : long dashes (2.1 mm)
           6 : very long dashes (2.8 mm)
? 3
     In which color (1 to 3) : 1

DATA SET  4
     Column number for the X variable : 3
     Column number for the Y variable : 14
     Do you want to use all the lines of the table (Y/N) : N
     Number of the first line to plot : 1
     Number of lines to be used : 13
     Choose among these 6 types of line drawing :
           1 : full line
           2 : very short dashes (0.8 mm)
           3 : short dashes (1.2 mm)
           4 : medium dashes (1.6 mm)
           5 : long dashes (2.1 mm)
           6 : very long dashes (2.8 mm)
? 5
     In which color (1 to 3) : 1

DATA SET  5
     Column number for the X variable : 3
     Column number for the Y variable : 16
     Do you want to use all the lines of the table (Y/N) : N
```

```
     Number of the first line to plot : 1
     Number of lines to be used : 13
     Choose among these 6 types of line drawing :
            1 : full line
            2 : very short dashes (0.8 mm)
            3 : short dashes (1.2 mm)
            4 : medium dashes (1.6 mm)
            5 : long dashes (2.1 mm)
            6 : very long dashes (2.8 mm)
? 5
     In which color (1 to 3) : 1

DATA SET  6
     Column number for the X variable : 3
     Column number for the Y variable : 22
     Do you want to use all the lines of the table (Y/N) : N
     Number of the first line to plot : 13
     Number of lines to be used : 18
     Choose among these 6 types of line drawing :
            1 : full line
            2 : very short dashes (0.8 mm)
            3 : short dashes (1.2 mm)
            4 : medium dashes (1.6 mm)
            5 : long dashes (2.1 mm)
            6 : very long dashes (2.8 mm)
? 1
     In which color (1 to 3) : 1

DATA SET  7
     Column number for the X variable : 3
     Column number for the Y variable : 24
     Do you want to use all the lines of the table (Y/N) : N
     Number of the first line to plot : 13
     Number of lines to be used : 18
     Choose among these 6 types of line drawing :
            1 : full line
            2 : very short dashes (0.8 mm)
            3 : short dashes (1.2 mm)
            4 : medium dashes (1.6 mm)
            5 : long dashes (2.1 mm)
            6 : very long dashes (2.8 mm)
? 3
     In which color (1 to 3) : 1

DATA SET  8
     Column number for the X variable : 3
     Column number for the Y variable : 26
     Do you want to use all the lines of the table (Y/N) : N
     Number of the first line to plot : 13
     Number of lines to be used : 18
     Choose among these 6 types of line drawing :
            1 : full line
            2 : very short dashes (0.8 mm)
            3 : short dashes (1.2 mm)
            4 : medium dashes (1.6 mm)
            5 : long dashes (2.1 mm)
            6 : very long dashes (2.8 mm)
? 3
     In which color (1 to 3) : 1

DATA SET  9
     Column number for the X variable : 3
     Column number for the Y variable : 32
     Do you want to use all the lines of the table (Y/N) : N
     Number of the first line to plot : 13
     Number of lines to be used : 18
     Choose among these 6 types of line drawing :
            1 : full line
            2 : very short dashes (0.8 mm)
            3 : short dashes (1.2 mm)
            4 : medium dashes (1.6 mm)
            5 : long dashes (2.1 mm)
            6 : very long dashes (2.8 mm)
? 5
     In which color (1 to 3) : 1

DATA SET  10
     Column number for the X variable : 3
     Column number for the Y variable : 34
     Do you want to use all the lines of the table (Y/N) : N
     Number of the first line to plot : 13
     Number of lines to be used : 18
     Choose among these 6 types of line drawing :
            1 : full line
            2 : very short dashes (0.8 mm)
            3 : short dashes (1.2 mm)
            4 : medium dashes (1.6 mm)
            5 : long dashes (2.1 mm)
            6 : very long dashes (2.8 mm)
? 5
     In which color (1 to 3) : 1

X min =  0       X max =  2703
Y min =  4.284       Y max =  10.977

     For the graph       X min : 0
                         X max : 2800
                         Y min : 0
                         Y max : 12

X axis : tick marks every ? units : 100
         labels every ? units : 500
         number of decimal characters for the labels : 0

Y axis : tick marks every ? units : .5
         labels every ? units : 2
         number of decimal characters for the labels : 0
```

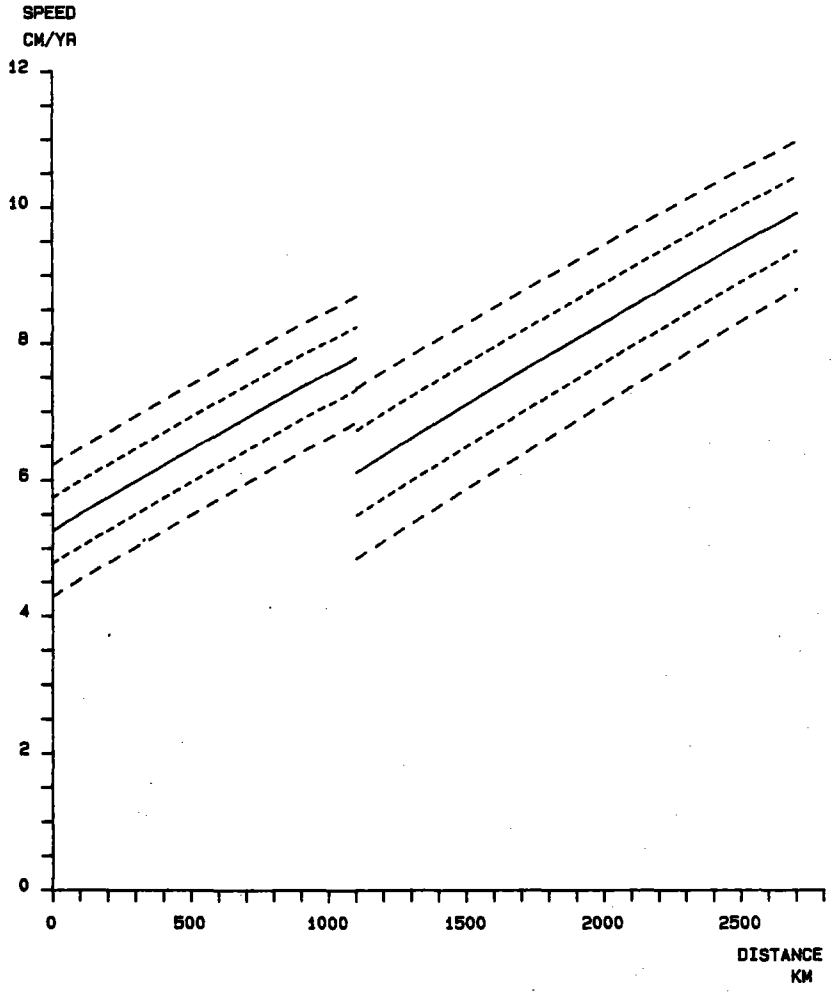
Do you want to print some lines of comments (up to 10) (Y/N) : Y
1st line :
? FIRST GROUP : COCOS-NORTH AMERICA PLATES COUPLE
2nd line (if no more line press only RETURN)
? SECOND GROUP : COCOS-CARIBBEAN PLATES COUPLE
3rd line (if no more line press only RETURN)
? FULL LINE : MEAN SPEED
4 th line (if no more line press only RETURN)
? SHORT DASHES : CONFIDENCE INTERVAL 70%
5 th line (if no more line press only RETURN)
? LONG DASHES : CONFIDENCE INTERVAL 95%
6 th line (if no more line press only RETURN)
?

PLOTTER READY (Y/N) : Y

D

MIDDLE AMERICA TRENCH

SPEED : COCOS-NORTH AMERICA & COCOS-CARIBBEAN COUPLES



FIRST GROUP : COCOS-NORTH AMERICA PLATES COUPLE
SECOND GROUP : COCOS-CARIBBEAN PLATES COUPLE
FULL LINE : MEAN SPEED
SHORT DASHES : CONFIDENCE INTERVAL 70%
LONG DASHES : CONFIDENCE INTERVAL 95%

REPRESENTATION GRAPHIQUE DES RESULTATS SUR CARTES

9.1 - DESSIN DE POINTS SUR UNE CARTE EN PROJECTION DE MERCATOR

A - Exemple de fichier de données

B - Listing du programme 'MERCATOR'

C - Copie d'écran d'un exemple d'exécution du programme 'MERCATOR'

D - Exemple de sortie sur traceur réduite de 55 %

9.2 - DESSIN DE VECTEURS SUR UNE CARTE EN PROJECTION DE MERCATOR

A - Exemple de fichier de données

B - Listing du programme 'MERVECTR.BAS'

C - Copie d'écran d'un exemple d'exécution du programme 'MERVECTR.BAS'

D - Exemple de sortie sur traceur réduite de 71%

9.1

A

```
10 REM ================================================================
20 REM
30 REM                          DATA  CIRCLES'
40 REM
50 REM                                        JACQUELINE ROUMP, OCT. 1984
60 REM ================================================================
70 REM
80 REM POSITION OF POINTS ALONG THE MIDDLE AMERICA TRENCH AND OF
90 REM CENTER OF THE CIRCLES COMPUTED BY THE PROGRAMS  DISSMCRC' AND
100 REM  DIRSMCRC'
110 REM A0 : NUMBER OF POINTS
120 REM A1 : NUMBER OF COLUMNS (LAT & LONG)
130 A0=66
140 A1=2
150 DIM A(A0,A1)
160 REM
170 REM  POSITIONS OF POINT ALONG THE TRENCH FOR DIRECTION CIRCLES
180 REM
190 DATA 18.45,-104.67,18.07,-104.17,17.62,-103.33,17.17,-102.33
200 DATA 16.83,-101.5,16.5,-100.67
210 DATA 15.97,-99.17,15.62,-98.17,15.38,-97.5,15.3,-96.5
220 DATA 14.77,-95,14.33,-94.42,14,-93.67,13.67,-93,13.1,-91.92
230 DATA 12.65,-90.83,12.3,-90,11.98,-89.17,11.65,-88.42
240 DATA 10.4,-86.83,9.38,-85.92,8.95,-85,8.5,-84.17
250 DATA 7.07,-82.22,6.78,-81.42,6.75,-80.75,6.82,-80.28
260 REM
270 REM  POSITIONS OF SUPPLEMENTARY POINTS FOR DISTANCE CIRCLES
280 REM
290 DATA 18.67,-104.97,18.21,-104.45,17.83,-103.75,17.37,-102.83
300 DATA 16.94,-101.83,16.67,-101.17,16.33,-100.17,16.17,-99.67
310 DATA 16.05,-99.35,15.75,-98.67,15.47,-97.75,15.32,-97.27,15.22,-95.8
320 DATA 14.97,-95.28,14.67,-94.83,14.18,-94,13.75,-93.32,13.47,-92.65
330 DATA 12.8,-91.22,12.5,-90.53,12.17,-89.58,11.78,-88.65,11.53,-88.17
340 DATA 10.83,-87.22,9.83,-86.33,9.15,-85.5,8.67,-84.48,8.37,-83.85
350 DATA 7.28,-82.5
360 REM
370 REM  POSITIONS OF CIRCLE CENTERS
380 REM
390 DATA 30.34,-95.11,22.29,-95.84,38.58,-76.22,14.24,-82.01,9.23,-80.94
400 DATA 31.51,-94.45,25.08,-94.98,38.04,-76.15,15.82,-80.3,10.32,-80.74
410 REM
420 READ A
```

B

```
10 REM ================================================================
20 REM
30 REM                          PROGRAM  MERCATOR'
40 REM
50 REM                                        JACQUELINE ROUMP, OCT. 1984
60 REM ================================================================
70 REM
80 REM     PRESENTATION
90 REM     ============
100 INIT
110 PAGE
120 SET DEGREES
130 CHARSIZE 4
140 PRINT "      MERCATOR PROJECTION MAP AND PLOTTING OF POINTS"
150 PRINT
160 PRINT
170 CHARSIZE 3
180 PRINT "PROGRAM TO PLOT POINTS ON A MERCATOR PROJECTION MAP (WGS72"
190 PRINT "ELLISOID) USING UP TO 9 SYMBOLS AND 3 COLORS. THE POINTS MAY"
200 PRINT "BE LABELLED BUT ONLY WITH NUMBERS."
210 PRINT
220 PRINT "THE DATA TABLE HAS BEEN CREATED IN ANOTHER FILE AND NAMED A."
230 PRINT "THIS DATA FILE MUST HAVE AT LEAST 2 COLUMNS : TWO FOR THE"
240 PRINT "LATITUDE AND LONGITUDE OF THE POINTS, A THIRD ONE MAY"
250 PRINT "CONTAIN A NUMBER WHICH MAY BE PRINTED ON THE MAP."
260 PRINT "   THIS FILE MUST ALSO HAVE TWO VARIABLES, A0 AND A1 WHICH"
270 PRINT "CONTAIN THE NUMBER OF LINES OF THE TABLE (A0) AND ITS NUMBER"
280 PRINT "OF COLUMNS (A1). THE FILE MUST ALSO HAVE TWO LINES WHICH"
290 PRINT "ARE : DIM A(A0,A1), AND READ A. THE FIRST LINE CAN BE PLACED"
300 PRINT "ANYWHERE WITHIN THE FILE BUT BEFORE THE LINE READ, THIS LAST"
310 PRINT "LINE MUST BE PLACED AFTER THE DATA LINES."
320 PRINT "   THE SCALE OF THE MAP IS IN INCHES PER DEGREE AT THE"
330 PRINT "EQUATOR. THE LATITUDES AND LONGITUDES ARE IN DECIMAL DEGREES"
340 PRINT
350 PRINT "   THE PROGRAM USE 3 PENS : PEN 1 FOR THE TITLE"
360 PRINT "                            PEN 2 FOR THE SUBTITLE"
370 PRINT "                            PEN 0 FOR THE FRAME, TICK MARKS,"
380 PRINT "                                  LABELS AND COMMENTS"
390 PRINT
400 PRINT "WHAT IS THE NUMBER OF YOUR DATA FILE (IF LOADED TYPE 0) ? ";
410 INPUT W
420 IF W=0 THEN 520
430 PRINT "IS THE TAPE READY (Y/N) ? ";
440 INPUT A$
450 A$=SEG$(A$,1,1)
460 IF A$<>"Y" THEN 430
470 REM
480 REM    LOADING THE DATA FILE
490 REM    -------------------
500 FIND W
510 APPEND 560
520 REM ----------------------------------------------------------------
530 REM
540 REM     PLACE FOR THE DATA FILE
550 REM
560 REM
570 REM ----------------------------------------------------------------
580 REM
590 REM     QUESTIONS
600 REM     =========
```

```
610 REM
620 PRINT "  TITLE OF THE PLOT ? ";
630 INPUT T$
640 PRINT "SUBTITLE ? ";
650 INPUT S$
660 PRINT
670 PRINT "THE TABLE HAS ";A0;" LINES AND ";A1;" COLUMNS."
680 PRINT "HOW MANY DATA SETS DO YOU WANT TO PLOT ? ";
690 INPUT N
700 PRINT "DO YOU WANT TO PRINT A NUMERICAL LABEL NEAR THE POINT"
710 PRINT "(Y/N) ? ";
720 INPUT Q$
730 IF Q$="Y" THEN 760
740 DIM B(N,6)
750 GO TO 770
760 DIM B(N,7)
770 REM
780 REM   QUESTIONS ABOUT THE COLUMNS AND LINES TO PLOT,
790 REM     WHICH SYMBOL AND WHICH COLOR TO USE
800 REM     ----------------------------------------------
810 PRINT
820 FOR I=1 TO N
830 CHARSIZE 3
840 PRINT "DATA SET ";I
850 CHARSIZE 2
860 PRINT "   COLUMN NUMBER FOR LATITUDE ? ";
870 INPUT B(I,1)
880 PRINT "   COLUMN NUMBER FOR LONGITUDE ? ";
890 INPUT B(I,2)
900 IF Q$<>"Y" THEN 930
910 PRINT "   COLUMN NUMBER FOR THE NUMBER ? ";
920 INPUT B(I,7)
930 PRINT "   DO YOU WANT TO USE ALL THE LINES OF THE TABLE (Y/N) ? ";
940 INPUT R$
950 R$=SEG(R$,1,1)
960 IF R$<>"Y" THEN 1000
970 B(I,3)=1
980 B(I,4)=A0
990 GO TO 1040
1000 PRINT "NUMBER OF THE FIRST LINE TO BE PLOTTED ? ";
1010 INPUT B(I,3)
1020 PRINT "NUMBER OF LINES TO PLOT ? ";
1030 INPUT B(I,4)
1040 PRI "CHOOSE AMONG THESE TYPES OF SYMBOL FOR THE POINTS LOCATION :"
1050 PRINT "1 : + ;    2 : * ;    3 : # ;    4 : $ ;    5 : ."
1060 PRINT "6 : 0 ;    7 : S ;    8 : X ;    9 : Z"
1070 INPUT B(I,5)
1080 PRINT "IN WHICH COLOR (0 TO 2) ? ";
1090 INPUT B(I,6)
1100 IF B(I,6)<0 OR B(I,6)>2 THEN 1080
1110 PRINT
1120 NEXT I
1130 REM
1140 REM     SEARCH FOR S & N LAT AND W & E LONG LIMITS AND
1150 REM     THE MIN AND MAX SPEED IN THE DATA TABLE
1160 REM     ----------------------------------------------
1170 DIM X(7),Y(7)
1180 X(1)=A(B(1,3),B(1,1))
1190 X(2)=X(1)
1200 Y(1)=A(B(1,3),B(1,2))
1210 Y(2)=Y(1)
1220 FOR I=1 TO N
1230 FOR J=B(I,3) TO B(I,3)+B(I,4)-1
1240 X(1)=X(1) MIN A(J,B(I,1))
1250 X(2)=X(2) MAX A(J,B(I,1))
1260 Y(1)=Y(1) MIN A(J,B(I,2))
1270 Y(2)=Y(2) MAX A(J,B(I,2))
1280 NEXT J
1290 NEXT I
1300 IF Y(2)-Y(1)<=180 THEN 1340
1310 Y(3)=Y(1)
1320 Y(1)=Y(2)
1330 Y(2)=Y(3)
1340 REM
1350 REM    QUESTIONS FOR THE GRAPH
1360 REM    -----------------------
1370 PRINT "LAT S = ";X(1);"   LAT N = ";X(2)
1380 PRINT "LONG W = ";Y(1);"   LONG E = ";Y(2)
1390 PRINT
1400 REM
1410 REM   LIMITS
1420 REM
1430 PRINT "FOR THE GRAPH : LAT S ? LAT N ? LONG W ? LONG E ? ";
1440 INPUT X(3),X(4),Y(3),Y(4)
1450 IF X(3)<X(4) THEN 1500
1460 X0=X(4)
1470 X(4)=X(3)
1480 X(3)=X0
1490 REM
1500 REM   SCALE
1510 REM
1520 PRINT "WHAT SCALE (IN INCHES PER DEGREE) FOR THE MAP ? ";
1530 INPUT S
1540 Y9=Y(4)-Y(3)
1550 IF Y9>0 THEN 1570
1560 Y9=360+Y9
1570 IF S*Y9<=34 THEN 1620
1580 PRINT "MAP TOO LONG : ";S*Y9;" INCHES INSTEAD OF 34 INCHES"
1590 PRINT "CHANGE THE SCALE OR REDUCE THE LONGITUDE"
1600 PRINT "INTERVAL"
1610 GO TO 1370
1620 IF Y(4)>Y(3) THEN 1660
1630 L9=1
1640 Y(4)=Y(4)+360
1650 GO TO 1670
1660 L9=0
1670 PRINT
1680 REM
1690 REM   GRID LINES
1700 REM
1710 PRINT "DO YOU WANT GRID LINES (Y/N) ? ";
1720 INPUT N$
1730 N$=SEG(N$,1,1)
1740 IF N$<>"Y" THEN 1800
1750 PRINT "INTERVAL IN DECIMAL DEGREES ALONG LAT ? ALONG LONG ? ";
1760 INPUT X(5),Y(5)
1770 REM
1780 REM   TICK MARKS
1790 REM
1800 PRINT "DO YOU WANT TICK MARKS (Y/N) ? ";
```

```
1810 INPUT M$
1820 M$=SEG(M$,1,1)
1830 IF M$<>"Y" THEN 1890
1840 PRINT "INTERVAL IN DECIMAL DEGREES ALONG LAT ? ALONG LONG ? ";
1850 INPUT X(6),Y(6)
1860 REM
1870 REM   LABELS
1880 REM
1890 PRINT "DO YOU WANT LABELS (Y/N) ? ";
1900 INPUT L$
1910 L$=SEG(L$,1,1)
1920 IF L$<>"Y" THEN 1950
1930 PRINT "INTERVAL IN DECIMAL DEGREES ALONG LAT, ALONG LONG ? ";
1940 INPUT X(7),Y(7)
1950 PRINT
1960 REM
1970 REM    LINES OF COMMENTS
1980 REM
1990 PRINT "DO YOU WANT TO PRINT LINES OF COMMENT (UP TO 8) (Y/N) ? ";
2000 INPUT R$
2010 R$=SEG(R$,1,1)
2020 IF R$="Y" THEN 2050
2030 D=0
2040 GO TO 2370
2050 PRINT "     1ST LINE : "
2060 INPUT A$
2070 D=1
2080 PRINT "     2ND LINE (IF NONE PRESS THE SPACE BAR AND RETURN) :"
2090 INPUT B$
2100 K$=" "
2110 IF B$=K$ THEN 2370
2120 D=2
2130 PRINT "     3RD LINE (IF NONE PRESS THE SPACE BAR AND RETURN) :"
2140 INPUT C$
2150 IF C$=K$ THEN 2370
2160 D=3
2170 PRINT "     4TH LINE (IF NONE PRESS THE SPACE BAR AND RETURN) :"
2180 INPUT D$
2190 IF D$=K$ THEN 2370
2200 D=4
2210 PRINT "     5TH LINE (IF NONE PRESS THE SPACE BAR AND RETURN) :"
2220 INPUT E$
2230 IF E$=K$ THEN 2370
2240 D=5
2250 PRINT "     6TH LINE (IF NONE PRESS THE SPACE BAR AND RETURN) :"
2260 INPUT F$
2270 IF F$=K$ THEN 2370
2280 D=6
2290 PRINT "     7TH LINE (IF NONE PRESS THE SPACE BAR AND RETURN) :"
2300 INPUT G$
2310 IF G$=K$ THEN 2370
2320 D=7
2330 PRINT "     8TH & LAST LINE (IF NONE PRESS THE SPACE BAR, RETURN):"
2340 INPUT H$
2350 IF H$=K$ THEN 2370
2360 D=8
2370 PRINT
2380 REM ------------------------------------------------------------
2390 REM
2400 REM     COMPUTATION PART
2410 REM     ================
2420 REM
2430 REM    WGS72 EXCENTRICITY AND DEFINITION OF FUNCTION TO DETERMINE
2440 REM    Y POSITION OF LATITUDES
2450 REM    ------------------------------------------------------------
2460 A0=SQR(0.00669943178)
2470 DEF FNB(X)=A0*SIN(X)
2480 DEF FNA(X)=LOG(TAN(45+X/2))-A0/2*LOG((1+FNB(X))/(1-FNB(X)))
2490 REM
2500 REM    FRAME LIMITS IN CM FOR THE PLOT
2510 REM    -------------------------------
2520 DIM XO(2,2)
2530 FOR I=1 TO 2
2540 XO(2,1)=0
2550 XO(1,1)=(FNA(X(4))-FNA(X(3)))*180/PI*S*2.54
2560 XO(1,2)=0
2570 XO(2,2)=Y9*S*2.54
2580 NEXT I
2590 REM
2600 REM    TABLE OF THE X & Y POSITIONS ON THE PLOT FOR THE GRID LINES
2610 REM    ------------------------------------------------------------
2620 IF N$<>"Y" THEN 2760
2630 X5=INT((X(4)-X(3))/X(5)+1.0E-4)+1
2640 V0=X(4)-(X(4)/X(5)-INT(X(4)/X(5)+1.0E-4))*X(5)
2650 DIM X1(X5)
2660 FOR I=1 TO X5
2670 X1(I)=(FNA(X(4))-FNA(V0-(I-1)*X(5)))*180/PI*S*2.54
2680 NEXT I
2690 Y5=INT((Y(4)-Y(3))/Y(5)+1.0E-4)+1
2700 Y0=Y(3)+(Y(3)/Y(5)-INT(Y(3)/Y(5)+1.0E-4))*Y(5)
2710 DIM Y1(Y5)
2720 FOR I=1 TO Y5
2730 Y1(I)=(Y0+(I-1)*Y(5)-Y(3))*S*2.54
2740 NEXT I
2750 REM
2760 REM    TABLE OF THE X & Y POSITIONS ON THE PLOT FOR THE TICK MARKS
2770 REM    ------------------------------------------------------------
2780 IF M$<>"Y" THEN 2920
2790 X6=INT((X(4)-X(3))/X(6)+1.0E-4)+1
2800 V0=X(4)-(X(4)/X(6)-INT(X(4)/X(6)+1.0E-4))*X(6)
2810 DIM X2(X6)
2820 FOR I=1 TO X6
2830 X2(I)=(FNA(X(4))-FNA(V0-(I-1)*X(6)))*180/PI*S*2.54
2840 NEXT I
2850 Y6=INT((Y(4)-Y(3))/Y(6)+1.0E-4)+1
2860 Y0=Y(3)+(Y(3)/Y(6)-INT(Y(3)/Y(6)+1.0E-4))*Y(6)
2870 DIM Y2(Y6)
2880 FOR I=1 TO Y6
2890 Y2(I)=(Y0+(I-1)*Y(6)-Y(3))*S*2.54
2900 NEXT I
2910 REM
2920 REM    TABLE OF THE X & Y POSITIONS ON THE PLOT FOR THE LABELS
2930 REM    ------------------------------------------------------
2940 IF L$<>"Y" THEN 3120
2950 X7=INT((X(4)-X(3))/X(7)+1.0E-4)+1
2960 V0=X(4)-(X(4)/X(7)-INT(X(4)/X(7)+1.0E-4))*X(7)
2970 DIM X3(X7,2)
2980 FOR I=1 TO X7
2990 X3(I,1)=(FNA(X(4))-FNA(V0-(I-1)*X(7)))*180/PI*S*2.54
3000 X3(I,2)=V0-(I-1)*X(7)
```

```
3010 NEXT I
3020 Y7=INT((Y(4)-Y(3))/Y(7)+1.0E-4)+1
3030 YO=Y(3)+(Y(3)/Y(7)-INT(Y(3)/Y(7)+1.0E-4))*Y(7)
3040 DIM Y3(Y7,2)
3050 FOR I=1 TO Y7
3060 Y3(I,1)=(YO+(I-1)*Y(7)-Y(3))*S*2.54
3070 Y3(I,2)=YO+(I-1)*Y(7)
3080 IF Y3(I,2)<=180 THEN 3100
3090 Y3(I,2)=Y3(I,2)-360
3100 NEXT I
3110 REM ------------------------------------------------------------------
3120 REM
3130 REM     DRAWING PART
3140 REM     ============
3150 REM
3160 PRINT "PUT THE PEN AT THE RIGHT END OF THE PLOTTER"
3170 PRINT "PLOTTER READY (Y,N) ? ";
3180 INPUT R$
3190 R$=SEG(R$,1,1)
3200 IF R$<>"Y" THEN 3170
3210 REM
3220 REM    INITIALIZATION OF THE PLOTTER
3230 REM    ----------------------------
3240 GOSUB 6220
3250 REM
3260 REM    TITLE AND SUBTITLE
3270 REM    ------------------
3280 REM   SETTING CHARACTER TYPE, ORIENTATION AND COLOR
3290 REM
3300 Z4=0.35
3310 Z5=0.4
3320 Z9=90
3330 GOSUB 6260
3340 Z6=1
3350 GOSUB 6270
3360 REM
3370 REM   PRINTING THE TITLE TWICE
3380 REM
3390 Z$=T$
3400 Z3=LEN(Z$)
3410 Z2=0
3420 ZO=0
3430 Z1=0
3440 GOSUB 6230
3450 GOSUB 6250
3460 ZO=ZO+0.02
3470 GOSUB 6230
3480 GOSUB 6250
3490 REM
3500 REM   PRINTING THE SUBTITLE
3510 REM
3520 Z4=0.3
3530 Z5=0.3
3540 GOSUB 6260
3550 ZO=2
3560 Z1=0
3570 GOSUB 6230
3580 Z6=2
3590 GOSUB 6270
3600 Z$=S$
3610 Z3=LEN(Z$)
3620 GOSUB 6250
3630 REM
3640 REM   CHANGING THE COLOR
3650 REM
3660 Z6=0
3670 GOSUB 6270
3680 REM
3690 REM   CHANGING THE ORIGIN
3700 REM
3710 ZO=4.5
3720 Z1=0
3730 GOSUB 6230
3740 ZO=0
3750 Z1=0
3760 GOSUB 6240
3770 REM
3780 REM    DRAWING THE FRAME
3790 REM    -----------------
3800 ZO=XO(1,1)
3810 Z2=1
3820 GOSUB 6230
3830 Z1=XO(2,2)
3840 GOSUB 6230
3850 ZO=0
3860 GOSUB 6230
3870 Z1=0
3880 GOSUB 6230
3890 Z2=0
3900 REM
3910 REM    DRAWING THE GRID LINES
3920 REM    ----------------------
3930 IF N$<>"Y" THEN 4180
3940 ZO=0
3950 Z1=0
3960 GOSUB 6230
3970 FOR I=1 TO X5
3980 IF X1(I)<=0 OR X1(I)=>XO(1,1) THEN 4060
3990 ZO=X1(I)
4000 Z1=0
4010 GOSUB 6230
4020 Z1=XO(2,2)
4030 Z2=1
4040 GOSUB 6230
4050 Z2=0
4060 NEXT I
4070 FOR I=1 TO Y5
4080 IF Y1(I)<=0 OR Y1(I)=>XO(2,2) THEN 4160
4090 ZO=0
4100 Z1=Y1(I)
4110 GOSUB 6230
4120 ZO=XO(1,1)
4130 Z2=1
4140 GOSUB 6230
4150 Z2=0
4160 NEXT I
4170 REM
4180 REM    DRAWING THE TICK MARKS
4190 REM    ----------------------
4200 IF M$<>"Y" THEN 4650
```

```
4210 Z0=0
4220 Z1=0
4230 GOSUB 6230
4240 FOR I=1 TO X6
4250 IF X2(I)<0 OR X2(I)>X0(1,1) THEN 4330
4260 Z0=X2(I)
4270 Z1=0
4280 GOSUB 6230
4290 Z1=Z1-0.3
4300 Z2=1
4310 GOSUB 6230
4320 Z2=0
4330 NEXT I
4340 FOR I=1 TO Y6
4350 IF Y2(I)<0 OR Y2(I)>X0(2,2) THEN 4430
4360 Z0=X0(1,1)
4370 Z1=Y2(I)
4380 GOSUB 6230
4390 Z0=Z0+0.3
4400 Z2=1
4410 GOSUB 6230
4420 Z2=0
4430 NEXT I
4440 FOR I=1 TO Y6
4450 IF Y2(I)<0 OR Y2(I)>X0(2,2) THEN 4530
4460 Z0=0
4470 Z1=Y2(I)
4480 GOSUB 6230
4490 Z0=Z0-0.3
4500 Z2=1
4510 GOSUB 6230
4520 Z2=0
4530 NEXT I
4540 FOR I=1 TO X6
4550 IF X2(I)<0 OR X2(I)>X0(1,1) THEN 4630
4560 Z0=X2(I)
4570 Z1=X0(2,2)
4580 GOSUB 6230
4590 Z1=Z1+0.3
4600 Z2=1
4610 GOSUB 6230
4620 Z2=0
4630 NEXT I
4640 REM
4650 REM    PRINTING THE GRID LABELS
4660 REM    ------------------------
4670 IF L$<>"Y" THEN 5070
4680 FOR I=1 TO X7
4690 IF X3(I,1)<0 OR X3(I,1)>X0(1,1) THEN 4760
4700 Z$=STR(X3(I,2))
4710 Z3=LEN(Z$)
4720 Z0=X3(I,1)
4730 Z1=-0.5-(Z3+1)*Z4
4740 GOSUB 6230
4750 GOSUB 6250
4760 NEXT I
4770 FOR I=1 TO Y7
4780 IF Y3(I,1)<0 OR Y3(I,1)>X0(2,2) THEN 4850
4790 Z0=X0(1,1)+0.8+Z5
4800 Z1=Y3(I,1)-(Z3-1)/2*Z4
4810 GOSUB 6230
4820 Z$=STR(Y3(I,2))
4830 Z3=LEN(Z$)
4840 GOSUB 6250
4850 NEXT I
4860 FOR I=1 TO Y7
4870 IF Y3(I,1)<0 OR Y3(I,1)>X0(2,2) THEN 4940
4880 Z$=STR(Y3(I,2))
4890 Z3=LEN(Z$)
4900 Z0=-0.8
4910 Z1=Y3(I,1)-(Z3-1)/2*Z4
4920 GOSUB 6230
4930 GOSUB 6250
4940 NEXT I
4950 FOR I=1 TO X7
4960 IF X3(I,1)<0 OR X3(I,1)>X0(1,1) THEN 5050
4970 Z$=STR(X3(I,2))
4980 Z3=LEN(Z$)
4990 Z0=X3(I,1)
5000 Z1=X0(2,2)+0.5
5010 GOSUB 6230
5020 Z$=STR(X3(I,2))
5030 Z3=LEN(Z$)
5040 GOSUB 6250
5050 NEXT I
5060 REM
5070 REM    COMPUTATION OF THE POSITION OF EACH DATA POINT AND PLOT
5080 REM    ------------------------------------------------------
5090 Z4=0.2
5100 Z5=0.2
5110 GOSUB 6260
5120 DIM C(1,5)
5130 FOR I=1 TO N
5140 FOR J=B(I,3) TO B(I,3)+B(I,4)-1
5150 C(1,1)=(FNA(X(4))-FNA(A(J,B(I,1))))*180/PI*S*2.54
5160 IF L9=0 OR A(J,B(I,2))=>0 THEN 5180
5170 A(J,B(I,2))=360+A(J,B(I,2))
5180 C(1,2)=(A(J,B(I,2))-Y(3))*S*2.54
5190 C(1,3)=B(I,5)
5200 C(1,4)=B(I,6)
5210 IF Q$<>"Y" THEN 5230
5220 C(1,5)=A(J,B(I,7))
5230 IF C(1,1)<0 OR C(1,1)>X0(1,1) THEN 5400
5240 IF C(1,2)<0 OR C(1,2)>X0(2,2) THEN 5400
5250 Z0=C(1,1)
5260 Z1=C(1,2)
5270 GOSUB 6230
5280 Z6=C(1,4)
5290 GOSUB 6270
5300 GOSUB C(1,3) OF 5460,5490,5520,5550,5580,5610,5640,5670,5700
5310 Z1=Z1-0.1
5320 Z3=LEN(Z$)
5330 GOSUB 6230
5340 GOSUB 6250
5350 IF Q$<>"Y" THEN 5400
5360 Z$=STR(C(1,5))
5370 Z$=" "&Z$
5380 Z3=LEN(Z$)
5390 GOSUB 6250
5400 NEXT J
```

```
5410 NEXT I
5420 GO TO 5730
5430 REM
5440 REM    POINT SYMBOLS
5450 REM    -------------
5460 Z$="+"
5470 ZO=ZO+0.08
5480 RETURN
5490 Z$="*"
5500 ZO=ZO+0.08
5510 RETURN
5520 Z$="#"
5530 ZO=ZO+0.1
3540 RETURN
5550 Z$="$"
5560 ZO=ZO+0.1
5570 RETURN
5580 Z$="."
5590 ZO=ZO+0.04
5600 RETURN
5610 Z$="O"
5620 ZO=ZO+0.1
5630 RETURN
5640 Z$="S"
5650 ZO=ZO+0.1
5660 RETURN
5670 Z$="X"
5680 ZO=ZO+0.1
5690 RETURN
5700 Z$="Z"
5710 ZO=ZO+0.1
5720 RETURN
5730 REM
5740 REM    PRINTING OF THE COMMENTS
5750 REM    ------------------------
5760 ZO=XO(1,1)+1.5
5770 Z1=0
5780 GOSUB 6230
5790 IF D=0 THEN 6130
5800 Z6=0
5810 GOSUB 6270
5820 Z4=0.3
5830 Z5=0.3
5840 GOSUB 6260
5850 ZO=0
5860 Z1=0
5870 GOSUB 6240
5880 FOR I=1 TO D
5890 GOSUB I OF 5960,5980,6000,6020,6040,6060,6080,6100
5900 ZO=ZO+0.5
5910 GOSUB 6230
5920 Z3=LEN(Z$)
5930 GOSUB 6250
5940 NEXT I
5950 GO TO 6130
5960 Z$=A$
5970 RETURN
5980 Z$=B$
5990 RETURN
6000 Z$=C$
6010 RETURN
6020 Z$=D$
6030 RETURN
6040 Z$=E$
6050 RETURN
6060 Z$=F$
6070 RETURN
6080 Z$=G$
6090 RETURN
6100 Z$=H$
6110 RETURN
6120 REM
6130 REM    DISPLAYING THE PLOT
6140 REM    -------------------
6150 ZO=ZO+1
6160 GOSUB 6230
6170 END
6180 REM ------------------------------------------------------------
6190 REM
6200 REM     BENSON PLOTTER SUBROUTINES
6210 REM     ==========================
6220 GO TO 6300
6230 GO TO 6540
6240 GO TO 6830
6250 GO TO 6890
6260 GO TO 6990
6270 GO TO 7130
6280 GO TO 7220
6290 REM
6300 REM    INITIALIZATION OF THE BENSON 1332 PLOTTER
6310 REM    -----------------------------------------
6320 DIM Z(25),Z$(80)
6330 ON SRQ THEN 6520
6340 PRINT "BENSON ADDRESS ? ";
6350 INPUT Z(25)
6360 PRINT @Z(25),"76 1";
6370 Z$=CHR(34)
6380 PRINT @Z(25),Z$;"#$%&'()*+,-./0123456789:;<=>?";
6390 PRINT @Z(25),"@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]↑←";
6400 PRINT @Z(25), USING "A/L",Z$ "
6410 PRINT @Z(25),"76                                    ";
6420 PRINT @Z(25),"                                     ";
6430 PRINT @Z(25), USING "A/L",Z$ "
6440 Z(1)=0
6450 Z(2)=0
6460 Z(3)=200
6470 Z(5)=60
6480 Z(6)=0
6490 Z(13)=0
6500 Z(20)=Z(3)
6510 Z(21)=Z(3)
6520 RETURN
6530 REM
6540 REM    MOVING THE PEN
6550 REM    --------------
6560 Z(7)=INT(ZO*Z(20))
6570 Z(8)=INT(Z1*Z(21))
6580 Z(9)=Z(7)-Z(1)
6590 Z(10)=Z(8)-Z(2)
6600 Z(1)=Z(7)
```

```
6610 Z(2)=Z(8)
6620 Z(7)=ABS(Z(9))
6630 Z(8)=ABS(Z(10))
6640 IF Z(7)>16383 THEN 6730
6650 IF Z(8)>16383 THEN 6730
6660 Z(7)=0
6670 Z(11)=Z(9)
6680 Z(12)=Z(10)
6690 PRINT @Z(25),"PL ";Z(11);Z(12);Z2,
6700 PRINT @Z(25), USING "A/L",," "
6710 IF Z(7)>1 THEN 6620
6720 RETURN
6730 Z(7)=INT((Z(7)+16383)/16384)
6740 Z(8)=INT((Z(8)+16383)/16384)
6750 IF Z(7)>Z(8) THEN 6770
6760 Z(7)=Z(8)
6770 Z(11)=INT(Z(9)/Z(7))
6780 Z(12)=INT(Z(10)/Z(7))
6790 Z(9)=Z(9)-Z(11)
6800 Z(10)=Z(10)-Z(12)
6810 GO TO 6690
6820 REM
6830 REM    CHANGING THE ORIGIN
6840 REM    -------------------
6850 Z(1)=INT(Z0*Z(20))
6860 Z(2)=INT(Z1*Z(21))
6870 RETURN
6880 REM
6890 REM    WRITING A STRING OF CHARACTERS
6900 REM    ------------------------------
6910 PRINT @Z(25),"TX ";Z3;" ";Z$,
6920 PRINT @Z(25), USING "A/L",," "
6930 Z(1)=Z(1)+Z(5)*Z3
6940 Z(2)=Z(2)+Z(6)*Z3
6950 PRINT @Z(25),"NT ";Z(5)*Z3;" ";Z(6)*Z3,
6960 PRINT @Z(25), USING "A/L",," "
6970 RETURN
6980 REM
6990 REM    INITIALIZATION OF CHARACTER TYPE AND ORIENTATION
7000 REM    -----------------------------------------------
7010 Z(5)=Z4*COS(Z9)
7020 Z(6)=Z4*SIN(Z9)
7030 Z(16)=INT(4*Z(5)*Z(20)/3)
7040 Z(17)=INT(-4*Z5*Z(20)*SIN(Z9)/3)
7050 Z(18)=INT(4*Z(6)*Z(21)/3)
7060 Z(19)=INT(4*Z5*Z(21)*COS(Z9)/3)
7070 PRINT @Z(25),"SZ ";Z(16);" ";Z(17);" ";Z(18);" ";Z(19),
7080 PRINT @Z(25), USING "A/L",," "
7090 Z(5)=INT(Z(5)*Z(20))
7100 Z(6)=INT(Z(6)*Z(21))
7110 RETURN
7120 REM
7130 REM    CHANGING THE PEN
7140 REM    ----------------
7150 Z(14)=INT(ABS(Z6))-3*INT(INT(ABS(Z6))/3)
7160 IF Z(14)=Z(13) THEN 7200
7170 PRINT @Z(25),"NP ";Z(14),
7180 PRINT @Z(25), USING "A/L",," "
7190 Z(13)=Z(14)
7200 RETURN
7210 REM
7220 REM    PRINTING OF NUMBERS
7230 REM    -------------------
7240 Z7=INT(Z7)
7250 Z(9)=ABS(INT(Z7))
7260 IF Z7>-2 THEN 7440
7270 IF Z(9)<10 THEN 7290
7280 Z(9)=9
7290 Z$=STR(Z(9))
7300 Z$=Z$&"E"
7310 Z(9)=Z(9)+7
7320 IF Z8>0 THEN 7360
7330 Z(9)=Z(9)+1
7340 PRINT @Z(25),"TX ";Z(9)," ",
7350 GO TO 7370
7360 PRINT @Z(25),"TX ";Z(9),
7370 PRINT @Z(25), USING Z$,Z8,
7380 PRINT @Z(25), USING "A/L",," "
7390 Z(1)=Z(1)+Z(9)*Z(5)
7400 Z(2)=Z(2)+Z(9)*Z(6)
7410 PRINT @Z(25),"NT ";Z(9)*Z(5)," ";Z(9)*Z(6),
7420 PRINT @Z(25), USING "A/L",," "
7430 RETURN
7440 Z(7)=INT(LGT(ABS(Z8)+0.5))+1
7450 IF Z(7)>0 THEN 7470
7460 Z(7)=1
7470 IF Z8=>0 THEN 7490
7480 Z(7)=Z(7)+1
7490 Z$=STR(Z(7))
7500 Z$=Z$&"D"
7510 IF Z7>-1 THEN 7540
7520 Z(9)=Z(7)
7530 GO TO 7340
7540 Z(8)=LEN(Z$)+2
7550 Z$=Z$&". 0"
7560 IF Z(9)<10 THEN 7580
7570 Z(9)=9
7580 GO TO Z(9)+1 OF 7590,7610,7630,7650,7670,7690,7710,7730,7750,7770
7590 Z$=REP(" ",Z(8)+1,1)
7600 GO TO 7780
7610 Z$=REP("1",Z(8),1)
7620 GO TO 7780
7630 Z$=REP("2",Z(8),1)
7640 GO TO 7780
7650 Z$=REP("3",Z(8),1)
7660 GO TO 7780
7670 Z$=REP("4",Z(8),1)
7680 GO TO 7780
7690 Z$=REP("5",Z(8),1)
7700 GO TO 7780
7710 Z$=REP("6",Z(8),1)
7720 GO TO 7780
7730 Z$=REP("7",Z(8),1)
7740 GO TO 7780
7750 Z$=REP("8",Z(8),1)
7760 GO TO 7780
7770 Z$=REP("9",Z(8),1)
7780 Z(9)=Z(9)+Z(7)+1
7790 GO TO 7340
```

C
★

MERCATOR PROJECTION MAP AND PLOTTING OF POINTS

PROGRAM TO PLOT POINTS ON A MERCATOR PROJECTION MAP (WGS72 ELLISOID) USING UP TO 9 SYMBOLS AND 3 COLORS. THE POINTS MAY BE LABELLED BUT ONLY WITH NUMBERS.

THE DATA TABLE HAS BEEN CREATED IN ANOTHER FILE AND NAMED A. THIS DATA FILE MUST HAVE AT LEAST 2 COLUMNS : TWO FOR THE LATITUDE AND LONGITUDE OF THE POINTS, A THIRD ONE MAY CONTAIN A NUMBER WHICH MAY BE PRINTED ON THE MAP.

THIS FILE MUST ALSO HAVE TWO VARIABLES, A0 AND A1 WHICH CONTAIN THE NUMBER OF LINES OF THE TABLE (A0) AND ITS NUMBER OF COLUMNS (A1). THE FILE MUST ALSO HAVE TWO LINES WHICH ARE : DIM A(A0,A1), AND READ A. THE FIRST LINE CAN BE PLACED ANYWHERE WITHIN THE FILE BUT BEFORE THE LINE READ, THIS LAST LINE MUST BE PLACED AFTER THE DATA LINES.

THE SCALE OF THE MAP IS IN INCHES PER DEGREE AT THE EQUATOR. THE LATITUDES AND LONGITUDES ARE IN DECIMAL DEGREES

THE PROGRAM USES 3 PENS : PEN 1 FOR THE TITLE
PEN 2 FOR THE SUBTITLE
PEN 0 FOR THE FRAME, TICK MARKS, LABELS AND COMMENTS

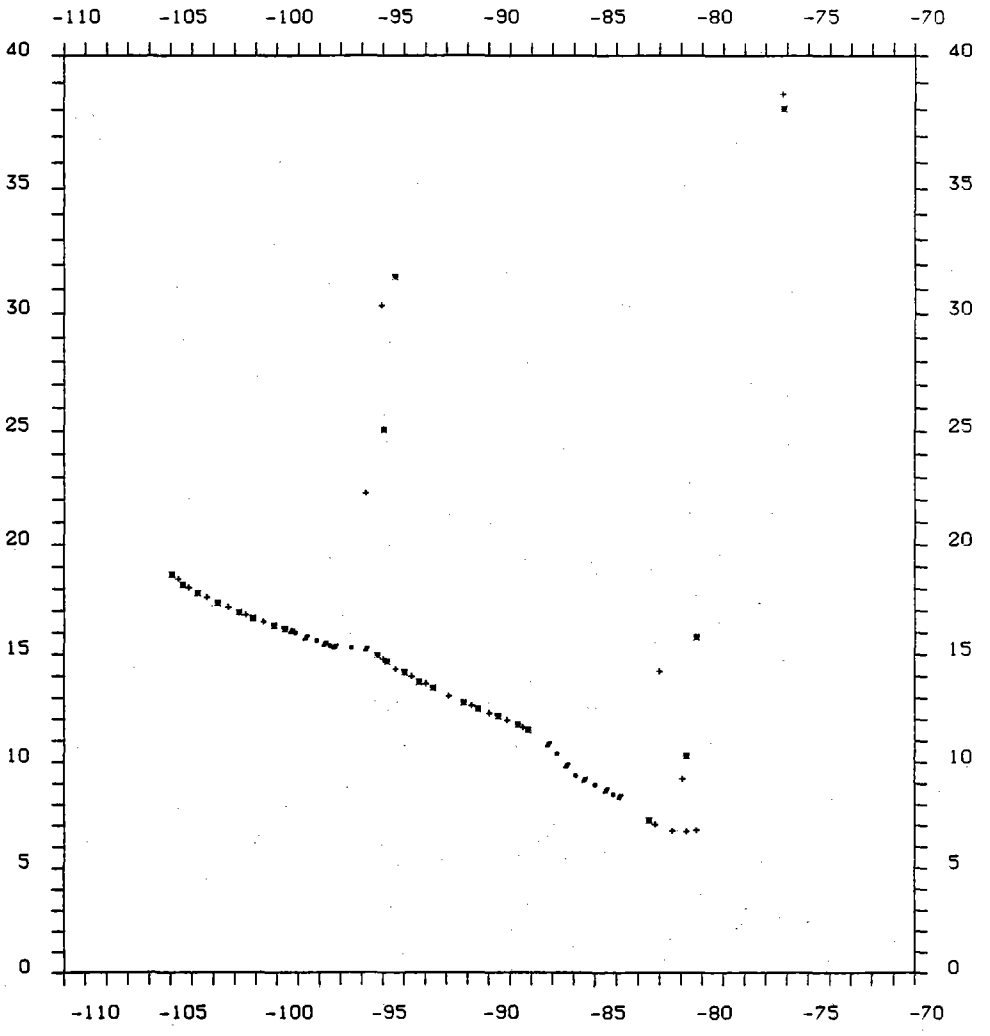
```
WHAT IS THE NUMBER OF YOUR DATA FILE (IF LOADED TYPE 0) ? :15
IS THE TAPE READY (Y/N) ? Y
  TITLE OF THE PLOT ? MIDDLE AMERICA TRENCH
SUBTITLE ? CIRCLES

THE TABLE HAS 66 LINES AND 2 COLUMNS.
HOW MANY DATA SETS DO YOU WANT TO PLOT ? 12
DO YOU WANT TO PRINT A NUMERICAL LABEL NEAR THE POINT
(Y/N) ? N

DATA SET 1
   COLUMN NUMBER FOR LATITUDE ? 1
   COLUMN NUMBER FOR LONGITUDE ? 2
   DO YOU WANT TO USE ALL THE LINES OF THE TABLE (Y/N) ? N
NUMBER OF THE FIRST LINE TO BE PLOTTED ? 1
NUMBER OF LINES TO PLOT ? 6
CHOOSE AMONG THESE TYPES OF SYMBOL FOR THE POINTS LOCATION :
1 : + ;    2 : # ;    3 : @ ;    4 : & ;    5 : .
6 : O ;    7 : S ;    8 : X ;    9 : Z
1
IN WHICH COLOR (0 TO 2) ? 0
DATA SET 2
   COLUMN NUMBER FOR LATITUDE ? 1
   COLUMN NUMBER FOR LONGITUDE ? 2
   DO YOU WANT TO USE ALL THE LINES OF THE TABLE (Y/N) ? N
NUMBER OF THE FIRST LINE TO BE PLOTTED ? 7
NUMBER OF LINES TO PLOT ? 4
CHOOSE AMONG THESE TYPES OF SYMBOL FOR THE POINTS LOCATION :
1 : + ;    2 : # ;    3 : @ ;    4 : & ;    5 : .
6 : O ;    7 : S ;    8 : X ;    9 : Z
5
IN WHICH COLOR (0 TO 2) ? 0
DATA SET 3
   COLUMN NUMBER FOR LATITUDE ? 1
   COLUMN NUMBER FOR LONGITUDE ? 2
   DO YOU WANT TO USE ALL THE LINES OF THE TABLE (Y/N) ? N
NUMBER OF THE FIRST LINE TO BE PLOTTED ? 11
NUMBER OF LINES TO PLOT ? 8
CHOOSE AMONG THESE TYPES OF SYMBOL FOR THE POINTS LOCATION :
1 : + ;    2 : # ;    3 : @ ;    4 : & ;    5 : .
6 : O ;    7 : S ;    8 : X ;    9 : Z
1
IN WHICH COLOR (0 TO 2) ? 0
DATA SET 4
   COLUMN NUMBER FOR LATITUDE ? 1
   COLUMN NUMBER FOR LONGITUDE ? 2
   DO YOU WANT TO USE ALL THE LINES OF THE TABLE (Y/N) ? N
NUMBER OF THE FIRST LINE TO BE PLOTTED ? 20
NUMBER OF LINES TO PLOT ? 4
CHOOSE AMONG THESE TYPES OF SYMBOL FOR THE POINTS LOCATION :
1 : + ;    2 : # ;    3 : @ ;    4 : & ;    5 : .
6 : O ;    7 : S ;    8 : X ;    9 : Z
5
IN WHICH COLOR (0 TO 2) ? 0
DATA SET 5
   COLUMN NUMBER FOR LATITUDE ? 1
   COLUMN NUMBER FOR LONGITUDE ? 2
   DO YOU WANT TO USE ALL THE LINES OF THE TABLE (Y/N) ? N
NUMBER OF THE FIRST LINE TO BE PLOTTED ? 24
NUMBER OF LINES TO PLOT ? 4
CHOOSE AMONG THESE TYPES OF SYMBOL FOR THE POINTS LOCATION :
1 : + ;    2 : # ;    3 : @ ;    4 : & ;    5 : .
6 : O ;    7 : S ;    8 : X ;    9 : Z
1
IN WHICH COLOR (0 TO 2) ? 0
DATA SET 6
   COLUMN NUMBER FOR LATITUDE ? 1
   COLUMN NUMBER FOR LONGITUDE ? 2
   DO YOU WANT TO USE ALL THE LINES OF THE TABLE (Y/N) ? N
NUMBER OF THE FIRST LINE TO BE PLOTTED ? 28
NUMBER OF LINES TO PLOT ? 8
CHOOSE AMONG THESE TYPES OF SYMBOL FOR THE POINTS LOCATION :
1 : + ;    2 : # ;    3 : @ ;    4 : & ;    5 : .
6 : O ;    7 : S ;    8 : X ;    9 : Z
2
IN WHICH COLOR (0 TO 2) ? 0
DATA SET 7
   COLUMN NUMBER FOR LATITUDE ? 1
   COLUMN NUMBER FOR LONGITUDE ? 2
   DO YOU WANT TO USE ALL THE LINES OF THE TABLE (Y/N) ? N
NUMBER OF THE FIRST LINE TO BE PLOTTED ? 36
NUMBER OF LINES TO PLOT ? 6
CHOOSE AMONG THESE TYPES OF SYMBOL FOR THE POINTS LOCATION :
1 : + ;    2 : # ;    3 : @ ;    4 : & ;    5 : .
6 : O ;    7 : S ;    8 : X ;    9 : Z
3
IN WHICH COLOR (0 TO 2) ? 0
DATA SET 8
   COLUMN NUMBER FOR LATITUDE ? 1
   COLUMN NUMBER FOR LONGITUDE ? 2
   DO YOU WANT TO USE ALL THE LINES OF THE TABLE (Y/N) ? N
NUMBER OF THE FIRST LINE TO BE PLOTTED ? 41
NUMBER OF LINES TO PLOT ? 10
CHOOSE AMONG THESE TYPES OF SYMBOL FOR THE POINTS LOCATION :
1 : + ;    2 : # ;    3 : @ ;    4 : & ;    5 : .
6 : O ;    7 : S ;    8 : X ;    9 : Z
2
IN WHICH COLOR (0 TO 2) ? 0
DATA SET 9
   COLUMN NUMBER FOR LATITUDE ? 1
   COLUMN NUMBER FOR LONGITUDE ? 2
   DO YOU WANT TO USE ALL THE LINES OF THE TABLE (Y/N) ? N
NUMBER OF THE FIRST LINE TO BE PLOTTED ? 51
NUMBER OF LINES TO PLOT ? 5
CHOOSE AMONG THESE TYPES OF SYMBOL FOR THE POINTS LOCATION :
1 : + ;    2 : # ;    3 : @ ;    4 : & ;    5 : .
6 : O ;    7 : S ;    8 : X ;    9 : Z
3
IN WHICH COLOR (0 TO 2) ? 0
DATA SET 10
   COLUMN NUMBER FOR LATITUDE ? 1
   COLUMN NUMBER FOR LONGITUDE ? 2
   DO YOU WANT TO USE ALL THE LINES OF THE TABLE (Y/N) ? N
NUMBER OF THE FIRST LINE TO BE PLOTTED ? 56
NUMBER OF LINES TO PLOT ? 1
CHOOSE AMONG THESE TYPES OF SYMBOL FOR THE POINTS LOCATION :
1 : + ;    2 : # ;    3 : @ ;    4 : & ;    5 : .
6 : O ;    7 : S ;    8 : X ;    9 : Z
2
IN WHICH COLOR (0 TO 2) ? 0
```

```
DATA SET 11
   COLUMN NUMBER FOR LATITUDE ? 1
   COLUMN NUMBER FOR LONGITUDE ? 2
   DO YOU WANT TO USE ALL THE LINES OF THE TABLE (Y/N) ? N
NUMBER OF THE FIRST LINE TO BE PLOTTED ? 57
NUMBER OF LINES TO PLOT ? 5
CHOOSE AMONG THESE TYPES OF SYMBOL FOR THE POINTS LOCATION :
1 : + ;    2 : * ;    3 : # ;    4 : @ ;    5 : .
6 : O ;    7 : S ;    8 : X ;    9 : Z
1
IN WHICH COLOR (0 TO 2) ? 0
DATA SET 12
   COLUMN NUMBER FOR LATITUDE ? 1
   COLUMN NUMBER FOR LONGITUDE ? 2
   DO YOU WANT TO USE ALL THE LINES OF THE TABLE (Y/N) ? N
NUMBER OF THE FIRST LINE TO BE PLOTTED ? 62
NUMBER OF LINES TO PLOT ? 5
CHOOSE AMONG THESE TYPES OF SYMBOL FOR THE POINTS LOCATION :
1 : + ;    2 : * ;    3 : # ;    4 : @ ;    5 : .
6 : O ;    7 : S ;    8 : X ;    9 : Z
2
IN WHICH COLOR (0 TO 2) ? 0

LAT S = 6.75   LAT N = 30.58
LONG W = -104.97   LONG E = -76.15

FOR THE GRAPH : LAT S ? LAT N ? LONG W ? LONG E ? 0,40,-110,-70
WHAT SCALE (IN INCHES PER DEGREE) FOR THE MAP ? .2

DO YOU WANT GRID LINES (Y/N) ? N
DO YOU WANT TICK MARKS (Y/N) ? Y
INTERVAL IN DECIMAL DEGREES ALONG LAT ? ALONG LONG ? 1,1
DO YOU WANT LABELS (Y/N) ? Y
INTERVAL IN DECIMAL DEGREES ALONG LAT, ALONG LONG ? 5,5

DO YOU WANT TO PRINT LINES OF COMMENT (UP TO 8) (Y/N) ? N

PUT THE PEN AT THE RIGHT END OF THE PLOTTER
PLOTTER READY (Y,N) ? Y
BENSON ADDRESS ? 41
```

D

MIDDLE AMERICA TRENCH

CIRCLES



9.2

A

FICHIER DE DONNEES : MIDAMEPL.DAT

```
  64  4
 18.678 -104.963  5.2596 39.9337
 18.373 -104.555  5.3944 40.0197
 17.767 -103.733  5.6633 40.1628
 17.372 -102.778  5.9206 39.6980
 16.965 -101.892  6.1658 39.3860
 16.645 -101.195  6.3579 39.1605
 16.340 -100.167  6.6126 38.5584
 16.038  -99.358  6.8228 38.2307
 15.773  -98.655  7.0053 37.9698
 15.495  -97.793  7.2207 37.5956
 15.293  -97.230  7.3642 37.3986
 15.218  -95.813  7.6572 36.2806
 15.007  -95.333  7.7845 36.2091
 14.702  -94.843  7.9274 36.2717
 14.168  -93.995  8.1741 36.3820
 13.817  -93.317  8.3586 36.3596
 13.483  -92.667  8.5341 36.3356
 12.812  -91.240  8.9064 36.2070
 12.505  -90.542  9.0835 36.1222
 12.167  -89.583  9.3131 35.9122
 11.777  -88.647  9.5434 35.7890
 11.555  -88.167  9.6636 35.7532
 10.833  -87.218  9.9374 35.9912
  9.815  -86.333 10.2359 36.5472
  9.150  -85.500 10.4738 36.7382
  8.655  -84.425 10.7281 36.6202
  8.355  -83.868 10.8640 36.6007
  8.178  -83.377 10.9716 36.5037
  7.833  -82.992 11.0817 36.6133
  7.317  -82.833 11.1747 36.9849
 15.007  -95.333  6.1137 27.3836
.
.
  7.317  -82.833  9.9170 31.1104
 19.700  -77.000  1.9658 83.2951
 19.200  -80.000  1.9636 80.1974
 18.900  -81.200  1.9614 78.9481
 18.167  -81.667  1.9454 78.3701
 17.500  -83.000  1.9356 76.9160
 16.900  -85.000  1.9327 74.7981
 17.000  -85.600  1.9391 74.2223
 16.800  -85.900  1.9362 73.8837
 16.400  -87.000  1.9341 72.7120
 15.467  -88.683  1.9243 70.8439
 15.683  -89.000  1.9322 70.5918
 15.292  -89.933  1.9306 69.5928
 14.833  -90.000  1.9200 69.4043
 14.933  -91.000  1.9314 68.4785
 15.400  -91.633  1.9484 68.0143
 15.400  -92.800  1.9593 66.9329
```

B

```
10 ' ************************************************************************
20 ' *                                                                      *
30 ' *                        PROGRAM 'MERVECTR.BAS'                        *
40 ' *                                                                      *
50 ' *   adaptation of 'MERVECTOR' on Tektronix for a Benson 1332 plotter,  *
60 ' *   made in Oct. 1984                      Jacqueline ROUMP, May 1987  *
70 ' ************************************************************************
80 '
90 KEY OFF : WIDTH 80 : SCREEN 0,0,0
100 '
110 '          PRESENTATION
120 '          ============
130 '
140 PRINT : PRINT "               MERCATOR PROJECTION MAP FOR VECTOR DRAWING"
150 PRINT : PRINT
160 PRINT "    Program to draw vectors on a Mercator projection map (WGS72"
170 PRINT "ellipsoid)."
180 PRINT
190 PRINT "     The data file can be created and must be corrected using a"
200 PRINT "text editor. The first line has the number of lines, followed by"
210 PRINT "the number of columns. The subsequent lines have the LAT and LONG"
220 PRINT "of the data point followed by the direction of the vector and its"
230 PRINT "magnitude."
240 PRINT
250 PRINT "    The plot will be drawn on a HP 7475A compatible plotter in the"
260 PRINT "21*29.7 cm format."
270 PRINT : PRINT
280 INPUT "Do you want to create a new data set (1) or to use an old data file
 (2) : ",ANS
290   IF ANS<>1 AND ANS<>2 THEN BEEP : GOTO 280
300   IF ANS=2 THEN 730
310 '
320 '     Data input from the Keyboard
330 '     ----------------------------
340 '
350 PRINT : PRINT : PRINT "DATA INPUT" : PRINT
360 '
370 '     Data input from the Keyboard
380 '     ----------------------------
390 '
400 INPUT "Number of lines : ",LN
410 INPUT "Number of columns (at least 4 for lat, long, dir and length) : ",CN
420   IF CN<4 THEN BEEP : GOTO 410
430 DIM DATASET(LN,CN)                'data table
440 FOR I=1 TO LN
450     PRINT
460     PRINT "Column";I
470     FOR J=1 TO CN
480         PRINT "       line ";J;"  :";
490         INPUT ,DATASET(J,I)
500     NEXT J
510 NEXT I
520 PRINT
530 PRINT "Name of the data files on the default disk : " : PRINT
540 FILES "*.DAT"
550 PRINT : INPUT "Name of your data file : ",FILNAME$
```

```
560 OPEN "O",£1,FILNAME$
570 PRINT £1,LN,CN
580 FOR I=1 TO LN
590     A$=""
600     FOR J=1 TO CN
610         LENA=LEN(A$)
620         PLU$=STR$(DATASET(I,J))
630         LENP=LEN(PLU$)
640           IF LENA+LENP<125 THEN A$=A$+PLU$ : GOTO 670
650         PRINT £1,A$
660         A$=PLU$
670     NEXT J
680       IF LEN(A$)<>0 THEN PRINT £1,A$
690 NEXT I
700 CLOSE
710 GOTO 920
720 '
730 '     Data input from a file
740 '     ----------------------
750 '
760 PRINT
770 PRINT "Name of the data files on the default disk : " : PRINT
780 FILES "*.DAT"
790 PRINT : INPUT "Name of your data file : ",FILNAME$
800 OPEN "I",£1,FILNAME$
810 INPUT £1,LN,CN
820 DIM DATASET(LN,CN)                'data table
830 FOR I=1 TO LN
840     FOR J=1 TO CN
850         INPUT £1,DATASET(I,J)
860     NEXT J
870 NEXT I
880 CLOSE
890 '
900 ' ------------------------------------------------------------------------
910 '
920 '          ADDITIONAL QUESTIONS
930 '          ====================
940 '
950 PRINT
960 INPUT "Title of the plot : ",TIT$
970 INPUT "Subtitle : ",SUBTIT$
980 PRINT
990 PRINT "This table has ";LN;" lines and ";CN;" columns."
1000 PRINT
1010 INPUT "Number of X-Y data sets to plot : ",NSET
1020 PRINT : PRINT
1030 PRINT "You will need 3 pens on the plotter, all black or in color :"
1040 PRINT "      Pen 1 is for the axes and the comments"
1050 PRINT "      Pen 2 is for the title"
1060 PRINT "      Pen 3 is for the subtitle"
1070 '
1080 '     Questions about the columns and the lines to plot,
1090 '     which kind of line to use and which color
1100 '     -----------------------------------------------
1110 '
1120 DIM LINPAR(NSET,10)              'parameter table
1130 FOR I=1 TO NSET
1140     PRINT : PRINT "DATA SET ";I
1150     INPUT "     Column number for latitude : ",LINPAR(I,1)
1160     INPUT "     Column number for longitude : ",LINPAR(I,2)
1170     INPUT "     Column number for vector length : ",LINPAR(I,3)
1180     INPUT "     Column number for vector direction : ",LINPAR(I,4)
1190     INPUT "     Do you want to use all the lines of the table (Y/N) : ",A
N$
1200     AN$=LEFT$(AN$,1)
1210       IF AN$<>"Y" AND AN$<>"y" AND AN$<>"N" AND AN$<>"n" THEN 1190
1220       IF AN$="Y" OR AN$="y" THEN LINPAR(I,5)=1 : LINPAR(I,6)=LN : GOTO 12
50
1230     INPUT "     Number of the first line to plot : ",LINPAR(I,5)
1240     INPUT "     Number of lines to be used : ",LINPAR(I,6)
1250     PRINT "Choose among these types of dots for the points location :"
1260     PRINT "1 : + ;    2 : * ;    3 : £  ;   4 : $ ;    5 : ."
1270     PRINT "6 : o ;    7 : s ;    8 : x  ;   9 : z"
1280     PRINT "10 : no sign"
1290     INPUT  LINPAR(I,7)
1300       IF LINPAR(I,7)<>10 THEN 1320
1310     LINPAR(I,9)=0 : GOTO 1340
1320     INPUT "       In which color (1 to 6) : ",LINPAR(I,9)
1330       IF LINPAR(I,9)<1 OR LINPAR(I,9)>6 THEN BEEP : GOTO 1320
1340     PRINT "     Choose among these 6 types of line drawing : "
1350     PRINT "           1 : full line"
1360     PRINT "           2 : very short dashes (0.8 mm)"
1370     PRINT "           3 : short dashes (1.2 mm)"
1380     PRINT "           4 : medium dashes (1.6 mm)"
1390     PRINT "           5 : long dashes (2.1 mm)"
1400     PRINT "           6 : very long dashes (2.8 mm)"
1410     INPUT LINPAR(I,8)
1420       IF LINPAR(I,8)<0 OR LINPAR(I,8)>6 THEN BEEP : GOTO 1340
1430     INPUT "     In which color (1 to 6) : ",LINPAR(I,10)
1440       IF LINPAR(I,10)<1 OR LINPAR(I,10)>6 THEN BEEP : GOTO 1430
1450 NEXT I
1460 '
1470 '     Search for the S & N Lat and W & E Long limits and
1480 '     the min and max vector length in the data table
1490 '     ----------------------------------------------
1500 '
1510 LAMIN=DATASET(LINPAR(1,5),LINPAR(1,1)) : LAMAX=LAMIN
1520 LOMIN=DATASET(LINPAR(1,5),LINPAR(1,2)) : LOMAX=LOMIN
1530 VMIN=DATASET(LINPAR(1,5),LINPAR(1,3)) : VMAX=VMIN
1540 FOR I=1 TO NSET
1550     FOR J=LINPAR(I,5) TO LINPAR(I,5)+LINPAR(I,6)-1
1560         IF DATASET(J,LINPAR(I,1))<LAMIN THEN LAMIN=DATASET(J,LINPAR(I,1))
 : GOTO 1580
1570         IF DATASET(J,LINPAR(I,1))>LAMAX THEN LAMAX=DATASET(J,LINPAR(I,1))
1580         IF DATASET(J,LINPAR(I,2))<LOMIN THEN LOMIN=DATASET(J,LINPAR(I,2))
 : GOTO 1600
1590         IF DATASET(J,LINPAR(I,2))>LOMAX THEN LOMAX=DATASET(J,LINPAR(I,2))
1600         IF DATASET(J,LINPAR(I,3))<VMIN THEN VMIN=DATASET(J,LINPAR(I,3)) :
 GOTO 1620
1610         IF DATASET(J,LINPAR(I,3))>VMAX THEN VMAX=DATASET(J,LINPAR(I,3))
1620     NEXT J
1630 NEXT I
1640   IF LOMAX-LOMIN<=180 THEN 1660
1650 SWAP YMIN,YMAX
1660 '
```

```
1670 '     Questions for the graph
1680 '     -----------------------
1690 '
1700 '     maximum and minimum values
1710 '     ~~~~~~~~~~~~~~~~~~~~~~~~~~
1720 PRINT
1730 PRINT "Lat S = ";LAMIN;"     Lat N = ";LAMAX
1740 PRINT "Long W = ";LOMIN;"     Long E = ";LOMAX
1750 PRINT
1760 INPUT "     For the graph      Lat S : ",LAMIN
1770 INPUT "                        Lat N : ",LAMAX
1780 INPUT "                        Long W : ",LOMIN
1790 INPUT "                        Long E : ",LOMAX
1800 PRINT
1810   IF LAMAX<LAMIN THEN SWAP LAMIN,LAMAX
1820 '
1830 '     vector length
1840 '     ~~~~~~~~~~~~~
1850 PRINT "Min vector length : ";VMIN;"   Max vector length : ";VMAX
1860 INPUT "What length in cm for a vector one unit long : ",VLEN
1870 '
1880 '     lines of comments
1890 '     ~~~~~~~~~~~~~~~~~
1900 PRINT : PRINT
1910 INPUT "Do you want to print some lines of comments (up to 10 ) (Y/N) : ",
AN$
1920 AN$=LEFT$(AN$,1)
1930   IF AN$<>"Y" AND AN$<>"y" AND AN$<>"N" AND AN$<>"n" THEN 1910
1940   IF AN$="N" OR AN$="n" THEN NCOM=0 : GOTO 2050
1950 PRINT "     1st line : " : INPUT COMM$(1) : NCOM=1
1960 PRINT "     2nd line (if no more line press only RETURN)"
1970 INPUT COMM$(2) : IF COMM$(2)="" THEN 2050 ELSE NCOM=2
1980 PRINT "     3rd line (if no more line press only RETURN)"
1990 INPUT COMM$(3) : IF COMM$(3)="" THEN 2050 ELSE NCOM=3
2000 FOR I=4 TO 10
2010     PRINT "    ";I;"th line (if no more line press only RETURN)"
2020     INPUT COMM$(I) : IF COMM$(I)="" THEN 2050 ELSE NCOM=I
2030 NEXT I
2040 '
2050 '     scale
2060 '     ~~~~~
2070 PRINT : PRINT
2080 INPUT "What scale (in inches per degree) for the map : ",SCALE
2090 '
2100 '     WGS72 excentricity and definition of function to determine
2110 '     Y position of latitudes
2120 '     ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
2130 FEXCEN#=SQR(.00666943178#)
2140 FPI#=3.1415926535898887#/180
2150 DEF FNB(X)=FEXCEN#*SIN(X*FPI#)
2160 DEF FNA(X)=LOG(TAN(45*FPI#+X*FPI#/2))-FEXCEN#/2*LOG((1+FNB(X))/(1-FNB(X))
)
2170 '
2180 '     frame limits in cm for the plot
2190 '     ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
2200 DELTLONG=LOMAX-LOMIN
2210   IF DELTLONG<0 THEN DELTLONG=DELTLONG+360
2220 X0(1,1)=(FNA(LAMAX)-FNA(LAMIN))/FPI#*SCALE*2.54
2230 X0(2,2)=DELTLONG*SCALE*2.54
2240 LHEIGHT=X0(1,1)+5.5+NCOM
2250 LWIDTH=X0(2,2)+2
2260 LQUOT=LHEIGHT/LWIDTH
2270   IF LHEIGHT=<27 AND LWIDTH=<40 THEN 2350
2280   IF LQUOT>27/40 THEN SWAP LHEIGHT,LWIDTH : ROT=1
2290   IF LHEIGHT>27 THEN PRINT "The height of the map is too large : ";LHEIGH
T;" instead of 27 cm ; " : PRINT "    you will have to save on the scale and/o
r on the comment lines"
2300   IF LWIDTH>40 THEN PRINT "The width of the map is too large : ";LWIDTH;"
 instead of 40 cm ;" : PRINT "    you will have to save on the scale"
2310 ROT=0 : INPUT "Do you want to save only on the scale (1) or also on the c
omments (2) : ",ANS
2320   IF ANS<>1 AND ANS<>2 THEN BEEP : GOTO 2310
2330   IF ANS=1 THEN 2050 ELSE GOTO 1880
2340 '
2350 '     grid lines
2360 '     ~~~~~~~~~~
2370 PRINT : PRINT
2380 INPUT "Do you want grid lines (Y/N) : ",GRID$
2390 GRID$=LEFT$(GRID$,1)
2400   IF GRID$<>"Y" AND GRID$<>"y" AND GRID$<>"N" AND GRID$<>"n" THEN BEEP :
GOTO 2380
2410   IF GRID$<>"Y" AND GRID$<>"y" THEN 2450
2420 INPUT "  Interval in decimal degrees along the latitudes : ",GINLAT
2430 INPUT "  Interval in decimal degrees along the longitudes : ",GINLONG
2440 '
2450 '     tick marks
2460 '     ~~~~~~~~~~
2470 PRINT
2480 INPUT "Do you want tick marks (Y/N) : ",TICK$
2490 TICK$=LEFT$(TICK$,1)
2500   IF TICK$<>"Y" AND TICK$<>"y" AND TICK$<>"N" AND TICK$<>"n" THEN BEEP :
GOTO 2480
2510   IF TICK$<>"Y" AND TICK$<>"y" THEN 2550
2520 INPUT "  Interval in decimal degrees along the latitudes : ",TINLAT
2530 INPUT "  Interval in decimal degrees along the longitudes : ",TINLONG
2540 '
2550 '     labels
2560 '     ~~~~~~
2570 PRINT
2580 INPUT "Do you want labels (Y/N) : ",LABEL$
2590 LABEL$=LEFT$(LABEL$,1)
2600   IF LABEL$<>"Y" AND LABEL$<>"y" AND LABEL$<>"N" AND LABEL$<>"n" THEN BEE
P : GOTO 2580
2610   IF LABEL$<>"Y" AND LABEL$<>"y" THEN 2670
2620 INPUT "  Interval in decimal degrees along the latitudes : ",LINLAT
2630 INPUT "    number of decimal characters : ",LINLATDEC
2640 INPUT "  Interval in decimal degrees along the longitudes : ",LINLONG
2650 INPUT "    number of decimal characters : ",LINLONGDEC
2660 '
2670 ' ------------------------------------------------------------------------
2680 '
2690 '          COMPUTATION PART
2700 '          ================
2710 '
2720 '     Tables of the X & Y positions on the map for the grid lines
2730 '     -----------------------------------------------------------
```

```
2740 '
2750   IF GRID$()"Y" AND GRID$()"y" THEN 2890
2760 GLAN=INT((LAMAX-LAMIN)/GINLAT+.0001)+1
2770 GLAFST=LAMIN+(LAMIN/GINLAT-INT(LAMIN/GINLAT+.0001))*GINLAT
2780 DIM GLAT(GLAN)
2790 FOR I=1 TO GLAN
2800     GLAT(I)=(-FNA(LAMIN)+FNA(GLAFST+(I-1)*GINLAT))/FPIE*SCALE*2.54
2810 NEXT I
2820 GLON=INT((LOMAX-LOMIN)/GINLONG+.0001)+1
2830 GLOFST=LOMIN+(LOMIN/GINLONG-INT(LOMIN/GINLONG+.0001))*GINLONG
2840 DIM GLONG(GLON)
2850 FOR I=1 TO GLON
2860     GLONG(I)=(GLOFST+(I-1)*GINLONG-LOMIN)*SCALE*2.54
2870 NEXT I
2880 '
2890 '     Tables of the X & Y positions on the map for the tick marks
2900 '     ------------------------------------------------------------
2910 '
2920   IF TICK$()"Y" AND TICK$()"y" THEN 3060
2930 TLAN=INT((LAMAX-LAMIN)/TINLAT+.0001)+1
2940 TLAFST=LAMIN+(LAMIN/TINLAT-INT(LAMIN/TINLAT+.0001))*TINLAT
2950 DIM TLAT(TLAN)
2960 FOR I=1 TO TLAN
2970     TLAT(I)=(-FNA(LAMIN)+FNA(TLAFST+(I-1)*TINLAT))/FPIE*SCALE*2.54
2980 NEXT I
2990 TLON=INT((LOMAX-LOMIN)/TINLONG+.0001)+1
3000 TLOFST=LOMIN+(LOMIN/TINLONG-INT(LOMIN/TINLONG+.0001))*TINLONG
3010 DIM TLONG(TLON)
3020 FOR I=1 TO TLON
3030     TLONG(I)=(TLOFST+(I-1)*TINLONG-LOMIN)*SCALE*2.54
3040 NEXT I
3050 '
3060 '     Tables of the X & Y positions and values on the map for the labels
3070 '     ------------------------------------------------------------------
3080 '
3090   IF LABEL$()"Y" AND LABEL$()"y" THEN 3240
3100 LLAN=INT((LAMAX-LAMIN)/LINLAT+.0001)+1
3110 LLAFST=LAMIN+(LAMIN/LINLAT-INT(LAMIN/LINLAT+.0001))*LINLAT
3120 DIM LLAT(LLAN,2)
3130 FOR I=1 TO LLAN
3140     LLAT(I,1)=(-FNA(LAMIN)+FNA(LLAFST+(I-1)*LINLAT))/FPIE*SCALE*2.54
3150     LLAT(I,2)=LLAFST+(I-1)*LINLAT
3160 NEXT I
3170 LLON=INT((LOMAX-LOMIN)/LINLONG+.0001)+1
3180 LLOFST=LOMIN+(LOMIN/LINLONG-INT(LOMIN/LINLONG+.0001))*LINLONG
3190 DIM LLONG(LLON,2)
3200 FOR I=1 TO LLON
3210     LLONG(I,1)=(LLOFST+(I-1)*LINLONG-LOMIN)*SCALE*2.54
3220     LLONG(I,2)=LLOFST+(I-1)*LINLONG
3230 NEXT I
3240 '
3250 ' ------------------------------------------------------------------------
3260 '
3270 '          DRAWING PART
3280 '          ============
3290 '
3300 PRINT : PRINT :
3310 INPUT "PLOTTER READY (Y/N) : ",AN$
3320 AN$=LEFT$(AN$,1)
3330   IF AN$()"Y" AND AN$()"y" AND AN$()"N" AND AN$()"n" THEN 3310
3340   IF AN$="N" OR AN$="n" THEN 3310
3350 '
3360 '     Initialization of the plotter
3370 '     -----------------------------
3380 '
3390 OPEN "COM1:9600,S,7,1,RS,CS65535,DS,CD" AS 2
          'opening commucation line for the plotter (serial interface)
3400 PRINT #2,"PS0";                                  'A3 paper
3410   IF ROT=0 THEN PRINT #2,"IN"; : GOTO 3430       'initialisation
3420 PRINT #2,"IN RO90 IP IW";                        '  + rotation
3430 CEND$=CHR$(3)                    'character marking end of label
3440 PRINT #2,"DT" CEND$;
3450 '
3460 '     Title
3470 '     -----
3480 '
3490 '     setting character type and orientation, and color
3500 '     ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
3510 SIZEH=.25 : SIZEV=.3 : GOSUB 6440            'character width and height
3520 PENN$="2" : GOSUB 6510                       'choosing the pen
3530 '
3540 '     printing the title twice
3550 '     ~~~~~~~~~~~~~~~~~~~~~~~~
3560 CHAINN$=TIT$ : LTIT=LEN(TIT$)
3570 PENM=0
3580   IF ROT=0 THEN Y=27 ELSE Y=39
3590 X=2.5 : GOSUB 6320                           'moving pen in X,Y
3600 GOSUB 6390                                   'printing the title
3610 X=X+.02 : GOSUB 6320                         'moving pen in X,Y
3620 GOSUB 6390                                   'printing the title
3630 '
3640 '     printing the subtitle
3650 '     ~~~~~~~~~~~~~~~~~~~~~
3660 SIZEH=.25 : SIZEV=.2 : GOSUB 6440            'character width and height
3670 X=2.5 : Y=Y-1.5 : GOSUB 6320                 'moving pen in X,Y
3680 PENN$="3" : GOSUB 6510                       'choosing the pen
3690 CHAINN$=SUBTIT$
3700 GOSUB 6390                                   'printing the subtitle
3710 '
3720 '     changing the color
3730 '     ~~~~~~~~~~~~~~~~~~
3740 PENN$="1" : GOSUB 6510                       'choosing the pen
3750 '
3760 '
3770 '     Changing the origin
3780 '     -------------------
3790 '
3800 XXOR=2.5 : YOR=27-1.5-1.5-X0(1,1)
3810 X=XXOR : Y=YOR : GOSUB 6320                  'moving pen to X,Y
3820 '
3830 '     Drawing the frame
3840 '     -----------------
3850 '
3860 Y=YOR+X0(1,1) : PENM=1 : GOSUB 6320          '1st side
3870 X=XXOR+X0(2,2) : GOSUB 6320                  '2nd side
3880 Y=YOR : GOSUB 6320                           '3rd side
```

```
3890 X=XXOR : GOSUB 6310                              '4th side
3900 PENM=0
3910 '
3920 '     Drawing the grid lines
3930 '     ----------------------
3940 '
3950   IF GRID$<>"Y" AND GRID$<>"y" THEN 4100
3960 X=XXOR : Y=YOR : GOSUB 6310                      'moving pen to X,Y
3970 FOR I=1 TO GLAN
3980       IF GLAT(I)<=.00001 OR GLAT(I)>=X0(1,1)-.00001 THEN 4020
3990     X=XXOR : Y=YOR+GLAT(I) : GOSUB 6310          'drawing grid lines
4000     X=XXOR+X0(2,2) : PENM=1 : GOSUB 6310         'along parallels
4010     PENM=0
4020 NEXT I
4030 FOR I=1 TO GLON
4040       IF GLONG(I)<=.00001 OR GLONG(I)>=X0(2,2)-.00001 THEN 4080
4050     Y=YOR : X=XXOR+GLONG(I) : GOSUB 6310         'drawing grid lines
4060     Y=YOR+X0(1,1) : PENM=1 : GOSUB 6310          'along meridians
4070     PENM=0
4080 NEXT I
4090 END
4100 '     Drawing the tick marks
4110 '     ----------------------
4120 '
4130   IF TICK$<>"Y" AND TICK$<>"y" THEN 4400
4140 X=XXOR : Y=YOR : GOSUB 6310                      'moving pen to X,Y
4150 FOR I=1 TO TLAN
4160       IF TLAT(I)<0 OR TLAT(I)>X0(1,1) THEN 4200
4170     Y=YOR+TLAT(I) : X=XXOR : GOSUB 6310          'drawing tick marks
4180     X=X-.2 : PENM=1 : GOSUB 6310                 'along left side
4190     PENM=0
4200 NEXT I
4210 FOR I=1 TO TLON
4220       IF TLONG(I)<0 OR TLONG(I)>X0(2,2) THEN 4260
4230     Y=YOR+X0(1,1) : X=XXOR+TLONG(I) : GOSUB 6310    'drawing tick marks
4240     Y=Y+.2 : PENM=1 : GOSUB 6310                    'along top side
4250     PENM=0
4260 NEXT I
4270 FOR I=1 TO TLON
4280       IF TLONG(I)<0 OR TLONG(I)>X0(2,2) THEN 4320
4290     Y=YOR : X=XXOR+TLONG(I) : GOSUB 6310         'drawing tick marks
4300     Y=Y-.2 : PENM=1 : GOSUB 6310                 'along bottom side
4310     PENM=0
4320 NEXT I
4330 FOR I=1 TO TLAN
4340       IF TLAT(I)<0 OR TLAT(I)>X0(1,1) THEN 4380
4350     Y=YOR+TLAT(I) : X=XXOR+X0(2,2) : GOSUB 6310     'drawing tick marks
4360     X=X+.2 : PENM=1 : GOSUB 6310                    'along right side
4370     PENM=0
4380 NEXT I
4390 '
4400 '     Printing the labels
4410 '     -------------------
4420 '
4430   IF LABEL$<>"Y" AND LABEL$<>"y" THEN 4720
4440 FOR I=1 TO LLAN
4450       IF LLAT(I,1)<0 OR LLAT(I,1)>X0(1,1) THEN 4500
4460     NUMB=LLAT(I,2) : NPUIS=LINLATDEC : GOSUB 4740  'preparing label
4470     LENCH=LEN(CHAINN$)
4480     Y=YOR+LLAT(I,1) : X=XXOR-.5-LENCH*SIZEH         'printing label
4490     GOSUB 6310 : GOSUB 6380                         'on left side
4500 NEXT I
4510 FOR I=1 TO LLON
4520       IF LLONG(I,1)<0 OR LLONG(I,1)>X0(2,2) THEN 4570
4530     NUMB=LLONG(I,2) : NPUIS=LINLONGDEC : GOSUB 4740     'preparing label
4540     LENCH=LEN(CHAINN$)
4550     Y=YOR+X0(1,1)+.7 : X=XXOR+LLONG(I,1)-LENCH*SIZEH/2  'printing label
4560     GOSUB 6310 : GOSUB 6380                             'on top side
4570 NEXT I
4580 FOR I=1 TO LLON
4590       IF LLONG(I,1)<0 OR LLONG(I,1)>X0(2,2) THEN 4640
4600     NUMB=LLONG(I,2) : NPUIS=LINLONGDEC : GOSUB 4740     'preparing label
4610     LENCH=LEN(CHAINN$)
4620     Y=YOR-.7 : X=XXOR+LLONG(I,1)-LENCH*SIZEH/2          'printing label
4630     GOSUB 6310 : GOSUB 6380                             'on bottom side
4640 NEXT I
4650 FOR I=1 TO LLAN
4660       IF LLAT(I,1)<0 OR LLAT(I,1)>X0(1,1) THEN 4710
4670     NUMB=LLAT(I,2) : NPUIS=LINLATDEC : GOSUB 4740  'preparing label
4680     LENCH=LEN(CHAINN$)
4690     Y=YOR+LLAT(I,1) : X=XXOR+.5+X0(2,2)             'printing label
4700     GOSUB 6310 : GOSUB 6380                         'on right side
4710 NEXT I
4720 GOTO 4930
4730 '
4740 '     Subroutine to prepare labels for latitudes and longitudes
4750 '     ---------------------------------------------------------
4760 '
4770 NPUI=-NPUIS-3
4780 NUMB=NUMB+10^NPUI
4790   IF NPUIS<0 THEN CHAINN$=STR$(INT(NUMB)) : IF NUMB=0 THEN CHAINN$=RIGH
T$(CHAINN$,LEN(CHAINN$)-1) : GOTO 4910 ELSE GOTO 4910
4800 CHN$=STR$(NUMB) : IF NUMB=0 THEN CHN$=RIGHT$(CHN$,LEN(CHN$)-1)
4810 POINTPOS=INSTR(CHN$,".")
4820 CHAININ$=STR$(INT(NUMB))+"."                   'integer part
4830   IF POINTPOS=0 THEN CHAININ$=CHN$+"." : CHAINDEC$="" : GOTO 4870
4840 CHAINDEC$=MID$(CHN$,POINTPOS+1)                'decimal part
4850   IF LEN(CHAINDEC$)<NPUIS THEN 4870
4860 CHAINDEC$=LEFT$(CHAINDEC$,NPUIS) : GOTO 4900
4870 FOR J2=1 TO NPUIS-LEN(CHAINDEC$)
4880     CHAINDEC$=CHAINDEC$+"0"
4890 NEXT J2
4900 CHAINN$=CHAININ$+CHAINDEC$                     'label to be printed
4910 RETURN
4920 '
4930 '     Computing the X & Y positions of each data point
4940 '     before plotting it
4950 '     -------------------------------------------------
4960 '
4970 SIZEH=.2 : SIZEV=.24 : GOSUB 6430              'choosing size of symbols
4980 '
4990 '     preparing the parameters for each symbol and vector
5000 '     and then printing the symbol and drawing the vector
5010 '     ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
5020 DIM CHAR(10)
5030 FOR I=1 TO NSET
```

```
5040     CHAR(3)=LINPAR(I,7)
5050     CHAR(4)=LINPAR(I,8)
5060     CHAR(5)=LINPAR(I,9)
5070     CHAR(8)=LINPAR(I,10)
5080     FOR J=LINPAR(I,5) TO LINPAR(I,5)+LINPAR(I,6)-1
5090         CHAR(1)=(FNA(DATASET(J,LINPAR(I,1)))-FNA(LAMIN))/FPI#*SCALE*2.54
5100           IF TYP=0 OR DATASET(J,LINPAR(I,2))=0 THEN 5120
5110         DATASET(J,LINPAR(I,2))=DATASET(J,LINPAR(I,2))+360
5120         CHAR(2)=(DATASET(J,LINPAR(I,2))-LOMIN)*SCALE*2.54
5130         CHAR(6)=CHAR(1)+COS(DATASET(J,LINPAR(I,4))*FPI#)*DATASET(J,LINPAR
(I,3))*VLEN
5140         CHAR(7)=CHAR(2)+SIN(DATASET(J,LINPAR(I,4))*FPI#)*DATASET(J,LINPAR
(I,3))*VLEN
5150         CHAR(9)=DATASET(J,LINPAR(I,3))
5160         CHAR(10)=DATASET(J,LINPAR(I,4))
5170           IF CHAR(1)<0 OR CHAR(1)>X0(1,1) THEN 5280
5180           IF CHAR(2)<0 OR CHAR(2)>X0(2,2) THEN 5280
5190         X=XXOR+CHAR(2) : Y=YOR+CHAR(1) : GOSUB 6320
5200           IF CHAR(5)=0 THEN 5210
5210         PENN$=RIGHT$(STR$(CHAR(5)),1) : GOSUB 6510
5220           IF CHAR(3)<>10 THEN 5240
5230           IF J=LINPAR(I,5) THEN 5260 ELSE GOTO 5270
5240         ON CHAR(3) GOSUB 5350,5370,5390,5410,5430,5450,5470,5490,5510
5250         X=X-.1 : GOSUB 6320 : GOSUB 6390
5260         PENN$=RIGHT$(STR$(CHAR(8)),1) : GOSUB 6510
5270         ON CHAR(4) GOSUB 5570,5660,5660,5660,5660,5660,5660,5660
5280     NEXT J
5290 NEXT I
5300 GOTO 6090
5310 '
5320 '     Point symbols
5330 '     -------------
5340 '
5350 CHAINN$="+" : Y=Y-.12
5360 RETURN
5370 CHAINN$="*" : Y=Y-.12
5380 RETURN
5390 CHAINN$="#" : Y=Y-.144
5400 RETURN
5410 CHAINN$="$" : Y=Y-.144
5420 RETURN
5430 CHAINN$="." : Y=Y-.014
5440 RETURN
5450 CHAINN$="o" : Y=Y-.12
5460 RETURN
5470 CHAINN$="s" : Y=Y-.12
5480 RETURN
5490 CHAINN$="x" : Y=Y-.12
5500 RETURN
5510 CHAINN$="z" : Y=Y-.12
5520 RETURN
5530 '
5540 '     Vector drawing subroutine
5550 '     -------------------------
5560 '
5570 '     full line
5580 '     ~~~~~~~~~
5590 Y=YOR+CHAR(6) : X=XXOR+CHAR(7) : GOSUB 6320   'drawing first half vector
5600 Y=YOR+CHAR(1)-(CHAR(6)-CHAR(1))
5610 X=XXOR+CHAR(2)-(CHAR(7)-CHAR(2))
5620 PENM=1 : GOSUB 6320                           'drawing second half vector
5630 PENM=0
5640 RETURN
5650 '
5660 '     broken lines
5670 '     ~~~~~~~~~~~~
5680 ON CHAR(4) GOSUB 5980,6000,6020,6040,6060     'choosing the dash length
5690 Y=YOR+CHAR(6) : X=XXOR+CHAR(7) : GOSUB 6320   'going to end of vector
5700 NNDASH=CHAR(9)*VLEN/DASHLEN
5710 PENM=1
5720 NDASH=INT(NNDASH+.00001)                      'number of dashes by 1/2 vector
5730 FOR J2=1 TO NDASH
5740     X=X+(CHAR(2)-CHAR(7))/NNDASH
5750     Y=Y+(CHAR(1)-CHAR(6))/NNDASH
5760     GOSUB 6320                    'drawing one dash or moving to next dash
5770       IF PENM=0 THEN PENM=1 ELSE PENM=0
5780 NEXT J2
5790 Y=YOR+CHAR(1) : X=XXOR+CHAR(2) : GOSUB 6320   'going to middle point
5800 PENM=0
5810 X=XXOR+CHAR(2)-(CHAR(7)-CHAR(2))
5820 Y=YOR+CHAR(1)-(CHAR(6)-CHAR(1))
5830 GOSUB 6320                                    'going to second end of vector
5840 PENM=1
5850 FOR J2=1 TO NDASH
5860     Y=Y-(CHAR(1)-CHAR(6))/NNDASH
5870     X=X-(CHAR(2)-CHAR(7))/NNDASH
5880     GOSUB 6320                    'drawing one dash or moving to next dash
5890       IF PENM=0 THEN PENM=1 ELSE PENM=0
5900 NEXT J2
5910 Y=YOR+CHAR(1) : X=XXOR+CHAR(2) : GOSUB 6320   'going to middle point
5920 PENM=0
5930 RETURN
5940 '
5950 '     Length of the dashes in cm
5960 '     --------------------------
5970 '
5980 DASHLEN=.05
5990 RETURN
6000 DASHLEN=.08
6010 RETURN
6020 DASHLEN=.12
6030 RETURN
6040 DASHLEN=.17
6050 RETURN
6060 DASHLEN=.23
6070 RETURN
6080 '
6090 '     Printing the comments
6100 '     ---------------------
6110 '
6120   IF NCOM=0 THEN 6220
6130 X=XXOR
6140 PENN$="1" : GOSUB 6510                        'choosing pen
6150 SIZEH=.3 : SIZEV=.3 : GOSUB 6440              'character size and height
6160 FOR I=1 TO NCOM
```

```
6170     Y=YOR-1.5-I*.5
6180     CHAINN$=COMM$(I)
6190     GOSUB 6320 : GOSUB 6390                    'printing the comment lines
6200 NEXT I
6210 '
6220 '     End
6230 '     ---
6240 '
6250 END
6260 '
6270 ' ------------------------------------------------------------------------
6280 '
6290 '          PLOTTER SUBROUTINES
6300 '          ===================
6310 '
6320 '     Moving the pen
6330 '     --------------
6340 '
6350 XX=(X-1)*1016/2.54 : YY=(Y-1)*1016/2.54
6360   IF PENN=0 THEN PRINT #2,"PU";XX;YY; ELSE PRINT #2,"PD";XX;YY;
6370 RETURN
6380 '
6390 '     Writing a string of characters
6400 '     ------------------------------
6410 '
6420 PRINT #2,"LB";CHAINN$;CEND$
6430 '
6440 '     Initialization of character type and orientation
6450 '     -----------------------------------------------
6460 '
6470 SIZEHH=SIZEH*2/3
6480 PRINT #2,"SI";SIZEHH;SIZEV;
6490 RETURN
6500 '
6510 '     Changing the pen
6520 '     ----------------
6530 '
6540 PRINT #2,"SP";PENN$;
6550 RETURN
```

C

★ MERCATOR PROJECTION MAP FOR VECTOR DRAWING

Program to draw vectors on a Mercator projection map (WGS72 ellipsoid).

The data file can be created and must be corrected using a text editor. The first line has the number of lines, followed by the number of columns. The subsequent lines have the LAT and LONG of the data point followed by the direction of the vector and its magnitude.

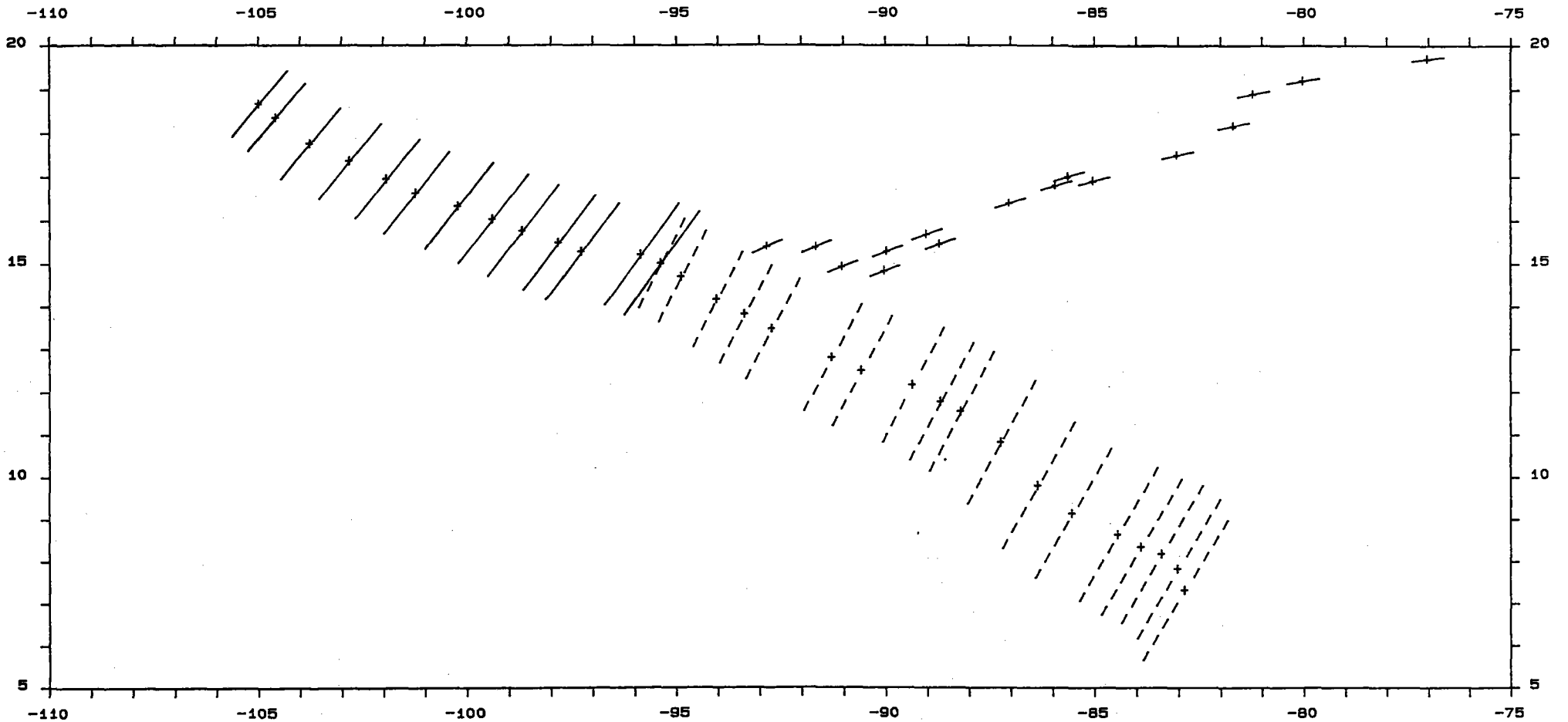
The plot will be drawn on a HP 7475A compatible plotter in the 21*29.7 cm format.

```
Do you want to create a new data set (1) or to use an old data file (2) : 2


Name of the data files on the default disk :


MIDAMEPL.DAT      MATPLATE.DAT


Name of your data file : MIDAMEPL.DAT

Title of the plot : MIDDLE AMERICA REGION
Subtitle : "COCOS-NORTH AMERICA, COCOS-CARIBBEAN & CARIBBEAN-NORTH AMERICA"

This table has  47  lines and  4  columns.

Number of X-Y data sets to plot : 3


You will need 3 pens on the plotter, all black or in color :
     Pen 1 is for the axes and the comments
     Pen 2 is for the title
     Pen 3 is for the subtitle

DATA SET  1
    Column number for latitude : 1
    Column number for longitude : 2
    Column number for vector length : 3
    Column number for vector direction : 4
    Do you want to use all the lines of the table (Y/N) : N
    Number of the first line to plot : 1
    Number of lines to be used : 13
Choose among these types of dots for the points location :
1 : + ;    2 : * ;    3 : # ;    4 : $ ;    5 : .
6 : o ;    7 : s ;    8 : x ;    9 : z
10 : no sign
? 1
      In which color (1 to 6) : 1
```

```
     Choose among these 6 types of line drawing :
           1 : full line
           2 : very short dashes (0.8 mm)
           3 : short dashes (1.2 mm)
           4 : medium dashes (1.6 mm)
           5 : long dashes (2.1 mm)
           6 : very long dashes (2.8 mm)
? 1
     In which color (1 to 6) : 1

DATA SET  2
     Column number for latitude : 1
     Column number for longitude : 2
     Column number for vector length : 3
     Column number for vector direction : 4
     Do you want to use all the lines of the table (Y/N) : N
     Number of the first line to plot : 31
     Number of lines to be used : 18
Choose among these types of dots for the points location :
1 : + ;    2 : * ;    3 : # ;    4 : $ ;    5 : .
6 : o ;    7 : s ;    8 : x ;    9 : z
10 : no sign
? 1
       In which color (1 to 6) : 1
     Choose among these 6 types of line drawing :
           1 : full line
           2 : very short dashes (0.8 mm)
           3 : short dashes (1.2 mm)
           4 : medium dashes (1.6 mm)
           5 : long dashes (2.1 mm)
           6 : very long dashes (2.8 mm)
? 5
     In which color (1 to 6) : 1

DATA SET  3
     Column number for latitude : 1
     Column number for longitude : 2
     Column number for vector length : 3
     Column number for vector direction : 4
     Do you want to use all the lines of the table (Y/N) : N
     Number of the first line to plot : 49
     Number of lines to be used : 16
Choose among these types of dots for the points location :
1 : + ;    2 : * ;    3 : # ;    4 : $ ;    5 : .
6 : o ;    7 : s ;    8 : x ;    9 : z
10 : no sign
? 1
       In which color (1 to 6) : 1
     Choose among these 6 types of line drawing :
           1 : full line
           2 : very short dashes (0.8 mm)
           3 : short dashes (1.2 mm)
           4 : medium dashes (1.6 mm)
           5 : long dashes (2.1 mm)
           6 : very long dashes (2.8 mm)
? 3
     In which color (1 to 6) : 1

Lat S =  7.32      Lat N =  19.7
Long W = -104.96     Long E = -77

     For the graph      Lat S : 5
                        Lat N : 20
                        Long W : -110
                        Long E : -75

Min vector length :  1.92    Max vector length :  9.917
What length in cm for a vector one unit long : .2


Do you want to print some lines of comments (up to 10 ) (Y/N) : Y
     1st line :
? FULL LINE : COCOS-NORTH AMERICA MEAN VALUES
     2nd line (if no more line press only RETURN)
? LONG DASHES : COCOS-CARIBBEAN MEAN VALUES
     3rd line (if no more line press only RETURN)
? SHORT DAHES : CARIBBEAN-NORTH AMERICA MEAN VALUES
     4 th line (if no more line press only RETURN)
? SCALE : 0.4 IN/DEG
     5 th line (if no more line press only RETURN)
? SPEED : 2 MM = 1CM/YR
     6 th line (if no more line press only RETURN)
?


What scale (in inches per degree) for the map : .4


Do you want grid lines (Y/N) : N

Do you want tick marks (Y/N) : Y
  Interval in decimal degrees along the latitudes : 1
  Interval in decimal degrees along the longitudes : 1

Do you want labels (Y/N) : Y
  Interval in decimal degrees along the latitudes : 5
    number of decimal characters : 0
  Interval in decimal degrees along the longitudes : 5
    number of decimal characters : 0


PLOTTER READY (Y/N) : Y
```

D MIDDLE AMERICA REGION

COCOS-NORTH AMERICA, COCOS-CARIBBEAN & CARIBBEAN-NORTH AMERICA



FULL LINE : COCOS-NORTH AMERICA MEAN VALUES
LONG DASHES : COCOS-CARIBBEAN MEAN VALUES
SHORT DASHES : CARIBBEAN-NORTH AMERICA MEAN VALUES
SCALE : 0.4 IN/DEG
SPEED : 2 MM = 1 CM/YR

SEISMES :
CARTE DES MAGNITUDES ET COUPES

10.1 - ENTREE DES DONNEES DE SEISMICITE

A - Listing du programme 'SEISEDIT'

B - Copie d'écran d'exemples d'exécution du programme 'SEISEDIT' :
B1 - création du fichier de données
B2 - correction du fichier de données

10.2 - TRANSFORMATION DU FICHIER DE DONNEES A ACCES SEQUENTIEL EN UN FICHIER DE DONNEES A ACCES ALEATOIRE

A - Listing du programme 'ERSQEDIT'

B - Copie d'écran d'un exemple d'exécution du programme 'ERSQEDIT'

10.3 - CALCUL DE LA MAGNITUDE AUX NOEUDS D'UNE GRILLE GEOGRAPHIQUE

A - Listing du programme 'SEISMES'

B - Copie d'écran d'un exemple d'exécution du programme 'SEISMES'

C - Exemple de sortie sur imprimante

10.4 - TRI DES SEISMES DANS UN POLYGONE POUR LES COUPES DE SEISMES

A - Listing du programme 'SEISSORT'

B - Copie d'écran d'un exemple d'exécution du programme 'SEISSORT'*

C - Exemple de sortie sur imprimante*

10.5 - DESSIN DES COUPES DE SEISMES

A - Exemple de fichier de données

B - Listing du programme `CROSSECTION'

C - Copie d'écran d'un exemple d'exécution du programme `CROSSECTION'

D - Exemple de sortie sur traceur réduite de 63 %

* les lettres a et b de la partie B correspondent à des sorties sur imprimante (partie C) portant le même nom.

10.1

A

```
10 '  ================================================================
20 '
30 '                              PROGRAM 'SEISEDIT'
40 '
50 '                                                 Jacqueline ROUMP, Sept. 1984
60 '  ================================================================
70 '
80 ON ERROR GOTO 1720
90 CLS
100 '
110 '    PRESENTATION
120 '    ============
130 PRINT "PROGRAM TO CREATE A SEQUENTIAL DATA FILE FOR EARTHQUAKE DATA AND"
140 PRINT "TO MODIFY IT"
150 PRINT
160 PRINT "The program ERSQEDIT will translate the sequential data file in a"
170 PRINT "random access data file that can be used by the programs SEISSORT"
180 PRINT "and SEISMES"
190 PRINT
200 PRINT "The program SEISSORT will be used to sort earthquakes in a chosen"
210 PRINT "area that may have up to 10 sides, depth and magnitude ranges"
220 PRINT "being other criteria of sort."
230 PRINT "The program SEISMES is used to compute the mean magnitude in an"
240 PRINT "area on a geographical-like grid."
250 PRINT
260 PRINT "PART 1 : to create the earthquake data file and to modify the file"
270 PRINT "(to add data at the end of the old file see Part 2)"
280 PRINT : PRINT
290 PRINT "Press : J to jump"
300 PRINT "        N to continue to next line"
310 PRINT "        I to insert a line before the last line on the screen"
320 PRINT "        S to suppress the whole line"
330 PRINT "        W to change the whole line"
340 PRINT "        1,2,3 or 4 to change the 1st, 2nd, 3rd or 4th number"
350 '
360   INPUT "Disk ready (Y/N) ";R$ : R$=LEFT$(R$,1)
370       IF R$<>"Y" AND R$<>"y" THEN 360
380 PRINT "Names of the data files of the disk :"
390 FILES "*.DAT"
400   INPUT "What is your earthquakes sequential data file : ";X$
410 PRINT
420 N=1
430 '-------------------------------------------------------------------------
440 '
450 '    READING AND MODIFYING THE OLD DATA AND COPYING THEM INTO THE NEW FILE
460 '    ====================================================================
470 '
480 '    Opening the data files
490 '    ----------------------
500 Y$="ERSQAK.DAT"
510 FILES "ERSQAK.DAT"
520 OPEN "I",#1,X$
530 PRINT "Give a name for the intermediate data file (one that does not"
540   INPUT "exist : ";Y$
550 OPEN "O",#2,Y$
560 '
570 '    To modify the old data file
580 '    ---------------------------
590 N=1
600   INPUT "Do you want any modification to the old file (Y/N) ";R$
610       IF LEFT$(R$,1)="Y" OR LEFT$(R$,1)="y" THEN 620 ELSE 1310
620 PRINT
630 PRINT "Approximate number from the beginning of the file to the data"
640   INPUT "to modify ";Q
650       IF EOF(1) THEN 1370                    'EOF = end of file
660 '
670 '    Data go from the disk to the memory
680 '    -----------------------------------
690 INPUT #1,T,G,M,P : IF N<Q THEN 1120
700 '
710 '    And from memory to the screen when wanted
720 '    -----------------------------------------
730 PRINT N
740 PRINT T;"  ";G;"  ";M;"  ";P
750 '
760 '    Waiting for the character input
770 '    -------------------------------
780 A$=INKEY$ : IF A$="" THEN 780
790 '
800 '    Tests on the characters
810 '    -----------------------
820       IF A$="I" OR A$="i" AND (C$="J" OR C$="j")
          THEN PRINT "Press N if good, or another key to correct the line only"
             : GOTO 780
830       IF A$="J" OR A$="j" THEN 600
840       IF A$="N" OR A$="n" THEN 1120
850       IF A$="I" OR A$="i" THEN 1020
860       IF A$="S" OR A$="s" THEN 650
870       IF A$="W" OR A$="w" THEN 1030
880 '
890 '    To change the values one by one
900 '    -------------------------------
910       IF A$="1" THEN 920 ELSE 930
920   INPUT "Lat ";T : GOTO 740
930       IF A$="2" THEN 940 ELSE 950
940   INPUT "Long ";G : GOTO 740
950       IF A$="3" THEN 960 ELSE 970
960   INPUT "Magn ";M : GOTO 740
970       IF A$="4" THEN 980 ELSE 1080
980   INPUT "Depth ";P : GOTO 740
990 '
1000 '    To change a whole line and to insert lines
1010 '    ------------------------------------------
1020 TB=T : GB=G : MB=M : PB=P
1030   INPUT "Lat ";T
1040   INPUT "Long ";G
1050   INPUT "Magn ";M
1060   INPUT "Depth ";P : GOTO 740
1070 '
1080 '    Wrong Key
1090 '    ---------
1100 BEEP : GOTO 780
1110 '
1120 '     Writing the data in the new file
1130 '     --------------------------------
```

```
1140 PRINT #2,T;G;M;P
1150 C$=A$
1160 N=N+1
1170 '
1180 '    Inserting other lines ?
1190 '    -----------------------
1200       IF MB<>0 THEN 1210 ELSE 650
1210   INPUT "Do you want to insert another line (Y/N) ";B$
1220       IF B$="Y" OR B$="y" THEN 1030
1230 T=TB : G=GB : M=MB : P=PB : TB=0 : GB=0 :MB=0 : PB=0
1240 GOTO 710
1250 '
1260 '    Closing the old data file
1270 '    -------------------------
1280 CLOSE #1
1290 PRINT #2,T;G;M;P : GOTO 1380
1300 '
1310 '    Copying the whole sequential data file and erasing it
1320 '    -----------------------------------------------------
1330       IF EOF(1) THEN 1370
1340 INPUT #1,T,G,M,P
1350 PRINT #2,T;G;M;P
1360 GOTO 1330
1370 CLOSE #1
1380 KILL X$
1390 PRINT
1400 '-------------------------------------------------------------------------
1410 '
1420 '    PART 2 : NEW DATA
1430 '    =================
1440 '
1450 '    ------------
1460 PRINT "PART 2 : to add new earthquake data"
1470 PRINT "To exit, enter 1000 at the querry `Lat '"
1480 '
1490 '    Data input from the keyboard, and verifications
1500 '    -----------------------------------------------
1510 PRINT "N = ";N
1520   INPUT "Lat "; T : IF T=1000 THEN 1630
1530   INPUT "Long "; G
1540   INPUT "Magn "; M
1550   INPUT "Depth "; P
1560 '
1570 '    Writing the new data into the new data file
1580 '    -------------------------------------------
1590 PRINT #2,T;G;M;P
1600 N=N+1
1610 GOTO 1510
1620 '
1630 '    Closing the new data file
1640 '    -------------------------
1650 CLOSE #2
1660 '
1670 '    Changing the filename back to old one
1680 '    -------------------------------------
1690 NAME Y$ AS X$
1700 END
1710 '
```

```
1720 '    Only error treatment : the new file does not exist yet
1730 '    ------------------------------------------------------
1740 IF ERR=53 AND ERL=510 THEN RESUME 520
1750 IF ERR=53 AND ERL=520 THEN RESUME 1760 ELSE 1770
1760 FILES "ERSQAK.DAT"
1770 IF ERR=53 AND ERL=1760 THEN 1780 ELSE 1800
1780 OPEN "O",#2,"ERSQAK.DAT" : RESUME 1460
1790 '
1800 '    Other errors
1810 '    ------------
1820 CLOSE : PRINT "ERROR NUMBER = ";ERR;"    LINE NUMBER = ";ERL
1830 END
```

B

★

```
PROGRAM TO CREATE A SEQUENTIAL DATA FILE FOR EARTHQUAKE DATA AND
TO MODIFY IT

The program ERSQEDIT will translate the sequential data file in a
random access data file that can be used by the programs SEISSORT
and SEISMES

The program SEISSORT will be used to sort earthquakes in a chosen
area that may have up to 10 sides, depth and magnitude ranges
being other criteria of sort.
The program SEISMES is used to compute the mean magnitude in an
area on a geographical-like grid.

PART 1 : to create the earthquake data file and to modify the file
(to add data at the end of the old file see Part 2)


Press : J to jump
        N to continue to next line
        I to insert a line before the last line on the screen
        S to suppress the whole line
        W to change the whole line
        1,2,3 or 4 to change the 1st, 2nd, 3rd or 4th number
Disk ready (Y/N) ? Y
```

B 1

```
Names of the data files of the disk :
ERSQUAKE.DAT MATPNTSS.DAT MATPNTFZ.DAT MATPNTTA.DAT MATPNTRM.DAT RIVPNTSS.DAT
RIVPNTPL.DAT DISCMAT3.DAT DISCMAT2.DAT MATPNTS1.DAT DISCMAT4.DAT DISCMAT1.DAT
MATPNTPL.DAT RIVPOINT.DAT
What is your earthquakes sequential data file : ? EARTHQAK.DAT

PART 2 : to add new earthquake data
To exit, enter 1000 at the querry 'Lat '
N =  1
Lat ? 7.5
Long ? -82
Magn ? 6.63
Depth ? 8
N =  2
Lat ? 7.5
Long ? -81
Magn ? 6.5
Depth ? 8
N =  3
Lat ? 17
Long ? -94.5
Magn ? 6.9
Depth ? 150
N =  4
Lat ? 6.8
Long ? -81
Magn ? 6.75
Depth ? 55
N =  5
Lat ? -3.1
Long ? -80.9
Magn ? 6.5
Depth ? 27
N =  6
Lat ? -3.1
Long ? -80.9
Magn ? 6.5
Depth ? 27
N =  7
Lat ? 12.7
Long ? -87.3
Magn ? 6.5
Depth ? 138
N =  8
Lat ? 8.1
Long ? -82.9
Magn ? 6.75
Depth ? 30
N =  9
Lat ? -10
Long ? -79
Magn ? 6.7
Depth ? 39
N =  10
Lat ? 17
Long ? -79
Magn ? 6.7
Depth ? 39
N =  11
Lat ? 1000
```

B2

```
Names of the data files of the disk :
ERSQUAKE.DAT MATPNTSS.DAT MATPNTFZ.DAT MATPNTTA.DAT MATPNTRM.DAT RIVPNTSS.DAT
RIVPNTPL.DAT DISCMAT3.DAT DISCMAT2.DAT MATPNTS1.DAT DISCMAT4.DAT DISCMAT1.DAT
MATPNTPL.DAT RIVPOINT.DAT EARTHQAK.DAT
What is your earthquakes sequential data file : ? EARTHQAK.DAT

Give a name for the intermediate data file (one that does not
exist : ? TEMP.DAT
Do you want any modification to the old file (Y/N) ? Y

Approximate number from the beginning of the file to the data
to modify ? 2
 2
 7.5   -81    6.5    0
N
 3
 17   -94.5    6.9    150
N
 4
 6.8   -81    6.75    55
I Lat ? -6.8
-6.8   -81    6.75    55
N
 5
-3.1   -80.9    6.5    27
N
 6
-3.1   -80.9    6.5    27
S
 8
 12.7   -87.3    6.5    138
N
 7
 8.100001   -82.9    6.75    30
I Lat ? 13.6
  Long ? -92.5
  Magn ? 6.5
  Depth ? 37
   13.6   -92.5    6.5    37
  Do you want to insert another line (Y/N) ? n
   8
N  8.100001   -82.9    6.75    30
   9
N -10   -79    6.7    39
   10
N  17   -79    6.7    39
W Lat ? 17
  Lons ? -99.6
  Magn ? 7
  Depth ? 40
N  17   -99.6    7    40
```

```
PART 2 : to add new earthquake data
To exit, enter 1000 at the querry 'Lat '
N =  11
Lat ? 17.2
Long ? -99.5
Magn ? 7.13
Depth ? 33
N =  12
Lat ? 7.4
Long ? -82.7
Magn ? 6.75
Depth ? 33
N =  13
Lat ? 5.2
Long ? 76.4
Magn ? 6.75
Depth ? 69
N =  14
Lat ? 5.2
Long ? -76.4
Magn ? 6.75
Depth ? 69
N =  15
Lat ? 2.1
Long ? -83.5
Magn ? 6.63
Depth ? 33
N =  16
Lat ? 5.4
Long ? -82.5
Magn ? 6.5
Depth ? 33
N =  17
Lat ? 16.3
Long ? -82.5
Magn ? 6.5
Depth ? 33
N =  18
Lat ? -9.176
Long ? -78.823
Magn ? 7.8
Depth ? 43
N =  19
Lat ? 1000
```

10.2

A

```
10 '  ============================================================
20 '
30 '                          PROGRAM `ERSQEDIT'
40 '
50 '                                          Jacqueline ROUMP. Sept. 1984
60 '  ============================================================
70 '
80 ON ERROR GOTO 2090
90 CLS
100 '
110 '    PRESENTATION
120 '    ============
130 PRINT "PROGRAM TO MODIFY EARTHQUAKE SEQUENTIAL DATA FILE"
140 PRINT "AND TO TRANSLATE IT INTO A FINAL RANDOM ACCESS DATA FILE"
150 PRINT "USED BY THE PROGRAMS `SEISSORT' AND `SEISMES'"
160 PRINT
170 PRINT "The program SEISSORT will be used to sort earthquakes in a chosen"
180 PRINT "area that may have up to 10 sides, depth and magnitude ranges"
190 PRINT "being other criteria of sort."
200 PRINT "The program SEISMES is used to compute the mean magnitude in an"
210 PRINT "area on a geographical-like grid."
220 PRINT
230 PRINT "   PART 1 : to modify the sequential earthquake data file"
240 PRINT "and to put the data in a random access data file."
250 PRINT "To add new data : see Part 2."
260 PRINT
270 PRINT "Press : J to jump"
280 PRINT "        N to continue to next line"
290 PRINT "        I to insert a line before the last line on the screen"
300 PRINT "        S to suppress the whole line"
310 PRINT "        W to change the whole line"
320 PRINT "        1,2,3 or 4 to change the 1st, 2nd, 3rd or 4th number"
330 PRINT "        (1 : lat, 2 : long, 3 : magnitude, 4 depth)"
340 PRINT
350 '
360   INPUT "Disk ready (Y/N) ";R$ : R$=LEFT$(R$,1)
370       IF R$<>"Y" AND R$<>"y" THEN 360
380 PRINT "Data files of the disk :"
390 FILES "*.DAT"
400   INPUT "Name of your earthquake sequential data file : ";X$
410   INPUT "Name of your earthquake random access data file : ";Y$
420 N=1
430 MB=0
440 '-------------------------------------------------------------------------
450 '
460 '    READING AND MODIFYING THE OLD DATA AND COPYING THEM INTO THE NEW FILE
470 '    ============================================================
480 '
490 '    Opening the data files
500 '    ----------------------
510 OPEN "I",#1,X$                          'sequential data file
520 OPEN "R",#2,Y$,14                       'random access data file
530 FIELD #2, 4 AS T$, 4 AS G$, 4 AS M$, 2 AS P$
540 '
550 '    Reading the number of records of the random access data file
560 '    -----------------------------------------------------------
570 GET #2,1
580 PRINT "Last record of the random access data file : "; CVS(T$)
590 PRINT
600 '
610 '    Looking at the random access data file
620 '    --------------------------------------
630 PRINT "Which record of the random access data file do you wan to see"
640   INPUT "(type 0 if none) ";S
650       IF S=0 THEN 720
660 GET #2,S+1
670 PRINT
680 PRINT CVS(T$),CVS(G$),CVS(M$),CVI(P$)
690 PRINT
700   INPUT "Do you want to see another record (Y/N) ";R$ : R$=LEFT$(R$,1)
710       IF R$="Y" OR R$="y" THEN 630
720 PRINT
730 PRINT "Number of the record of the random access data file to begin"
740   INPUT "to copy the sequential data file ";M2
750 '
760 '    To modify the old data file
770 '    ---------------------------
780   INPUT "Do you want any modification to the sequential data file (Y/N) ";
R$
790       IF LEFT$(R$,1)="Y" OR LEFT$(R$,1)="y" THEN 800 ELSE 1610
800 PRINT "Approximate number from the beginning of the sequential data file"
810   INPUT "to the data to modify ";Q
820 PRINT
830       IF EOF(1) THEN 1700                    'EOF = end of file
840 '
850 '    Data go from the disk to the memory
860 '    -----------------------------------
870 INPUT #1,T,G,M,P
880       IF N<Q THEN 1450
890 '         .
900 '    And from memory to the screen when asked
910 '    ----------------------------------------
920 PRINT N; "  will become  ";M2
930 PRINT T;"  ";G;"  ";M;"  ";P
940 PRINT
950 '
960 '    Waiting for the character input
970 '    -------------------------------
980 A$=INKEY$ : IF A$="" THEN 980
990 '
1000 '    Tests on the characters
1010 '    -----------------------
1020       IF A$="I" OR A$="i" AND (C$="J" OR C$="j
       THEN PRINT "PRESS N IF GOOD, OR ANOTHER KEY TO CORRECT THE LINE ONLY"
            : GOTO 890
1030       IF A$="J" OR A$="j" THEN 760
1040       IF A$="N" OR A$="n" THEN 1450
1050       IF A$="I" OR A$="i" THEN 1330
1060       IF A$="S" OR A$="s" THEN 1080 ELSE 1200
1070 '
1080 '    Erasing lines
1090 '    -------------
1100   INPUT "How many lines do you want to suppress ";U
1110       IF U=1 THEN 830
```

```
1120 '---------------
1130 FOR I=1 TO U
1140       IF EOF(1) THEN 1700
1150    INPUT #1,T,G,M,P
1160 NEXT I
1170 '---------------
1180 GOTO 830
1190 '
1200       IF A$="W" OR A$="w" THEN 1360
1210 '
1220 '     To change the values one by one
1230 '     -------------------------------
1240       IF A$="1" THEN 1250 ELSE 1260
1250   INPUT "Lat ";T : GOTO 930
1260       IF A$="2" THEN 1270 ELSE 1280
1270   INPUT "Long ";G : GOTO 930
1280       IF A$="3" THEN 1290 ELSE 1300
1290   INPUT "Magn ";M : GOTO 930
1300       IF A$="4" THEN 1310 ELSE 1410
1310   INPUT "Depth ";P : GOTO 930
1320 '
1330 '     To change a whole line and to insert lines
1340 '     ------------------------------------------
1350 TB=T : GB=G : MB=M : PB=P
1360   INPUT "Lat ";T
1370   INPUT "Long ";G
1380   INPUT "Magn ";M
1390   INPUT "Depth ";P : GOTO 930
1400 '
1410 '     Wrong key
1420 '     ---------
1430 BEEP : GOTO 960
1440 '
1450 '     Writing the data in the new file
1460 '     --------------------------------
1470 LSET T$=MKS$(T) : LSET G$=MKS$(G) : LSET M$=MKS$(M) : LSET P$=MKI$(P)
1480 PUT #2,M2+1
1490 C$=A$
1500 N=N+1 : M2=M2+1
1510 '
1520 '     Inserting other lines ?
1530 '     -----------------------
1540       IF MB<>0 THEN 1550 ELSE 830
1550   INPUT "Do you want to insert another line ? (Y/N)";R$ : R$=LEFT$(R$,1)
1560       IF R$="Y" OR R$="y" THEN 1360
1570 T=TB : G=GB : M=MB : P=PB
1580 TB=0 : GB=0 : MB=0 : PB=0
1590 GOTO 920
1600 '
1610 '     Writing the whole sequential data file
1620 '     --------------------------------------
1630       IF EOF(1) THEN 1700
1640 INPUT #1,T,G,M,P
1650 LSET T$=MKS$(T) : LSET G$=MKS$(G) : LSET M$=MKS$(M) : LSET P$=MKI$(P)
1660 PUT #2,M2+1
1670 M2=M2+1
1680 GOTO 1630
1690 '
1700 '     Closing the sequential data file
1710 '     --------------------------------
1720 CLOSE #1
1730 PRINT : PRINT
1740 '-------------------------------------------------------------------------------
1750 '
1760 '     PART 2 : NEW DATA
1770 '     =================
1780 '
1790 PRINT "   PART 2 : to add new earthquake data to the random access file"
1800 PRINT "To exit, enter 1000 at the querry `Lat '"
1810 INPUT "Random access file record number to write the next record ";M2
1820 '
1830 '     Data input from the keyboard, and verifications
1840 '     -----------------------------------------------
1850 PRINT
1860 PRINT "N = ";M2
1870   INPUT "Lat "; T : IF T=1000 THEN 1990
1880   INPUT "Long "; G
1890   INPUT "Magn "; M
1900   INPUT "Depth "; P
1910 '
1920 '     Writing the new data into the new data file
1930 '     -------------------------------------------
1940 LSET T$=MKS$(T) : LSET G$=MKS$(G) : LSET M$=MKS$(M) : LSET P$=MKI$(P)
1950 PUT #2,M2+1
1960 N=N+1 : M2=M2+1
1970 GOTO 1850
1980 '
1990 '     Writing the number of records of the random access data file
2000 '     in the first record and closing the file
2010 '     ----------------------------------------------------------
2020 LSET T$=MKS$(M2-1) : LSET G$=MKS$(M2-1)
2030 LSET M$=MKS$(M2-1) : LSET P$=MKI$(M2-1)
2040 PUT #2,1
2050 CLOSE #2
2060 END
2070 '
2080 '     Only error treatment : the random access file does not exist yet
2090 '     ----------------------------------------------------------------
2100       IF ERR=53 AND ERL=510 THEN OPEN "R",#2,Y$,14 ELSE 2140
2110 GET #2,1 : PRINT "Last record in the file = ";CVS(T$)
2120 RESUME 1790
2130 '
2140 '     Other errors
2150 '     ------------
2160 CLOSE : PRINT "ERROR NUMBER = ";ERR;"    LINE NUMBER = ";ERL
2170 END
```

```
B
★
 PROGRAM TO MODIFY EARTHQUAKE SEQUENTIAL DATA FILE
 AND TO TRANSLATE IT INTO A FINAL RANDOM ACCESS DATA FILE
 USED BY THE PROGRAMS 'SEISSORT' AND 'SEISMES'

 The program SEISSORT will be used to sort earthquakes in a chosen
 area that may have up to 10 sides, depth and magnitude ranges
 being other criteria of sort.
 The program SEISMES is used to compute the mean magnitude in an
 area on a geographical-like grid.

    PART 1 : to modify the sequential earthquake data file
 and to put the data in a random access data file.
 To add new data : see Part 2.

 Press : J to jump
         N to continue to next line
         I to insert a line before the last line on the screen
         S to suppress the whole line
         W to change the whole line
         1,2,3 or 4 to change the 1st, 2nd, 3rd or 4th number
         (1 : lat, 2 : long, 3 : magnitude, 4 depth)

 Disk ready (Y/N) ? Y
 Data files of the disk :
 ERSQUAKE.DAT MATPNTSS.DAT MATPNTFZ.DAT MATPNTTA.DAT MATPNTRM.DAT RIVPNTSS.DAT
 RIVPNTPL.DAT DISCMAT3.DAT DISCMAT2.DAT MATPNTS1.DAT DISCMAT4.DAT DISCMAT1.DAT
 MATPNTPL.DAT RIVPOINT.DAT EARTHQAK.DAT
 Name of your earthquake sequential data file : ? EARTHQAK.DAT
 Name of your earthquake random access data file : ? SEIS.DAT
 Last record of the random access data file :  0

 Which record of the random access data file do you want to see
 (type 0 if none) ? 0

 Number of the record of the random access data file to begin
 to copy the sequential data file ? 1
 Do you want any modification to the sequential data file (Y/N) ? Y
 Approximate number from the beginning of the sequential data file
 to the data to modify ? 10

  10   will become   10
  17   -99.6    7    40
N
  11   will become   11
  17.2   -99.5    7.13    33
N
  12   will become   12
  7.4   -82.7    6.75    33
N
  13   will become   13
  5.2    76.4    6.75    69
2
 Long ? -76.4
  5.2   -76.4    6.75    69
  14   will become   14
  5.2   -76.4    6.75    69
S
 How many lines do you want to suppress ? 1
  14   will become   14
  2.1   -83.5    6.63    33
N
  15   will become   15
  5.4   -82.5    6.5    33
I
 Lat ? 7.5
 Long ? -82.3
 Magn ? 7
 Depth ? 33
  7.5   -82.3    7    33
N
 Do you want to insert another line ? (Y/N)? n
  16   will become   16
  5.4   -82.5    6.5    33
N
  17   will become   17
  16.3   -82.5    6.5    33
W
 Lat ? 16.3
 Long ? -95.8
 Magn ? 7.25
 Depth ? 20
  16.3   -95.8    7.25    20
N
  18   will become   18
 -9.175999   -78.823    7.8    43
N

    PART 2 : to add new earthquake data to the random access file
 To exit, enter 1000 at the querry 'Lat '
 Random access file record number to write the next record ? 19

 N =  19
 Lat ? 18.267
 Long ? -96.598
 Magn ? 7.2
 Depth ? 84

 N =  20
 Lat ? 17.813
 Long ? -101.276
 Magn ? 7.6
 Depth ? 49

 N =  21
 Lat ? 1000
```

10.3

A

```
10 ' ============================================================
20 '
30 '                          PROGRAM `SEISMES'
40 '
50 '                                        Jacqueline ROUMP, Oct. 1984
60 ' ============================================================
70 '
80 CLS : CLEAR : OPTION BASE 1
90 '
100 '     PRESENTATION AND QUESTIONS
110 '     ==========================
120 '
130 PRINT "   Program to compute the average seismic magnitude in an area."
140 PRINT
150 PRINT "   The data are read in a random access data file, created by the"
160 PRINT "programs SEISEDIT and ERSQEDIT. The results are presented on a"
170 PRINT "geographical-like table : the magnitude is computed for every knot"
180 PRINT "of a 0.2 degree grid, if the value is zero the sign is a period."
190 PRINT
200 PRINT "   The assignation of a magnitude to a knot of the grid is"
210 PRINT "different if the earthquake has taken place very close to a knot,"
220 PRINT "on a line between two knots or elsewhere."
230 PRINT
240 PRINT "   The program uses the relations between magnitude and energy, and
    "
250 PRINT "between the earthquake regional magnitude and body-wave magnitude,"
260 PRINT "and follows the method by Markus BATH described in two articles :"
270 PRINT "`Seismic Energy Mapping applied to Sweden', 1982, Tectonophysics,"
280 PRINT "vol. 81, p. 85-98, and `Earthquake Data Analysis : an Example from"
290 PRINT "Sweden', 1983, Earth-Science Reviews, vol. 19, p. 181-303. Two"
300 PRINT "modifications have been introduced : there is no area correction"
310 PRINT "in this version used for locations close to the Equator, and the"
320 PRINT "earthquake magnitude is distributed on the grid in a gaussian way."
330 PRINT
340 PRINT "   A fan-folded paper is recommanded."
350 PRINT
360 '
370   INPUT "DISK READY (Y/N) ";R$ : R$=LEFT$(R$,1)
380       IF R$<>"Y" AND R$<>"y" THEN 370
390 PRINT
400 PRINT "Data files on the disk :"
410 FILES "*.DAT"                                  'data files of the disk
420 PRINT
430   INPUT "What is your data file ";X$
440 PRINT
450 '
460 '    Questions about the title and subtitle
470 '    --------------------------------------
480   INPUT "Title : ";W$
490 PRINT
500   INPUT"   subtitle : ";S$
510 PRINT
520 '
530 '    Questions about the geographical limits of the study
540 '    ---------------------------------------------------
550 PRINT "North latitude in degrees (use + for the northern hemisphere"
560   INPUT "and - for the southern hemisphere) : ";LATN
570   INPUT "South latitude : ";LATS
580       IF LATN>LATS THEN 600
590 SWAP LATN,LATS                                 'exchange LATN and LATS
600 PRINT
610 PRINT "West longitude in degrees (use + East of Greenwich and - West of"
620   INPUT "Greenwich) : ";LONGW
630   INPUT "East longitude : ";LONGE
640 WIDS=LONGE-LONGW
650       IF WIDS<0 THEN WIDS=360+WIDS
660 PRINT "The width of the grid is ";CINT(WIDS*100)/100;" degrees.";
670   INPUT "  Is it OK (Y/N) ";R$ : R$=LEFT$(R$,1)
680       IF R$<>"Y" AND R$<>"y" THEN 610
690       IF LONGE<LONGW THEN LL=1                 '180 deg of longitude flag
700       IF LL=1 THEN LONGE=360+LONGE
710 W=INT((450/(LATN-LATS+.6)-.8)*5)/5
720 NSETS=INT((WIDS-.8)/W)+1
                    'number of tables to create to cover the whole area
730 PRINT
740 PRINT "The program will have to read the data file on the disk ";NSETS
750 PRINT "TIMES and to print ";NSETS;" sets of results."
760 PRINT
770 '
780 '    Questions about the magnitude choice
790 '    ------------------------------------
800 PRINT "MAGNITUDE CHOICE : number of intervals (up to 10), type 0 if any"
810   INPUT "magnitude : ";NM
820       IF NM=0 THEN 900
830       IF NM<0 OR NM>10 THEN PRINT "Number from 0 to 10." : GOTO 800
840 '---------------
850 FOR I=1 TO NM
860    PRINT "Interval ";I;" : minimum included "; : INPUT MI(I)
870    INPUT "               maximum excluded "; MA(I)
880 NEXT I
890 '---------------
900 PRINT
910 '
920 '    Questions about the depth choice
930 '    --------------------------------
940 PRINT "DEPTH CHOICE : number of intervals (up to 10), type 0 if all"
950   INPUT "depths (depths of 0 and 33 km mean undefined depth) : ";DN
960       IF DN=0 THEN 1060
970       IF DN<0 OR DN>10 THEN PRINT "Number from 0 to 10." : GOTO 940
980 '---------------
990 FOR I=1 TO DN
1000    PRINT "Interval ";I;" : minimum included "; : INPUT DI(I)
1010    INPUT "               maximum excluded ";DA(I)
1020 NEXT I
1030 '---------------
1040 '-------------------------------------------------------------------------
1050 '
1060 '     READING AND COMPUTATION PART
1070 '     ===========================
1080 '
1090 GM1=LONGW                      'GM1 = W long of the first set of table
1100 IF WIDS>(W+.8) THEN GM2=GM1+W+.8 ELSE GM2=LONGE
                                   'GM2 = E long of the first set of table
```

```
1110 LM=(LATN-LATS)*5+3           'number of lines of the table
1120 NTIMES=1                     'table sets counter
1130 NP=1                         'page counter
1140 GM=CINT((GM2-GM1)*5+3)       'number of columns of this set of table
1150 DIM MAG(LM,GM)               'table for the values on the grid knots
1160 PRINT
1170 PRINT "The program will use the disk for the ";NTIMES;" TIME,";
1180   INPUT " are you ready (Y/N) ";R$ : R$=LEFT$(R$,1)
1190       IF R$<>"Y" AND R$<>"y" THEN 1170
1200 CLS : LOCATE 15,5 : PRINT "READING AND COMPUTING THE ENERGY"
1210 PRINT
1220 OPEN "R",#1,X$,14            'X$ is the name of the data file used here
1230 FIELD #1, 4 AS T$, 4 AS G$, 4 AS M$, 2 AS P$
                                  'T$=lat, G$=long, M$=magnitude, P$=depth
1240 '
1250 '    Reading the number of records in the data file
1260 '    ----------------------------------------------
1270 GET #1, 1
1280 N=CVS(T$)
1290 '
1300 '    Reading the value records
1310 '    -------------------------
1320 '---------------
1330 FOR I%=2 TO N+1
1340    GET #1,I%
1350       IF CVS(T$)<LATS OR CVS(T$)>LATN THEN 2100
1360    T1=CVS(T$)-LATS
1370    G=CVS(G$)
1380       IF LL=1 AND G<0 THEN G=360+G
1390       IF G<GM1 OR G>GM2 THEN 2100
1400    M=CVS(M$)
1410       IF NM=0 THEN 1480
1420 '       ---------------
1430         FOR I=1 TO NM
1440               IF M>=MI(I) AND M<MA(I) THEN 1480
1450         NEXT I
1460 '       ---------------
1470    GOTO 2100    'the earthquake is outside the selected magnitude classes
1480    P=CVI(P$)
1490       IF DN=0 THEN 1560
1500 '       ---------------
1510         FOR I=1 TO DN
1520               IF P>=DI(I) AND P<DA(I) THEN 1560
1530         NEXT I
1540 '       ---------------
1550    GOTO 2100        'the earthquake is outside the selected depth classes
1560    A=CINT(T1*5)+2                  'closest line number in the table
1570    B=CINT(G*5)-GM1*5+2             'closest column number in the table
1580    E=(10^(2.57*M))
     'energy released by the earthquake (2.57 from M. BATH (1982 & 1983))
1590       IF CINT(T1*10) MOD 2=0 THEN X=1 ELSE X=0
                        'if X=1 earthquake close to an horizontal grid line
1600       IF CINT(ABS(G)*10) MOD 2=0 THEN Y=1 ELSE Y=0
                          'if Y=1 earthquake close to a vertical grid line
1610       IF X=1 AND Y=1 THEN 1830
1620       IF X=1 THEN 1920
1630       IF Y=1 THEN 2030
1640 '
1650 '       Earthquakes not close to a knot nor a grid line
1660 '       -----------------------------------------------
1670       IF T1*10-INT(T1*10)<.5 THEN 1690  'to determine the knots latitudes
1680    C=A+1 : A1=A-1 : C1=C+1 : GOTO 1700
                'the earthquake is closer to the lower grid latitude line
1690    C=A-1 : A1=A+1 : C1=C-1
                'the earthquake is closer to the upper grid latitude line
1700       IF ABS(G)*10-INT(ABS(G)*10)>.5 THEN 1720
                                          'to determine the knots longitudes
1710    D=B+1 : B1=B-1 : D1=D+1 : GOTO 1730
                'the earthquake is closer to the left grid longitude line
1720    D=B-1 : B1=B+1 : D1=D-1
                'the earthquake is closer to the right grid longitude line
1730    MAG(A,B)=MAG(A,B)+E : MAG(C,D)=MAG(C,D)+E
1740    MAG(A,D)=MAG(A,D)+E : MAG(C,B)=MAG(C,B)+E
1750    MAG(C1,B)=MAG(C1,B)+E/4 : MAG(C1,D)=MAG(C1,D)+E/4
1760    MAG(C,B1)=MAG(C,B1)+E/4 : MAG(A,B1)=MAG(A,B1)+E/4
1770    MAG(A1,D)=MAG(A1,D)+E/4 : MAG(A1,B)=MAG(A1,B)+E/4
1780    MAG(A,D1)=MAG(A,D1)+E/4 : MAG(C,D1)=MAG(C,D1)+E/4
1790    GOTO 2100
1800 '
1810 '       Earthquake close to a grid knot
1820 '       -------------------------------
1830    MAG(A,B)=MAG(A,B)+E
1840    MAG(A+1,B)=MAG(A+1,B)+E/2 : MAG(A-1,B)=MAG(A-1,B)+E/2
1850    MAG(A,B+1)=MAG(A,B+1)+E/2 : MAG(A,B-1)=MAG(A,B-1)+E/2
1860    MAG(A+1,B+1)=MAG(A+1,B+1)+E/4 : MAG(A+1,B-1)=MAG(A+1,B-1)+E/4
1870    MAG(A-1,B+1)=MAG(A-1,B+1)+E/4 : MAG(A-1,B-1)=MAG(A-1,B-1)+E/4
1880    GOTO 2100
1890 '
1900 '       Earthquake close to a latitude line but not to a longitude line
1910 '       ---------------------------------------------------------------
1920       IF ABS(G)*10-INT(ABS(G)*10)>.5 THEN 1940
                                          'to determine the knots longitudes
1930    D=B+1 : B1=B-1 : D1=D+1 : GOTO 1950
                  'the earthquake is closer to the left longitude line
1940    D=B-1 : B1=B+1 : D1=D-1
                  'the earthquake is closer to the right longitude line
1950    MAG(A,B)=MAG(A,B)+E : MAG(A,D)=MAG(A,D)+E
1960    MAG(A+1,B)=MAG(A+1,B)+E/2 : MAG(A-1,B)=MAG(A-1,B)+E/2
1970    MAG(A+1,D)=MAG(A+1,D)+E/2 : MAG(A-1,D)=MAG(A-1,D)+E/2
1980    MAG(A,B1)=MAG(A,B1)+E/4 : MAG(A,D1)=MAG(A,D1)+E/4
1990    GOTO 2100
2000 '
2010 '       Earthquake close to a longitude line but not to a latitude line
2020 '       ---------------------------------------------------------------
2030       IF T1*10-INT(T1*10)<.5 THEN 2050  'to determine the knots latitudes
2040    C=A+1 : A1=A-1 : C1=C+1 : GOTO 2060
                  'the earthquake is closer to the lower latitude line
2050    C=A-1 : A1=A+1 : C1=C-1
                  'the earthquake is closer to the upper latitude line
2060    MAG(A,B)=MAG(A,B)+E : MAG(C,B)=MAG(C,B)+E
2070    MAG(A,B+1)=MAG(A,B+1)+E/2 : MAG(A,B-1)=MAG(A,B-1)+E/2
2080    MAG(C,B+1)=MAG(C,B+1)+E/2 : MAG(C,B-1)=MAG(C,B-1)+E/2
2090    MAG(A1,B)=MAG(A1,B)+E/4 : MAG(C1,B)=MAG(C1,B)+E/4
2100    X=0 : Y=0
2110 NEXT I%
2120 '---------------
```

```
2130 CLOSE #1
2140 '
2150 '    Converting the cumulated energy into magnitude
2160 '    ----------------------------------------------
2170 CLS : LOCATE 15,5 : PRINT "COMPUTING THE MAGNITUDE
2180 '---------------
2190 FOR I=1 TO LM
2200 '       ---------------
2210         FOR J=1 TO GM
2220               IF MAG(I,J)=0 THEN 2240
2230            MAG(I,J)=(LOG(MAG(I,J))/LOG(10))/2.57
                                    '2.57 from M. BATH (1982 & 1983)
2240         NEXT J
2250 '       ---------------
2260 NEXT I
2270 '-------------------------------------------------------------------------
2280 '
2290 '     PRINTING PART
2300 '     =============
2310 BEEP.: CLS : LOCATE 15,5
2320   INPUT "PRINTER READY (Y/N) ";R$ : R$=LEFT$(R$,1)
2330       IF R$<>"Y" AND R$<>"y" THEN 2320
2340 N=1 : IF GM>21 THEN N1=21 ELSE N1=GM
                          '21 is the maximum number of columns for 1 page
2350 '
2360 '    Printing the title, subtitle and characteristics
2370 '    ------------------------------------------------
2380 LPRINT CHR$(27)"n" TAB(34) "page";NP
2390 LPRINT CHR$(27)"!" : LPRINT CHR$(27)"2'R" : LPRINT
2400 LPRINT TAB((72-LEN(W$))/2) CHR$(27)"X" W$              'printing the title
2410 LPRINT CHR$(27)"N" : LPRINT CHR$(27)"Y" : LPRINT
2420 LPRINT TAB(5) S$                                       'printing the subtitle
2430 LPRINT CHR$(27)CHR$(34)
2440 LPRINT "Magnitude Choice : ";
2450       IF NM=0 THEN LPRINT "all magnitudes" : GOTO 2470
2460    FOR I=1 TO NM : LPRINT "[";MI(I);"-";MA(I);"], "; : NEXT I
2470 LPRINT
2480 LPRINT "Depth Choice : ";
2490       IF DN=0 THEN LPRINT "all depths" : GOTO 2510
2500    FOR I=1 TO DN : LPRINT "[";DI(I);"-";DA(I);"], "; : NEXT I
2510 LPRINT
2520 '
2530 '    Printing the longitudes
2540 '    -----------------------
2550 LPRINT CHR$(27)"0"
2560 LPRINT "      ";
2570 '---------------
2580 FOR I=N TO N1
2590    LG=(GM1*5-2+I)/5                        'longitude value
2600       IF LL=1 AND LG>180 THEN LG=ABS(360-LG) ELSE LG=ABS(LG)
2610    LPRINT USING "###.# ";LG;
2620 NEXT I
2630 '---------------
2640 LPRINT : LPRINT
2650 '
2660 '    Printing the latitudes and the data
2670 '    -----------------------------------
2680 '---------------
2690 FOR J=LM TO 1 STEP -1
2700    LPRINT USING "###.# ";(J+LATS*5-2)/5;       'latitude value
2710 '       ---------------
2720         FOR I=N TO N1
2730               IF MAG(J,I)=0 THEN LPRINT "   .    "; : GOTO 2750
2740            LPRINT USING "##.## ";MAG(J,I);
2750         NEXT I
2760 '       ---------------
2770    LPRINT
2780 NEXT J
2790 '---------------
2800 '
2810 '    Preparing the next page of the table
2820 '    ------------------------------------
2830 LPRINT CHR$(12)                'CHR$(12)=form feed to the top of next page
2840 N=N+21 : N1=N1+21 : NP=NP+1
2850        IF N1>=GM THEN N1=GM
2860        IF N>=GM THEN 2920
2870 LPRINT
2880 GOTO 2360
2890 '
2900 '    Preparing the next table
2910 '    ------------------------
2920 GM1=GM1+W : GM2=GM2+W            'W degrees is to assure an overlapping
   of the different tables (one table is W+0.8 degrees wide only because of
   limited size memory)
2930        IF GM2>LONGE THEN GM2=LONGE
2940        IF GM1>LONGE THEN 2960
2950 ERASE MAG : NTIMES=NTIMES+1 : GOTO 1140      'the table must be erased to
  create another one if the last one was not large enough to finish to cover
  the area
2960 END
```

B

★

Program to compute the average seismic magnitude in an area.

The data are read in a random access data file, created by the programs SEISEDIT and ERSQEDIT. The results are presented on a geographical-like table : the magnitude is computed for every knot of a 0.2 degree grid, if the value is zero the sign is a period.

The assignation of a magnitude to a knot of the grid is different if the earthquake has taken place very close to a knot, on a line between two knots or elsewhere. This program uses the method imagined by Markus BATH, modified to have a gaussian distribution of the magnitude of an earthquake on the grid.

A fan-folded paper is recommanded.

DISK READY (Y/N) ? Y

Data files on the disk :
ERSQUAKE.DAT MATPNTSS.DAT MATPNTFZ.DAT MATPNTTA.DAT MATPNTRM.DAT RIVPNTSS.DAT
RIVPNTPL.DAT DISCMAT3.DAT DISCMAT2.DAT MATPNTSI.DAT DISCMAT4.DAT DISCMATI.DAT
MATPNTPL.DAT RIVPOINT.DAT

What is your data file ? ERSQUAKE.DAT

Title : ? TEST

subtitle : ? For listing illustration

North latitude in degrees (use + for the northern hemisphere
and - for the southern hemisphere) : ? 21
South latitude : ? 14

West longitude in degrees (use + East of Greenwich and - West of
Greenwich) : ? -102
East longitude : ? -95
The width of the grid is 7 degrees. Is it OK (Y/N) ? Y

The program will have to read the data file on the disk 1
TIMES and to print 1 sets of results.

MAGNITUDE CHOICE : number of intervals (up to 10), type 0 if any
magnitude : ? 0

DEPTH CHOICE : number of intervals (up to 10), type 0 if all
depths (depths of 0 and 33 km mean undefined depth) : ? 0

The program will use the disk for the 1 TIME, are you ready (Y/N) ? Y

★

READING AND COMPUTING THE ENERGY

★

COMPUTING THE MAGNITUDE

★

PRINTER READY (Y/N) ? Y

C

TEST

For listing illustration

Magnitude Choice : all magnitudes

Depth Choice : all depths

	102.2	102.0	101.8	101.6	101.4	101.2	101.0	100.8	100.6	100.4	100.2	100.0	99.8	99.6	99.4	99.2	99.0	98.8	98.6	98.4	98.2
21.2	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
21.0	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
20.8	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	4.88	4.88	.	.	.	.
20.6	.	.	.	.	.	.	.	.	.	.	.	.	.	.	4.77	5.30	5.40	5.27	.	.	.
20.4	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	5.39	5.50	5.38	.	.	.
20.2	.	.	.	.	.	.	.	.	.	5.38	5.39	.	.	.	.	5.27	5.38	5.27	.	.	.
20.0	.	.	.	.	.	.	.	.	5.27	5.50	5.51	5.28	.	.	.	.	.	.	.	.	.
19.8	.	.	.	.	.	.	.	.	.	5.39	5.40	4.90	.	.	.	.	.	.	.	.	.
19.6	.	.	.	.	.	.	4.98	4.98	.	.	4.78	.	.	.	.	.	.	.	.	.	.
19.4	.	.	5.37	.	.	4.87	5.10	5.10	4.87	4.17	.	.	.	.	.	.	.	.	.	.	.
19.2	.	5.48	5.60	5.49	4.97	4.71	4.98	4.99	4.40	4.29	.	.	.	.	.	.	4.07	4.18	4.07	.	.
19.0	.	5.48	5.60	5.50	5.09	5.13	5.09	4.71	4.89	4.51	4.07	.	.	.	.	.	4.18	4.31	4.20	.	.
18.8	.	.	5.37	5.04	5.18	5.32	5.31	5.17	5.13	5.04	4.73	5.37	.	.	.	.	4.08	4.21	4.12	3.67	.
18.6	4.17	4.68	4.91	5.01	5.21	5.35	5.33	5.34	5.15	6.65	5.56	5.60	5.48	4.54	4.39	4.17	4.28	4.26	4.87	3.77	4.39
18.4	4.45	4.57	4.82	4.90	5.13	5.27	5.43	5.50	6.76	6.88	6.76	5.62	5.50	5.09	4.62	4.69	4.41	4.99	5.10	4.99	4.51
18.2	4.51	4.62	4.53	4.99	5.41	5.42	5.41	5.57	6.76	6.88	6.76	5.40	5.30	5.31	5.12	4.92	4.79	4.99	5.10	4.99	4.44
18.0	4.37	4.48	4.87	5.33	7.48	7.48	5.30	5.29	5.49	6.65	4.95	5.07	5.30	5.31	5.15	5.02	4.79	3.82	4.87	4.41	4.29
17.8	.	4.57	4.77	7.37	7.60	7.60	7.37	5.50	4.95	5.79	5.79	4.90	5.17	5.19	5.11	5.12	4.87	4.33	4.29	4.28	4.17
17.6	4.68	4.80	4.85	5.10	7.48	7.48	6.78	6.81	6.59	6.47	5.90	6.04	5.15	5.17	5.15	5.12	5.13	5.10	4.68	3.40	3.67
17.4	4.68	4.80	4.68	4.93	5.28	6.67	6.90	6.93	6.81	6.59	6.16	6.25	6.13	7.01	7.01	5.41	5.32	5.32	5.10	5.37	3.90
17.2	.	4.57	.	4.91	5.14	5.27	6.78	6.81	6.60	6.48	6.15	6.25	6.97	7.17	7.15	6.90	5.46	5.31	5.50	5.60	5.49
17.0	.	.	4.38	4.50	4.99	4.99	4.56	5.97	5.97	5.47	5.59	6.39	6.91	7.13	7.08	5.61	5.41	5.11	5.50	5.61	5.52
16.8	.	.	4.38	4.50	4.39	4.10	4.33	4.31	4.80	4.92	4.98	5.16	6.79	6.90	6.77	5.64	5.22	5.06	5.26	5.46	5.34
16.6	.	.	.	4.27	.	3.87	3.87	4.80	4.95	4.96	5.21	5.32	5.38	5.64	5.72	5.61	6.38	6.38	5.30	5.33	5.30
16.4	.	.	.	.	.	.	.	4.79	4.98	4.99	5.20	5.31	5.21	5.20	5.52	6.27	6.50	6.50	6.27	5.40	5.33
16.2	.	.	.	.	.	.	.	.	4.79	4.80	4.92	5.12	4.83	5.39	5.50	5.40	6.38	6.38	5.36	5.44	5.37
16.0	.	.	.	.	.	.	.	.	.	4.87	5.10	5.12	4.92	5.38	5.51	5.40	5.05	5.05	5.23	5.33	5.78
15.8	.	.	.	.	.	.	.	.	.	4.87	5.10	5.11	4.91	6.27	6.27	4.98	4.82	4.63	4.81	4.97	5.07
15.6	.	.	.	.	.	.	3.88	3.88	4.97	4.97	4.87	4.87	6.27	6.50	6.50	6.27	4.47	4.49	4.43	4.68	4.77
15.4	.	.	.	.	.	3.77	4.00	4.97	5.20	5.20	4.97	3.48	6.27	6.50	6.50	6.27	3.90	4.61	4.59	4.28	4.62
15.2	.	.	.	.	.	.	3.88	4.97	5.20	5.20	4.97	3.97	4.27	6.27	6.27	4.40	4.47	4.70	4.70	4.60	4.72
15.0	.	.	.	.	.	.	.	.	4.97	4.97	4.20	3.97	.	4.38	4.43	3.90	3.97	4.59	4.58	4.09	4.68
14.8	.	.	.	.	.	.	.	.	.	3.97	3.97	.	.	.	.	3.78	3.78	3.67	.	.	3.77
14.6	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	4.37
14.4	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	3.47	3.58	3.47	.	4.37	4.68
14.2	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	3.58	3.70	3.58	.	4.37	4.68
14.0	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	3.47	3.58	3.47	.	.	4.37
13.8	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.

TEST

For listing illustration

Magnitude Choice : all magnitudes

Depth Choice : all depths

	98.0	97.8	97.6	97.4	97.2	97.0	96.8	96.6	96.4	96.2	96.0	95.8	95.6	95.4	95.2	95.0	94.8
21.2	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
21.0	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
20.8	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
20.6	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
20.4	.	.	.	.	.	.	.	.	3.97	.	.	.	.	.	.	.	.
20.2	.	.	.	.	.	.	.	4.08	4.20	4.08	.	.	.	.	.	.	.
20.0	.	.	.	.	.	.	.	4.08	4.20	4.08	.	.	.	.	.	.	.
19.8	.	.	.	.	.	.	.	.	3.97	.	.	.	.	.	.	.	.
19.6	.	.	.	.	.	.	.	.	.	4.38	4.38	.	.	.	.	.	.
19.4	.	.	.	.	.	.	.	.	4.27	4.50	4.50	5.27	.	.	.	.	.
19.2	.	.	.	.	.	.	.	.	.	4.38	5.38	5.50	5.38	.	.	.	.
19.0	.	.	.	.	3.77	3.88	3.77	.	.	.	5.38	5.50	5.38	.	.	.	.
18.8	.	.	.	.	3.88	4.00	3.88	.	.	.	.	5.27	3.58	3.47	.	.	.
18.6	4.39	3.92	.	3.67	4.47	4.47	3.77	6.97	.	.	4.17	4.29	4.18	3.58	.	.	.
18.4	4.52	4.35	4.14	4.47	4.74	4.77	7.08	7.20	7.08	4.07	4.32	4.40	4.29	3.47	.	.	.
18.2	4.42	4.25	4.22	4.48	4.77	4.82	7.08	7.20	7.08	4.30	4.36	4.47	4.64	4.58	.	3.57	.
18.0	3.67	4.10	4.08	3.74	4.58	4.65	4.47	6.97	4.08	4.30	4.46	4.67	4.78	4.72	4.47	4.67	3.68
17.8	3.68	3.80	3.68	3.57	4.57	4.68	4.57	3.48	3.60	4.07	4.40	4.62	4.72	4.64	4.81	4.91	4.78
17.6	3.79	3.80	3.68	3.87	4.68	4.80	4.68	.	3.37	.	3.98	5.38	5.38	4.49	5.19	4.95	4.81
17.4	3.95	3.91	4.02	4.17	4.59	4.69	4.57	.	.	.	5.27	5.50	5.50	5.39	5.47	5.40	4.47
17.2	5.06	4.99	4.30	4.40	4.46	4.34	4.09	3.77	.	3.65	3.79	5.38	5.38	5.39	5.58	5.54	5.27
17.0	5.33	5.31	5.21	4.49	4.49	4.42	4.41	4.33	3.77	4.97	4.97	4.20	4.40	5.27	5.52	5.50	5.27
16.8	5.41	6.98	6.98	5.18	4.78	4.44	4.58	4.55	4.97	5.20	5.20	4.97	4.29	4.62	5.27	5.27	3.57
16.6	6.87	7.10	7.10	6.87	5.15	5.38	5.38	4.76	4.97	5.20	5.20	7.02	4.68	4.82	4.83	4.69	4.47
16.4	5.20	6.98	6.98	5.42	5.53	5.63	5.68	5.64	5.47	5.30	7.13	7.25	7.13	4.90	4.86	4.78	4.59
16.2	6.17	6.27	6.13	5.44	5.58	5.66	7.67	7.78	7.67	5.57	7.13	7.25	7.13	5.04	5.22	5.28	5.15
16.0	6.20	6.28	6.15	5.28	5.41	5.48	7.78	7.90	7.78	5.64	5.50	7.02	5.05	5.18	5.37	5.42	5.28
15.8	5.88	6.08	4.82	4.55	4.89	5.06	7.67	7.78	7.67	5.48	5.52	5.60	5.51	5.38	5.40	5.38	5.19
15.6	4.80	4.62	4.86	4.58	.	4.27	4.75	5.15	5.15	4.74	5.48	5.82	5.92	5.82	5.47	5.33	5.07
15.4	4.61	4.91	5.03	4.91	.	3.97	4.20	4.65	5.77	5.88	5.77	5.90	6.02	5.92	5.53	5.30	5.07
15.2	4.71	4.90	5.01	4.88	.	3.97	4.20	4.61	5.88	6.00	5.88	5.77	5.90	5.81	5.35	5.19	4.88
15.0	4.60	4.14	4.77	4.08	.	.	4.98	5.00	5.77	5.88	5.77	5.88	5.00	5.20	5.19	4.43	4.14
14.8	3.77	4.19	4.60	4.58	.	4.87	5.10	5.10	4.88	4.47	.	4.88	4.88	.	4.18	4.18	.
14.6	4.37	4.57	4.80	4.80	4.58	4.28	4.98	4.98	.	.	.	4.77	.	.	.	.	.
14.4	4.60	4.61	4.80	4.80	4.59	4.40	4.28	.	.	.	4.88	5.00	4.88	.	.	.	.
14.2	4.68	4.37	4.57	4.57	4.17	4.28	4.17	.	.	.	4.88	5.00	4.88	.	.	.	.
14.0	4.37	.	.	.	.	.	.	.	.	.	.	4.77	.	.	.	.	.
13.8	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.

10.4 A

```
10 ' ============================================================
20 '
30 '                          PROGRAM `SEISSORT'
40 '
50 '                                          Jacqueline ROUMP, Oct. 1984
60 ' ============================================================
70 '
80 CLS : CLEAR : OPTION BASE 1
90 '
100 '     PRESENTATION AND QUESTIONS
110 '     ==========================
120 PRINT "Program to sort earthquakes : criteria are geographical (polygon"
130 PRINT "with up to 10 sides), magnitude, and depth, and projection along"
140 PRINT "up to 5 directions given by two points. Geographical sort is from"
150 PRINT "`A Long Way from Euclid' by C. Reid, p. 174."
160 PRINT "The data have been collected in a sequential data file by the"
170 PRINT "program SEISEDIT (lat and long of the point in decimal degrees,"
180 PRINT "magnitude (Richter scale) and depth in km) before being put in a"
190 PRINT "random access data file by the program `ERSQEDIT'."
200 PRINT
210   INPUT "TITLE : ";TITLE$
220 PRINT
230   INPUT "NUMBER OF SIDES of the polygon (up to 10) : ";NS
240       IF NS>10 THEN PRINT "Only up to 10 sides" : GOTO 230
250 '
260 '    Latitude and longitude of each corner of the polygon
270 '    ---------------------------------------------------
280 '---------------
290 FOR I=1 TO NS
300    PRINT "       Corner ";I;" lat ";
310       INPUT T(I)
320       INPUT "                  long "; G(I)
330 NEXT I
340 '---------------
350 PRINT
360 '
370   INPUT "How many lines of projection (up to 5) : ";LP
380       IF LP>5 THEN PRINT "Only 5 different lines of projection" : GOTO 370
390 '
400 '    Latitude and longitude of the two points necessary for the definition
410 '    of the projection line
420 '    ------------------------------------------------------------------
430 PRINT
440 '---------------
450 FOR I=1 TO LP
460    PRINT "LINE OF PROJECTION ";I
470       INPUT "     Name (up to 4 letters)  ";LP$(I) : LP$(I)=LEFT$(LP$(I),4)
480       INPUT "     1st point (origin) : lat ";LAT(I)
490       INPUT "                          long ";LONG(I)
500       INPUT "     2nd point          : lat ";LAT(I+5)
510       INPUT "                          long ";LONG(I+5)
520    PRINT
530 NEXT I
540 '---------------
550 '
560 '    Choice of magnitude intervals
570 '    -----------------------------
580    INPUT "MAGNITUDE CHOICE : number of classes (up to 10), type 0 if any m
agnitude : ";NM
590       IF NM=0 THEN 690
600       IF NM>0 AND NM<10 THEN 630
610 PRINT "Integer number between 0 and 10" : GOTO 580
620 '---------------
630 FOR I=1 TO NM
640    PRINT "Interval ";I;" minimum included : ";
650       INPUT MI(I)
660       INPUT "              maximum excluded : ";MA(I)
670 NEXT I
680 '---------------
690 PRINT
700 '
710 '     Choice of depth intervals
720 '     -------------------------
730   INPUT "DEPTH CHOICE : number of classes (up to 10), type 0 if all depth
s : ";DN
740       IF DN=0 THEN 830
750       IF DN<0 OR DN>10 THEN PRINT "Number between 0 and 10" : GOTO 730
760 '---------------
770 FOR I=1 TO DN
780    PRINT "Interval ";I;" minimum included : ";
790       INPUT DI(I)
800       INPUT "              maximum excluded : ";DA(I)
810 NEXT I
820 '---------------
830 PRINT
840   INPUT "DISK READY (Y/N) ";R$ : R$=LEFT$(R$,1)
850       IF R$<>"Y" AND R$<>"y" THEN 840
860 PRINT : FILES "*.DAT"                 'list of the data files of the disk
870 PRINT : INPUT "Name of your data file ";X$
880 '-------------------------------------------------------------------------
890 '
900 '     COMPUTATION PART
910 '     ================
920 '    Definition of function and parameters for the transformation of
930 '    latitude into Y positions (Mercator, WGS72, projection)
940 '    -------------------------------------------------------
950 B#=SQR(.00666943178#)                 'square root of the Earth excentricity
960 FPI#=3.1415926539#/180                'PI/180
970 DEF FNX(X)=(LOG(TAN((45+X/2)*FPI#))-B#/2*LOG((1+B#*SIN(X*FPI#))/(1-B#*SIN(
X*FPI#))))/FPI#
980 '
990 '    Transformation of the corner longitudes into Y positions
1000 '    -------------------------------------------------------
1010 '---------------
1020 FOR I=1 TO NS
1030    LT(I)=FNX(T(I))
1040 NEXT I
1050 '---------------
1060 '
1070 '    Search for the minimum and maximum corners latitudes and longitudes
1080 '    -------------------------------------------------------------------
1090 TMAX=T(1) : TMIN=T(1) : GMAX=G(1) : GMIN=G(1) : LMAX=LT(1) : LMIN=LT(1)
1100 '---------------
1110 FOR I=2 TO NS
1120       IF LT(I)>LMAX THEN LMAX=LT(I) : TMAX=T(I)
```

```
1130        IF LT(I)<LMIN THEN LMIN=LT(I) : TMIN=T(I)
1140        IF G(I)>GMAX THEN GMAX=G(I)
1150        IF G(I)<GMIN THEN GMIN=G(I)
1160 NEXT I
1170 '---------------
1180 '
1190 '    SORT 1 : inside the lat-long interval, and the magnitude and depth
1200 '    intervals, creating table A
1210 '    ------------------------------------------------------------------
1220 DIM A(1500,5)                  '1500 is the optimum size for this program
1230 CLS : LOCATE 15,5
1240 PRINT "READING THE DATA FILE"   'informing the user of the file reading
1250 N=1 : NR=0
1260 OPEN "R",#1,X$,14
1270 FIELD #1,4 AS T$,4 AS G$,4 AS M$,2 AS P$
                                    'T$=lat, G$=Long, M$=magnitude, P$=depth
1280 '
1290 '   reading the number of value records
1300 '   ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
1310 GET #1, 1
1320 N=CVS(T$)
1330 '
1340 '   reading the value records
1350 '   ~~~~~~~~~~~~~~~~~~~~~~~~
1360 '---------------
1370 FOR I%=2 TO N+1
1380    GET #1,I%
1390    T=CVS(T$)
1400       IF T<TMIN OR T>TMAX THEN 1640
1410    G=CVS(G$)
1420       IF G<GMIN OR G>GMAX THEN 1640
1430    M=CVS(M$)
1440       IF MN=0 THEN 1510                    'all depths
1450 '       ---------------
1460        FOR I=1 TO MN
1470              IF M>=MI(I) AND M<MA(I) THEN 1510
1480        NEXT I
1490 '       ---------------
1500    GOTO 1640      'the earthquake is outside the chosen magnitude classes
1510    P=CVI(P$)
1520       IF DN=0 THEN 1590                    'all magnitudes
1530 '       ---------------
1540        FOR I=1 TO DN
1550              IF P>=DI(I) AND P<DA(I) THEN 1590
1560        NEXT I
1570 '       ---------------
1580    GOTO 1640          'the earthquake is outside the chosen depth classes
1590    NR=NR+1
1600    A(NR,1)=T                   'latitude in degrees
1610    A(NR,2)=G                   'longitude in degrees
1620    A(NR,3)=M                   'magnitude
1630    A(NR,4)=P                   'depth in km
1640 NEXT I%
1650 '---------------
1660 '
1670 '   end of the search in the random access file
1680 '   ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
1690 CLOSE #1                       'closing the data file
1700 '
1710 PRINT "Sort 1 :";NR-1;"earthquakes"         'number of earthquakes inside
     the rectangle of max and min latitudes and longitudes of the polygon
1720 '
1730 LOCATE 15,5 : PRINT "SEARCH FOR THE EARTHQUAKES INTO THE POLYGON"
1740 '
1750 '   transformation of the latitude of the selected data into Y position
1760 '   ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
1770 '---------------
1780 FOR I=1 TO NR
1790    A(I,1)=FNX(A(I,1))
1800 NEXT I
1810 '---------------
1820 '
1830 '    SORT 2 : inside the polygon, creating table B
1840 '    ---------------------------------------------
1850 '   def. of the parameters of the lines bearing the sides of the polygon
1860 '   ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
1870 '---------------
1880 FOR I=1 TO NS-1
1890        IF G(I)=G(I+1) THEN W(I)=1 : GOTO 1940
1900        IF G(I)>G(I+1) THEN X1(I)=G(I) : X2(I)=G(I+1) : GOTO 1920
1910    X1(I)=G(I+1) : X2(I)=G(I)
1920    C(I)=(LT(I)-LT(I+1))/(G(I)-G(I+1))     'a of y=a*x+b
1930    D(I)=LT(I)-G(I)*C(I)                   'b of y=a*x+b
1940 NEXT I
1950 '---------------
1960       IF G(1)=G(NS) THEN W(NS)=1 : GOTO 2010
1970       IF G(NS)>G(1) THEN X1(NS)=G(NS) : X2(NS)=G(1) : GOTO 1990
1980 X1(NS)=G(1) : X2(NS)=G(NS)
1990 C(NS)=(LT(NS)-LT(1))/(G(NS)-G(1))
2000 D(NS)=LT(NS)-G(NS)*C(NS)
2010 DIM B(450,9)                   'table of earthquakes inside the polygon
2020 '
2030 '    SORT 2 : test to compute how many times the earthquake is North of
2040 '    a polygon side, if the total number is odd the earthquake is inside
2050 '    the polygon, if it is even it is outside
2060 '    ------------------------------------------------------------------
2070 '---------------
2080 FOR I=1 TO NR
2090    S=0
2100 '       ---------------
2110         FOR J=1 TO NS
2120                IF W(J)=1 THEN 2160
2130                IF A(I,2)>X1(J) OR A(I,2)<X2(J) THEN 2160
2140             Y=C(J)*A(I,2)+D(J)                    'y=a*x+b
2150                IF Y<=A(I,1) THEN S=S+1
2160         NEXT J
2170 '       ---------------
2180       IF S MOD 2=0 THEN 2250
2190    H=H+1
2200 '       ---------------
2210         FOR K=1 TO 5
2220            B(H,K)=A(I,K)
2230         NEXT K
2240 '       ---------------
2250 NEXT I
2260 '---------------
```

```
2270 '
2280 PRINT "Sort 2 :";H;"earthquakes"  'number of earthquakes inside the polyg
on
2290 '
2300 LOCATE 15,5 : PRINT "COMPUTING THE DISTANCES ALONG THE PROJECTION LINES"
2310 '
2320 '    Def. of parameters and functions for the computation of distances
2330 '    using spherical trigonometry
2340 '    ------------------------------------------------------------------
2350 CT#=6378.135*FPI#                    'number of km per degree of long
2360 F#=.99333056822#            'correction factor for lat (WGS 72 ellipsoid)
2370 DEF FNA(X)=ATN(F#*TAN(X*FPI#))/FPI#   'corrected lat in degrees
2380 DEF FNB(X,Y,Z,W)=SIN(X*FPI#)*SIN(Z*FPI#)+COS(X*FPI#)*COS(Z*FPI#)*COS((W-Y
)*FPI#)              'distance between point A (lat = X, long = Y) and point B
                     (lat = Z, long = W)
2390 DEF FNC(X)=(-ATN(X/SQR(-X^2+1))+1.5708)/FPI#        'arc cosine
2400 '
2410 '    Computing the distance between the two points on the projection line
s
2420 '    ------------------------------------------------------------------
-
2430 DIM L(15)
2440 '---------------
2450 FOR I=1 TO LP
2460    L(I)=FNA(LAT(I))
2470    L(I+5)=FNA(LAT(I+5))
2480    L(I+10)=FNC(FNB(L(I),LONG(I),L(I+5),LONG(I+5)))
2490 NEXT I
2500 '---------------
2510 '
2520 '    Distance between the point from table A and the 2 points of the line
2530 '    of projection L1 and L7, angle between the line and the point L5,
2540 '    distance along the line from the origin to the projection in degrees
2550 '    L2, and in km B(I,5 to 10)
2560 '    ------------------------------------------------------------------
2570 '---------------
2580 FOR I=1 TO H
2590    L3=FNA(B(I,1))
2600 '         ---------------
2610          FOR J=1 TO LP
2620             L4=FNB(L(J),LONG(J),L3,B(I,2))
2630                IF L4=1 THEN B(I,J+4)=0 : GOTO 2740
2640             L1=FNC(L4)
2650             L6=FNB(L(J+5),LONG(J+5),L3,B(I,2))
2660                IF L6=1 THEN L2=L(J+10) : GOTO 2730
2670             L7=FNC(L6)
2680             D#=(COS(L7*FPI#)-COS(L1*FPI#)*COS(L(J+10)*FPI#))/(SIN(L1*FPI#
)*SIN(L(J+10)*FPI#))
2690                IF ABS(D#)).9999999999999999# THEN L5=0 : GOTO 2710
2700             L5=FNC(D#)
2710             L2=ATN(TAN(L1*FPI#)*COS(L5*FPI#))/FPI#
2720                IF L5=0 AND D#<0 THEN L2=-L2
2730             B(I,J+4)=L2*CT#
2740          NEXT J
2750 '         ---------------
2760 NEXT I
2770 '---------------
2780 '
2790 PRINT : BEEP
2800   INPUT "PRINTER READY (Y/N) ";R$
2810       IF LEFT$(R$,1)<>"Y" AND LEFT$(R$,1)<>"y" THEN 2790
2820 '-----------------------------------------------------------------------
2830 '
2840 '     PRINTING PART
2850 '     =============
2860 '    Preparing the printer and printing the title
2870 '    -------------------------------------------
2880 LPRINT CHR$(27)"n"CHR$(27)"!"CHR$(27)"L009"
2890 LPRINT TAB((72-9-4-LEN(TITLE$))/2) CHR$(27)"X" TITLE$
2900 '
2910 '    Printing the latitudes and longitudes of the polygon corners
2920 '    -----------------------------------------------------------
2930 LPRINT CHR$(27)"E" : LPRINT CHR$(27)"Y"
2940 LPRINT CHR$(27)"L012"CHR$(27)CHR$(34)
2950 LPRINT "CORNERS OF THE POLYGON :  No.    Lat       Long"
2960 LPRINT "                                 deg        deg"
2970 '---------------
2980 FOR I=1 TO NS
2990    LPRINT USING "                              #";I;
3000    LPRINT USING "   ###.###";T(I);
3010    LPRINT USING "   ####.###";G(I)
3020 NEXT I
3030 '---------------
3040 '
3050 '    Printing the lat and long of the projection lines points
3060 '    -------------------------------------------------------
3070 LPRINT CHR$(27)"q"
3080 LPRINT "PROJECTION LINE";
3090 '---------------
3100 FOR I=1 TO LP
3110    LPRINT SPC(5+(LEN(LP$(I)))/2);LP$(I);SPC(6+LEN(LP$(I))-(LEN(LP$(I)))/2
);
3120 NEXT I
3130 '---------------
3140 LPRINT
3150 LPRINT "                  ";
3160    FOR I=1 TO LP : LPRINT "Lat      Long      "; : NEXT I : LPRINT
3170 LPRINT "                  ";
3180    FOR I=1 TO LP : LPRINT "deg       deg      "; : NEXT I : LPRINT
3190 LPRINT "  Origin      ";
3200 '---------------
3210 FOR I=1 TO LP
3220    LPRINT USING "  ###.### ";LAT(I); : LPRINT USING " ####.###";LONG(I);
3230 NEXT I : LPRINT
3240 '---------------
3250 LPRINT "  Other point";
3260 '---------------
3270 FOR I=1 TO LP
3280    LPRINT USING "  ###.### ";LAT(I+5);
3290    LPRINT USING " ####.###";LONG(I+5);
3300 NEXT I : LPRINT
3310 '---------------
3320 '
3330 '    Printing the number of events
3340 '    -----------------------------
3350 LPRINT CHR$(27)"N"
```

```
3360 LPRINT H;" earthquakes" : LPRINT
3370 LPRINT " POINTS                                DISTANCES"
3380 LPRINT
3390 '
3400 '    Printing the results
3410 '    --------------------
3420 I1=1                           'number of the first line
3430 I2=33                          'number of the last line of the first page
3440 I3=1                           'page number
3450 IF H<I2 THEN I2=H              'test for the last line
3460 '
3470 '   columns heading and unit
3480 '   ~~~~~~~~~~~~~~~~~~~~~~~~
3490 LPRINT " No     Lat      Long    Mag  Dep";
3500    FOR J=1 TO LP : LPRINT "  ";LP$(J); : NEXT J : LPRINT
3510 LPRINT "         deg       deg          Km";
3520    FOR J=1 TO LP : LPRINT "   Km "; : NEXT J : LPRINT
3530 '
3540 '   printing the earthquake data and distances
3550 '   ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
3560 '---------------
3570 FOR I=I1 TO I2
3580    LPRINT USING "###  ";I;                          'number
3590    LPRINT USING "###.###  ";B(I,1);                 'latitude
3600    LPRINT USING "####.###  ";B(I,2);                'longitude
3610    LPRINT USING "#.##  ";B(I,3);                    'magnitude
3620    LPRINT USING "###  ";B(I,4);                     'depth
3630 '     ---------------
3640       FOR J=1 TO LP
3650          LPRINT USING "####  ";B(I,J+4);           'distances
3660       NEXT J
3670 '     ---------------
3680    LPRINT
3690 NEXT I
3700 '---------------
3710 '
3720 '   end of a page and beginning of the next one
3730 '   ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
3740       IF H=I2 THEN 3830
3750 I1=I2+1 : I2=I2+53 : I3=I3+1           'changing the page parameters
3760 PRINT
3770 PRINT "CHANGE THE PAPER FOR PAGE ";I3;", PRESS ANY KEY WHEN READY"
3780 A$=INKEY$ : IF A$="" THEN 3780
3790 LPRINT CHR$(27)"!" TITLE$ CHR$(27)CHR$(34) "     Page ";I3   'page title
3800 LPRINT
3810 GOTO 3450
3820 '
3830 END
```

B

★

Program to sort earthquakes : criteria are geographical (polygon with up to 10 sides), magnitude, and depth, and projection along up to 5 directions given by two points. Geographical sort is from 'A Long Way from Euclid' by C. Reid, p. 174.
The data have been collected in a sequential data file by the program SEISEDIT (lat and long of the point in decimal degrees, magnitude (Richter scale) and depth in km) before being put in a random access data file by the program 'ERSQEDIT'.

```
TITLE : ? MAT 6

NUMBER OF SIDES of the polygon (up to 10) : ? 4
       Corner  1  lat ? 15.397
                  long ? -102.35
       Corner  2  lat ? 20.587
                  long ? -97.501
       Corner  3  lat ? 20.36
                  long ? -96.413
       Corner  4  lat ? 15.097
                  long ? -101.33

How many lines of projection (up to 5) : ? 1

LINE OF PROJECTION  1
     Name (up to 4 letters)  ? _
     1st point (origin) : lat ? 16.5
                          long ? -100.68
     2nd point          : lat ? 20.155
                          long ? -97.265

MAGNITUDE CHOICE : number of classes (up to 10), type 0 if any magnitude : ? 0

DEPTH CHOICE : number of classes (up to 10), type 0 if all depths : ? 0

DISK READY (Y/N) ? Y

ERSQUAKE.DAT MATPNTSS.DAT MATPNTFZ.DAT MATPNTTR.DAT MATPNTRM.DAT RIVPNTSS.DAT
RIVPNTPL.DAT DISCMAT3.DAT DISCMAT2.DAT MATPNTSI.DAT DISCMAT4.DAT DISCMAT1.DAT
MATPNTPL.DAT RIVPOINT.DAT

Name of your data file ? ERSQUAKE.DAT
```

★

READING THE DATA FILE

★

SEARCH FOR THE EARTHQUAKES INTO THE POLYGON
Sort 1 : 571 earthquakes

★

COMPUTING THE DISTANCES ALONG THE PROJECTION LINES
Sort 2 : 82 earthquakess
PRINTER READY (Y/N) ? Y

a

CHANGE THE PAPER FOR PAGE 2 , PRESS ANY KEY WHEN READY

b

C
a

MAT 6

CORNERS OF THE POLYGON :

No.	Lat deg	Long deg
1	15.397	-102.350
2	20.587	-97.501
3	20.360	-96.413
4	15.097	-101.330

PROJECTION LINE

	Lat deg	Long deg
Origin	16.500	-100.680
Other point	20.155	-97.265

82 earthquakes

POINTS DISTANCES

No	Lat deg	Long deg	Mag	Dep km	km
1	17.143	-99.600	7.00	40	130
2	17.351	-99.500	7.13	33	154
3	17.455	-100.000	6.25	54	127
4	17.808	-100.273	5.90	41	137
5	17.322	-100.189	5.70	52	103
6	17.170	-99.735	6.60	36	122
7	18.499	-98.600	5.10	55	312
8	17.559	-100.200	5.10	91	122
9	17.768	-99.700	5.13	57	174
10	17.731	-100.235	5.10	60	134
11	17.826	-99.289	5.10	77	208
12	17.234	-99.811	5.20	50	122
13	18.326	-99.687	5.30	71	221
14	16.968	-99.777	5.00	51	103
15	17.514	-100.482	5.80	40	98
16	17.381	-99.507	5.20	35	156
17	16.664	-100.034	5.30	42	59
18	17.143	-99.600	4.60	51	130
19	17.351	-100.100	4.60	62	112
20	17.143	-99.600	4.80	33	130
21	16.529	-100.424	4.90	33	20
22	16.569	-100.358	4.50	33	28
23	18.226	-99.324	4.50	98	239
24	16.940	-100.219	4.60	54	69
25	17.052	-100.262	4.90	33	75
26	17.381	-99.470	4.50	33	158
27	16.740	-100.600	4.50	33	26
28	17.143	-100.077	4.70	33	96
29	16.607	-100.156	4.50	33	46
30	17.735	-99.779	4.80	80	166
31	17.360	-99.909	4.80	65	126
32	17.449	-100.004	4.50	78	126
33	18.249	-99.519	4.70	84	227

b MAT 6 Page 2

No	Lat deg	Long deg	Mag	Dep km	km
34	17.468	-99.315	4.90	41	177
35	18.282	-99.241	4.90	86	249
36	17.212	-99.792	4.80	75	122
37	17.190	-99.836	4.80	33	117
38	16.785	-99.953	4.70	33	75
39	16.604	-100.602	4.90	33	14
40	16.728	-100.300	4.20	33	46
41	17.351	-99.800	4.30	33	133
42	17.664	-99.800	4.00	33	159
43	16.832	-100.000	4.00	33	75
44	17.455	-100.400	4.40	33	99
45	17.768	-100.300	4.40	74	132
46	17.143	-99.700	4.10	33	123
47	17.143	-99.700	4.40	58	123
48	17.559	-99.900	4.10	98	143
49	17.247	-99.800	4.30	98	124
50	17.143	-100.800	4.10	56	45
51	16.832	-99.900	4.00	87	83
52	17.247	-99.900	4.40	47	117
53	17.247	-99.800	4.10	60	124
54	17.247	-99.500	4.30	33	145
55	18.604	-98.700	4.20	79	314
56	16.902	-100.431	4.10	40	51
57	16.963	-99.888	4.10	34	94
58	19.234	-98.841	4.30	6	356
59	17.770	-99.951	4.20	95	157
60	18.487	-99.517	4.30	112	247
61	17.127	-100.396	4.10	33	72
62	17.702	-100.100	4.40	73	141
63	17.355	-99.735	4.30	63	138
64	18.606	-99.047	4.40	102	290
65	17.070	-100.008	4.40	33	95
66	16.916	-99.913	4.40	59	89
67	17.072	-100.026	4.20	33	94
68	17.486	-99.610	4.30	80	157
69	17.450	-99.569	4.00	87	157
70	17.872	-99.500	3.60	117	197
71	19.024	-98.700	3.90	153	348
72	17.351	-100.300	3.80	33	98
73	17.977	-99.400	3.80	154	213
74	18.290	-99.000	3.60	163	267
75	18.290	-99.800	3.50	89	211
76	17.288	-99.845	3.80	36	124
77	17.455	-99.700	3.70	86	148
78	17.664	-99.300	3.70	118	194
79	17.089	-99.752	3.90	33	114
80	17.373	-100.497	3.80	33	86
81	17.737	-99.631	3.90	26	177
82	18.253	-98.697	3.80	136	285

10.5

A

```
10 REM ================================================================
20 REM
30 REM                          DATA  MAT 6'
40 REM
50 REM                                        JACQUELINE ROUMP, DEC. 1984
60 REM ================================================================
70 REM
80 REM DATA FOR THE PROGRAM  CROSSECTIONS'
90 REM
100 REM NUMBER OF CORNERS OF THE POLYGON
110 A4=4
120 DIM A3(A4,2)
130 REM LAT. AND LONG. OF THE CORNERS OF THE POLYGON
140 DATA 15.097,-101.33,20.36,-96.413,20.587,-97.501,15.397,-102.35
150 READ A3
160 REM NUMBER OF PROJECTION LINES
170 A6=1
180 DIM A5(A6,4)
190 REM LAT. AND LONG. OF THE ORIGIN & 2ND POINT OF THE PROJECTION LINE.
200 DATA 16.5,-100.68,20.155,-97.265
210 READ A5
220 REM
230 DIM A7(A6,2)
240 DATA -209,619
250 READ A7
260 REM EARTHQUAKE NUMBER
270 A0=82
280 A1=2+A6
290 REM EARTHQUAKE DATA : MAG, DEPTH, AND DISTANCES FROM ORIGIN TO LINES
300 DIM A(A0,A1)
310 DATA 7,40,118,7.13,33,141,6.25,54,114,5.9,41,123,5.7,52,90
320 DATA 6.6,36,110,5.1,55,296,5.1,91,109,5.13,57,160,5.1,60,120
330 DATA 5.1,77,194,5.2,50,110,5.3,71,206,5.51,91,5.8,40,85,5.2,35,143
340 DATA 5.3,42,49,4.6,51,118,4.6,62,99,4.8,33,118,4.9,33,10,4.5,33,18
350 DATA 4.5,98,223,4.6,54,58,4.9,33,64,4.5,33,146,4.5,33,15,4.7,33,84
360 DATA 4.5,33,36,4.8,80,152,4.8,65,113,4.5,78,114,4.7,84,211
370 DATA 4.9,41,164,4.9,86,233,4.8,75,110,4.8,33,105,4.7,33,64,4.9,33,4
380 DATA 4.2,33,35,4.3,33,120,4,33,145,4,33,65,4.4,33,86,4.4,74,118
390 DATA 4.1,33,111,4.4,58,111,4.1,98,130,4.3,98,112,4.1,56,33,4,87,72
400 DATA 4.4,47,105,4.1,60,112,4.3,33,133,4.2,79,297,4.1,40,40,4.1,34,83
410 DATA 4.3,6,337,4.2,95,143,4.3,112,230,4.1,33,60,4.4,73,127
420 DATA 4.3,63,125,4.4,102,273,4.4,33,83,4.4,59,77,4.2,33,82,4.3,80,144
430 DATA 4,87,144,3.6,117,183,3.9,153,330,3.8,33,85,3.8,154,198
440 DATA 3.6,163,251,3.5,89,195,3.8,36,112,3.7,86,135,3.7,118,180
450 DATA 3.9,33,103,3.8,33,73,3.9,26,163,3.8,136,270
460 READ A
```

B

```
10 REM  ================================================================
20 REM
30 REM                      PROGRAM  CROSSECTION'
40 REM
50 REM                                          JACQUELINE ROUMP, JAN. 1985
60 REM  ================================================================
70 PAGE
80 INIT
90 SET DEGREES
100 REM     PRESENTATION
110 REM     ============
120 CHARSIZE 4
130 PRINT "                    CROSSECTIONS"
140 PRINT
150 PRINT
160 CHARSIZE 3
170 PRINT "PROGRAM TO DRAW EARTHQUAKE CROSSECTIONS"
180 PRINT "THE DATA FILE MUST HAVE ALL THE NUMERICAL DATA CONCERNING"
190 PRINT "THE PLOT :"
200 PRINT "A TABLE A3 WITH A4 NUMBER OF CORNERS, READING THE LAT AND"
210 PRINT "LONG OF EACH CORNER."
220 PRINT "A TABLE A5 WITH A6 NUMBER OF PROJECTION LINES, READING THE"
230 PRINT "LAT AND LONG OF THE ORIGIN POINT AND THE SECOND POINT."
240 PRINT "A TABLE A7 READING THE MAXIMUM AND MINIMUM DISTANCES ON EACH"
250 PRINT "PROJECTION LINES OF THE CORNERS."
260 PRINT "A TABLE A WITH A0 EARTHQUAKES, READING THE MAGNITUDE, DEPTH,"
270 PRINT "AND DISTANCES ON THE PROJECTION LINES."
280 PRINT "THE SECTIONS WILL BE DRAWN ON A BENSON 1332 PLOTTER."
290 PRINT "THERE WILL BE UP TO 2 PAGES (29.7*42 CM FORMAT). THE FIRST"
300 PRINT "PAGE WILL HAVE THE TITLE, THE LEGEND, AND THE TWO FIRST"
310 PRINT "SECTIONS, THE SECOND PAGE THE REST OF THE SECTIONS."
320 PRINT "THE PLOTTER WILL USE 3 PENS :"
330 PRINT "     PEN 0 FOR THE LEGEND AND THE SECTIONS"
340 PRINT "     PEN 1 FOR THE TITLE AND SUBTITLES"
350 PRINT "   . PEN 2 FOR THE MAIN COMMENTS"
360 PRINT "THE THREE PENS MAY BE OF THE SAME COLOR OR OF 3 DIFFERENT"
370 PRINT "COLORS"
380 PRINT
390 PRINT "ALL THE COMPUTATIONS PREVIOUS TO THIS PLOT HAVE BEEN MADE"
400 PRINT "WITH VARIOUS PROGRAMS ON A ZENITH 100 COMPUTER."
410 PRINT
420 PRINT "WHAT IS THE NUMBER OF YOUR DATA FILE (IF ALREADY LOADED,"
430 PRINT "TYPE 0) ? ",
440 INPUT C
450 IF C=0 THEN 580
460 PRINT "IS THE TAPE READY (Y/N) ? ",
470 INPUT R$
480 R$=SEG(R$,1,1)
490 IF R$<>"Y" THEN 460
500 PRINT
510 FIND C
520 APPEND 580
530 PRINT
540 REM ----------------------------------------------------------------
550 REM
560 REM      PLACE FOR THE DATA FILE
570 REM
580 REM
590 REM ----------------------------------------------------------------
600 REM
```

```
610 REM     QUESTIONS
620 REM     =========
630 REM
640 REM   ASKING FOR THE TITLE, THE CROSSECTION NUMBER, THE SUBTITLES
650 REM   ------------------------------------------------------------
660 PRINT "TITLE OF THE PLOT ? ";
670 INPUT T$
680 PRINT
690 PRINT "THE TABLE HAS ";A6;" PROJECTION LINES"
700 PRINT "  HOW MANY CROSSECTIONS DO YOU WANT TO PLOT ? ";
710 INPUT N
720 IF N>A6 THEN 690
730 REM   SUBTITLES
740 DIM C0(N)
750 FOR I=1 TO N
760 PRINT "NUMBER OF THE CHOSEN CROSSECTION ";I;" ? ";
770 INPUT C0(I)
780 PRINT "SUBTITLE OF THE CROSSECTION ";I;" : ";
790 GO TO I OF 800,820,840,860,880
800 INPUT A$
810 GO TO 890
820 INPUT B$
830 GO TO 890
840 INPUT C$
850 GO TO 890
860 INPUT D$
870 GO TO 890
880 INPUT E$
890 NEXT I
900 REM
910 REM   SEARCH FOR THE MINIMUM AND MAXIMUM CORNER DISTANCES
920 REM   ---------------------------------------------------
930 M0=0
940 M9=0
950 FOR I=1 TO N
960 IF A7(C0(I),1)>M0 THEN 980
970 M0=A7(C0(I),1)
980 IF A7(C0(I),2)<M9 THEN 1000
990 M9=A7(C0(I),2)
1000 NEXT I
1010 PRINT
1020 PRINT "THE EXTREME MINIMUM AND MAXIMUM DISTANCES OF THE WHOLE"
1030 PRINT "CROSSECTIONS ARE : ";M0;" KM AND ";M9;" KM"
1040 PRINT "  WHAT MINIMUM AND MAXIMUM DISTANCES DO YOU WANT AS A FRAME"
1050 PRINT "  (EACH SECTION WILL BE DRAWN WITH ITS REAL LIMITS WITHIN"
1060 PRINT "  THE SAME FRAME TO ALLOW A COMMON SCALE) ? ";
1070 INPUT M1,M8
1080 IF M1<M8 THEN 1130
1090 M2=M1
1100 M1=M8
1110 M8=M2
1120 REM
1130 REM   SEARCH FOR THE MAXIMUM DEPTH
1140 REM   ----------------------------
1150 M7=A(1,2)
1160 FOR I=2 TO A0
1170 IF A(I,2)=33 THEN 1200
1180 IF A(I,2)<M7 THEN 1200
1190 M7=A(I,2)
1200 NEXT I
1210 PRINT
1220 PRINT "THE DEEPEST EARTHQUAKE IS AT ";M7;" KM"
1230 PRINT "  GIVE THE MAXIMUM DEPTH OF THE PLOT ? ";
1240 INPUT M7
1250 PRINT
1260 REM
1270 REM   ASKING FOR THE GRID CHARACTERISTICS
1280 REM   -----------------------------------
1290 REM
1300 REM   STEP TO PLOT
1310 PRINT "GIVE THE STEP TO DRAW THE CONTOUR OF THE SECTION"
1320 PRINT "(AN INTEGER NUMBER OF KM) : ";
1330 INPUT M5
1340 REM
1350 REM   LABELS AND TICK MARKS
1360 PRINT
1370 PRINT "DISTANCES IN ROUND KM BETWEEN TICK MARKS ALONG BOTH AXES (A"
1380 PRINT "MULTIPLE OF THE DRAWING STEP) ? ";
1390 INPUT M2
1400 IF M2/M5-INT(M2/M5)>0.01 THEN 1370
1410 PRINT "DISTANCES IN ROUND KM BETWEEN LABELS ALONG BOTH AXES (A"
1420 PRINT "MULTIPLE OF THE DISTANCE BETWEEN TICK MARKS) ? ";
1430 INPUT M3
1440 IF M3/M2-INT(M3/M2)>0.01 THEN 1410
1450 M2=INT(M2)
1460 M3=INT(M3)
1470 PRINT
1480 REM
1490 REM   ASKING FOR THE LIMITS, SIGNS AND COLORS OF EARTHQUAKE CLASSES
1500 REM   ------------------------------------------------------------
1510 M4=A(1,1)
1520 M6=M4
1530 FOR I=2 TO A0
1540 IF A(I,1)>M4 THEN 1570
1550 M4=A(I,1)
1560 GO TO 1590
1570 IF A(I,1)<M6 THEN 1590
1580 M6=A(I,1)
1590 NEXT I
1600 PRINT
1610 PRINT "THE MINIMUM AND MAXIMUM MAGNITUDES ARE : ";M4;" AND ";M6
1620 PRINT "  HOW MANY CLASSES OF MAGNITUDE DO YOU WANT (UP TO 10) ? ";
1630 INPUT N0
1640 DIM C3(N0,4)
1650 CHARSIZE 2
1660 FOR I=1 TO N0
1670 PRINT
1680 PRINT "CLASS ";I
1690 PRINT "    MINIMUM INCLUDED ? ";
1700 INPUT C3(I,1)
1710 PRINT "    MAXIMUM EXCLUDED ? ";
1720 INPUT C3(I,2)
1730 PRINT "    SIGN NUMBER CHOSEN AMONG THESE : "
1740 PRINT "1 : + ,   2 : * ,   3 : # ,   4 : $ ,   5 : o ,"
1750 PRINT "6 : 0 ,   7 : S ,   8 : X ,   9 : Z ,  10 : Ha"
1760 INPUT C3(I,3)
1770 PRINT "    IN WHICH COLOR (0 TO 2) ? ";
1780 INPUT C3(I,4)
1790 NEXT I
1800 REM ------------------------------------------------------------
```

```
1810 REM
1820 REM     COMMENTS LINES CREATION
1830 REM     ======================
1840 REM
1850 REM   LEGEND FOR THE MAGNITUDE CLASSES
1860 REM   --------------------------------
1870 FOR I=1 TO NO
1880 X$=" : ["
1890 GO TO I OF 1900,1930,1960,1990,2020,2050,2080,2110,2140,2170
1900 GOSUB 2210
1910 F$=X$&"["
1920 GO TO 2190
1930 GOSUB 2210
1940 G$=X$&"["
1950 GO TO 2190
1960 GOSUB 2210
1970 H$=X$&"["
1980 GO TO 2190
1990 GOSUB 2210
2000 I$=X$&"["
2010 GO TO 2190
2020 GOSUB 2210
2030 J$=X$&"["
2040 GO TO 2190
2050 GOSUB 2210
2060 K$=X$&"["
2070 GO TO 2190
2080 GOSUB 2210
2090 L$=X$&"["
2100 GO TO 2190
2110 GOSUB 2210
2120 M$=X$&"["
2130 GO TO 2190
2140 GOSUB 2210
2150 N$=X$&"["
2160 GO TO 2190
2170 GOSUB 2210
2180 O$=X$&"["
2190 NEXT I
2200 GO TO 2280
2210 Y$=STR(C3(I,1))
2220 X$=X$&Y$
2230 X$=X$&","
2240 Y$=STR(C3(I,2))
2250 X$=X$&Y$
2260 RETURN
2270 REM
2280 REM   CORNER LAT & LONG LINE
2290 REM   ----------------------
2300 P$="CORNERS : "
2310 Q$=" "
2320 S$=" "
2330 V$=""
2340 FOR I=1 TO A4
2350 GOSUB 2440
2360 GO TO I OF 2370,2370,2370,2390,2390,2390,2390,2410,2410,2410
2370 P$=P$&V$
2380 GO TO 2420
2390 Q$=Q$&V$
2400 GO TO 2420
2410 S$=S$&V$
2420 NEXT I
2430 GO TO 2780
2440 V$=""
2450 IF A3(I,1)<0 THEN 2480
2460 X$="N"
2470 GO TO 2490
2480 X$="S"
2490 V$=V$&X$
2500 X$=STR(ABS(A3(I,1)))
2510 X9=LEN(X$)
2520 X$=SEG(X$,2,X9-1)
2530 IF X9<=7 THEN 2550
2540 X$=SEG(X$,1,7)
2550 V$=V$&X$
2560 V$=V$&"-"
2570 IF A3(I,2)<0 THEN 2600
2580 X$="E"
2590 GO TO 2610
2600 X$="W"
2610 V$=V$&X$
2620 X$=STR(ABS(A3(I,2)))
2630 X9=LEN(X$)
2640 X$=SEG(X$,2,X9-1)
2650 IF X9<=8 THEN 2670
2660 X$=SEG(X$,1,7)
2670 V$=V$&X$
2680 IF I=3 THEN 2720
2690 IF I=7 THEN 2720
2700 IF I=10 THEN 2720
2710 V$=V$&"  "
2720 RETURN
2730 REM ------------------------------------------------------------------
2740 REM
2750 REM     COMPUTATION AND DRAWING PART
2760 REM     ===========================
2770 REM
2780 REM   COMPUTING THE POSITION OF THE SECTION
2790 REM   -------------------------------------
2800 L=2*6371*SIN((M9-M0)*90/10000/2)
2810 H=6371-(6371-M7)*COS((M9-M0)*90/10000/2)
2820 R=H/L
2830 IF R>0.38 THEN 2860
2840 S=L/21
2850 GO TO 2870
2860 S=H/8
2870 Z7=-1
2880 Y0=12.1
2890 Y1=-2.4-(8-H/S)/2
2900 REM
2910 REM   READYING THE PLOTTER
2920 REM   --------------------
2930 CHARSIZE 3
2940 PRINT " PLACE THE PEN HOLDERS AT THE LEFT END OF THE PLOTTER."
2950 PRINT "PLOTTER READY (Y/N) ? ",
2960 INPUT R$
2970 R$=SEG(R$,1,1)
2980 IF R$<>"Y" THEN 2950
2990 REM
3000 REM   PLOTTER INITIALIZATION
```

```
3010 GOSUB 7690
3020 REM
3030 REM   DRAWING THE FIRST FRAME
3040 REM   ----------------------
3050 Z2=0
3060 GOSUB 3100
3070 Z2=0
3080 GO TO 3270
3090 REM
3100 REM   SUBROUTINE TO DRAW A 29.7*42 CM FRAME
3110 REM   ------------------------------------
3120 Z0=0
3130 Z1=0
3140 GOSUB 7710
3150 Z2=1
3160 Z1=-42
3170 GOSUB 7700
3180 Z0=29.7
3190 GOSUB 7700
3200 Z1=0
3210 GOSUB 7700
3220 Z0=0
3230 GOSUB 7700
3240 Z2=0
3250 RETURN
3260 REM
3270 REM   TITLE AND MAIN COMMENTS PRINTING (TITLE DOUBLED)
3280 REM   -------------------------------------------------
3290 Z9=0
3300 Z4=0.6
3310 Z5=0.6
3320 GOSUB 7730
3330 Z6=1
3340 GOSUB 7740
3350 Z$=T$
3360 Z3=LEN(Z$)
3370 Z1=-2.8
3380 Z0=29.7/2-Z3/2*Z4
3390 GOSUB 5420
3400 Z0=Z0+0.02
3410 GOSUB 5420
3420 Z4=0.4
3430 Z5=0.4
3440 GOSUB 7730
3450 Z6=2
3460 GOSUB 7740
3470 Z1=-5.3
3480 Z0=3
3490 Z$="EARTHQUAKES CROSSECTIONS"
3500 GOSUB 5420
3510 Z$="NUMBER OF EARTHQUAKES : "
3520 V$=STR(A0)
3530 Z$=Z$&V$
3540 Z1=-6.4
3550 GOSUB 5420
3560 Z1=-7.5
3570 Z$=P$
3580 Z4=0.3
3590 Z5=0.35
3600 GOSUB 7730
3610 GOSUB 5420
3620 Z1=-8.1
3630 Z$=Q$
3640 GOSUB 5420
3650 Z1=-8.7
3660 Z$=S$
3670 GOSUB 5420
3680 REM
3690 REM   DRAWING OF THE FIRST SECTION
3700 REM   ----------------------------
3710 M=1
3720 Z0=3
3730 Z1=-9.7
3740 GOSUB 7700
3750 Z$=A$
3760 GOSUB 5490
3770 REM
3780 REM   DRAWING OF THE SECOND SECTION
3790 REM   -----------------------------
3800 IF N<2 THEN 3880
3810 M=2
3820 Z1=-11.9
3830 Z0=0
3840 GOSUB 7700
3850 Z$=B$
3860 GOSUB 5490
3870 REM
3880 REM   PRINTING OF THE LEGEND
3890 REM   ----------------------
3900 Z6=2
3910 GOSUB 7740
3920 Z4=0.4
3930 Z5=0.4
3940 GOSUB 7730
3950 Z$="MAGNITUDE CLASSES : "
3960 Z1=-12.6
3970 Z0=0
3980 GOSUB 5420
3990 Z4=0.35
4000 Z5=0.35
4010 GOSUB 7730
4020 Z6=0
4030 GOSUB 7740
4040 FOR I=1 TO NO
4050 GOSUB C3(I,3) OF 4210,4230,4250,4270,4290,4310,4330,4350,4370,4390
4060 GOSUB C3(I,4)+1 OF 4410,4430,4450
4070 GOSUB 7740
4080 GOSUB I OF 4470,4480,4490,4500,4510,4520,4530,4540,4550,4560
4090 GOSUB I OF 4170,4190,4170,4190,4170,4190,4170,4190,4170,4190
4100 GOSUB 5420
4110 Z6=0
4120 GOSUB 7740
4130 GOSUB I OF 4570,4600,4630,4660,4690,4720,4750,4780,4810,4840
4140 GOSUB 5420
4150 NEXT I
4160 GO TO 4900
4170 Z0=0
4180 RETURN
4190 Z0=9.1
4200 RETURN
```

```
4210 Z$="+"
4220 RETURN
4230 Z$="x"
4240 RETURN
4250 Z$="/"
4260 RETURN
4270 Z$="$"
4280 RETURN
4290 Z$="."
4300 RETURN
4310 Z$="O"
4320 RETURN
4330 Z$="S"
4340 RETURN
4350 Z$="X"
4360 RETURN
4370 Z$="Z"
4380 RETURN
4390 Z$="H"
4400 RETURN
4410 Z6=0
4420 RETURN
4430 Z6=1
4440 RETURN
4450 Z6=2
4460 RETURN
4470 Z1=-14
4480 RETURN
4490 Z1=-15
4500 RETURN
4510 Z1=-16
4520 RETURN
4530 Z1=-18
4540 RETURN
4550 Z1=-19
4560 RETURN
4570 Z$=F$
4580 Z0=0.7
4590 RETURN
4600 Z$=G$
4610 Z0=9.8
4620 RETURN
4630 Z$=H$
4640 Z0=0.7
4650 RETURN
4660 Z$=I$
4670 Z0=9.8
4680 RETURN
4690 Z$=J$
4700 Z0=0.7
4710 RETURN
4720 Z$=K$
4730 Z0=9.8
4740 RETURN
4750 Z$=L$
4760 Z0=0.7
4770 RETURN
4780 Z$=M$
4790 Z0=9.8
4800 RETURN
4810 Z$=N$
4820 Z0=0.7
4830 RETURN
4840 Z$=O$
4850 Z0=9.8
4860 RETURN
4870 REM
4880 REM   PRINTING THE DEPTH LINE COMMENT
4890 REM   ------------------------------
4900 Z$="ALL EARTHQUAKES PLOTTED AT THE SURFACE HAVE A NORMALIZED DEPTH"
4910 Z0=0
4920 Z1=Z1-1.3
4930 GOSUB 5420
4940 Z$="(0 OR 33 KM)"
4950 Z1=Z1-0.8
4960 GOSUB 5420
4970 IF N<3 THEN 5360
4980 REM
4990 REM    SECOND PAGE
5000 REM    -----------
5010 REM   GOING TO THE NEW ORIGIN AND DRAWING THE FRAME
5020 REM   ---------------------------------------------
5030 Z1=-23
5040 Z0=-2
5050 GOSUB 7700
5060 GOSUB 3100
5070 REM
5080 REM   THIRD SECTION
5090 REM   -------------
5100 M=3
5110 Z1=-2.9
5120 Z0=3
5130 GOSUB 7700
5140 Z$=C$
5150 GOSUB 5490
5160 IF N<4 THEN 5360
5170 REM
5180 REM   FOURTH SECTION
5190 REM   --------------
5200 M=4
5210 Z1=-11.9
5220 Z0=0
5230 GOSUB 7700
5240 Z$=D$
5250 GOSUB 5490
5260 IF N<>5 THEN 5360
5270 REM
5280 REM   FIFTH AND LAST SECTION
5290 REM   ----------------------
5300 M=5
5310 Z1=-11.9
5320 Z0=0
5330 GOSUB 7700
5340 Z$=E$
5350 GOSUB 5490
5360 END
5370 REM ------------------------------------------------------------------
5380 REM
5390 REM     SUBROUTINES
5400 REM     ===========
```

```
5410 REM
5420 REM   SUBROUTINE TO PRINT TEXT
5430 REM   ------------------------
5440 Z3=LEN(Z$)
5450 GOSUB 7700
5460 GOSUB 7720
5470 RETURN
5480 REM
5490 REM   SUBROUTINE TO DRAW A SECTION
5500 REM   ----------------------------
5510 Z0=0
5520 Z1=0
5530 GOSUB 7710
5540 REM
5550 REM   SUBTITLE PRINTING
5560 REM   - - - - - - - - -
5570 Z4=0.4
5580 Z5=0.4
5590 GOSUB 7730
5600 Z6=1
5610 GOSUB 7740
5620 Z1=-0.4
5630 GOSUB 5420
5640 REM
5650 REM   PROJECTION LINE COMMENT CREATION AND PRINTING
5660 REM   - - - - - - - - - - - - - - - - - - - - - - -
5670 Z$="PROJECTION LINE :  ORIGIN "
5680 F9=0
5690 GOSUB 5760
5700 Z$=Z$&V$
5710 Z$=Z$&"    SECOND POINT "
5720 F9=2
5730 GOSUB 5760
5740 Z$=Z$&V$
5750 GO TO 6010
5760 V$=""
5770 IF A5(M,1+F9)<0 THEN 5800
5780 X$="N"
5790 GO TO 5810
5800 X$="S"
5810 V$=V$&X$
5820 X$=STR(ABS(A5(M,1+F9)))
5830 X9=LEN(X$)
5840 X$=SEG(X$,2,X9-1)
5850 IF X9<=7 THEN 5870
5860 X$=SEG(X$,1,7)
5870 V$=V$&X$
5880 V$=V$&"-"
5890 IF A5(M,2+F9)<0 THEN 5920
5900 X$="E"
5910 GO TO 5930
5920 X$="W"
5930 V$=V$&X$
5940 X$=STR(ABS(A5(M,2+F9)))
5950 X9=LEN(X$)
5960 X$=SEG(X$,2,X9-1)
5970 IF X9<=8 THEN 5990
5980 X$=SEG(X$,1,7)
5990 V$=V$&X$
6000 RETURN
```

```
6010 Z1=-1.2
6020 Z4=0.3
6030 Z5=0.35
6040 GOSUB 7730
6050 Z6=2
6060 GOSUB 7740
6070 GOSUB 5420
6080 REM
6090 REM   SECTION DRAWING
6100 REM   - - - - - - - -
6110 Z4=0.3
6120 Z5=0.3
6130 GOSUB 7730
6140 Z6=0
6150 GOSUB 7740
6160 Z0=Y0
6170 Z1=Y1
6180 N5=INT(M9)-INT(M0)+1
6190 D3=M0-((M8-M1)/2+M1)
6200 N6=INT(M7)
6210 REM
6220 REM   TOP OF THE SECTION
6230 D6=0
6240 K=1
6250 GOSUB 6390
6260 REM
6270 REM   RIGHT SIDE
6280 GOSUB 6850
6290 REM
6300 REM   LEFT SIDE
6310 D6=M7
6320 K=-1
6330 GOSUB 6850
6340 REM
6350 REM   BOTTOM OF THE SECTION
6360 GOSUB 6390
6370 GO TO 7340
6380 REM
6390 REM   TOP AND BOTTOM DRAWING SUBROUTINE
6400 REM   - - - - - - - - - - - - - - - - -
6410 Z0=Y0+(6371-D6)*SIN(D3*90/10000)/S
6420 Z1=Y1-(6371-(6371-D6)*COS(D3*90/10000))/S
6430 GOSUB 7700
6440 Z2=1
6450 D3=INT(M0/M5+1)*M5-((M8-M1)/2+M1)
6460 D7=INT(M0/M5+1)*M5
6470 Z0=Y0+(6371-D6)*SIN(D3*90/10000)/S
6480 Z0=Y1-(6371-(6371-D6)*COS(D3*90/10000))/S
6490 FOR I=0 TO N5 STEP M5
6500 E1=(D3+I)*90/10000
6510 Z0=Y0+(6371-D6)*SIN(E1)/S
6520 Z1=Y1-(6371-(6371-D6)*COS(E1))/S
6530 GOSUB 7700
6540 REM
6550 IF ABS((D7+I)/M2-INT((D7+I)/M2))>0.1 THEN 6790
6560 Z1=Z1+K*0.2*COS(E1)
6570 Z0=Z0+K*0.2*SIN(E1)
6580 GOSUB 7700
6590 IF ABS((D7+I)/M3-INT((D7+I)/M3))>0.1 THEN 6750
6600 Z2=0
```

```
6610 Z8=D7+I
6620 Z$=STR(Z8)
6630 Z3=LEN(Z$)
6640 Z0=Z0-0.1*SIN(E1)-(Z3-1)/2*Z4
6650 Z1=Z1+K*0.1*COS(E1)
6660 IF K=1 THEN 6680
6670 Z1=Z1-Z5
6680 GOSUB 7700
6690 GOSUB 7750
6700 IF K=1 THEN 6720
6710 Z1=Z1+Z5
6720 Z0=Z0+0.1*SIN(E1)+(Z3-1)/2*Z4
6730 Z1=Z1-K*0.1*COS(E1)
6740 GOSUB 7700
6750 Z2=1
6760 Z0=Y0+(6371-D6)*SIN(E1)/S
6770 Z1=Y1-(6371-(6371-D6)*COS(E1))/S
6780 GOSUB 7700
6790 NEXT I
6800 Z0=Y0+(6371-D6)*SIN((D3+(M9-M0))*90/10000)/S
6810 Z1=Y1-(6371-(6371-D6)*COS((D3+M9-M0)*90/10000))/S
6820 GOSUB 7700
6830 Z2=0
6840 RETURN
6850 REM   LEFT AND RIGHT SIDES SECTION DRAWING SUBROUTINE
6860 REM   - - - - - - - - - - - - - - - - - - - - - - - -
6870 N6=INT(M7)
6880 IF K=-1 THEN 6910
6890 L=M9
6900 GO TO 6920
6910 L=M0
6920 D9=(L-((M8-M1)/2+M1))*90/10000
6930 D4=SIN(D9)
6940 D5=COS(D9)
6950 Z0=Y0+6371*D4/S
6960 Z1=Y1-(6371-6371*D5)/S
6970 GOSUB 7700
6980 Z2=1
6990 FOR I=0 TO N6 STEP M5
7000 Z0=Y0+(6371-I)*D4/S
7010 Z1=Y1-(6371-(6371-I)*D5)/S
7020 GOSUB 7700
7030 IF ABS(I/M2-INT(I/M2))>0.1 THEN 7270
7040 Z1=Z1-K*0.2*D4
7050 Z0=Z0+K*0.2*D5
7060 GOSUB 7700
7070 IF ABS(I/M3-INT(I/M3))>0.1 THEN 7230
7080 Z2=0
7090 Z8=I
7100 IF K=1 THEN 7150
7110 Z$=STR(I)
7120 Z3=LEN(Z$)
7130 Z0=Z0+K*0.1*D5-Z3*Z4
7140 GO TO 7160
7150 Z0=Z0+K*0.1*D5
7160 GOSUB 7700
7170 GOSUB 7750
7180 IF K=1 THEN 7210
7190 Z0=Z0-K*0.1*D5+Z3*Z4
7200 GO TO 7220
7210 Z0=Z0-K*0.1*D5
7220 GOSUB 7700
7230 Z2=1
7240 Z0=Y0+(6371-I)*D4/S
7250 Z1=Y1-(6371-(6371-I)*D5)/S
7260 GOSUB 7700
7270 NEXT I
7280 Z0=Y0+(6371-M7)*D4/S
7290 Z1=Y1-(6371-(6371-M7)*D5)/S
7300 GOSUB 7700
7310 Z2=0
7320 RETURN
7330 REM
7340 REM   EARTHQUAKES PLOTTING
7350 REM   - - - - - - - - - - -
7360 FOR I=1 TO A0
7370 IF A(I,2)<>33 THEN 7390
7380 A(I,2)=0
7390 IF A(I,2)>M7 THEN 7570
7400 FOR J=1 TO N0
7410 IF A(I,1)<C3(J,1) THEN 7570
7420 IF A(I,1)=>C3(J,2) THEN 7450
7430 G=J
7440 GO TO 7460
7450 NEXT J
7460 IF A(I,M+2)<M1 THEN 7570
7470 IF A(I,M+2)>M8 THEN 7570
7480 E1=(A(I,M+2)-((M8-M1)/2+M1))*90/10000
7490 Z0=Y0+(6371-A(I,2))*SIN(E1)/S
7500 Z1=Y1-(6371-(6371-A(I,2))*COS(E1))/S
7510 Z0=Z0-0.15
7520 GOSUB C3(G,3) OF 7590,7590,7610,7610,7630,7610,7610,7610,7610,7610
7530 GOSUB C3(G,3) OF 4210,4230,4250,4270,4290,4310,4330,4350,4370,4390
7540 GOSUB C3(G,4)+1 OF 4410,4430,4450
7550 GOSUB 7740
7560 GOSUB 5420
7570 NEXT I
7580 RETURN
7590 Z1=Z1-0.12
7600 RETURN
7610 Z1=Z1-0.15
7620 RETURN
7630 Z1=Z1-0.06
7640 RETURN
7650 REM ------------------------------------------------------------
7660 REM
7670 REM      BENSON PLOTTER SUBROUTINES
7680 REM      ==========================
7690 GO TO 7790
7700 GO TO 8030
7710 GO TO 8320
7720 GO TO 8380
7730 GO TO 8480
7740 GO TO 8620
7750 GO TO 8710
7760 REM
7770 REM   INITIALIZATION OF THE BENSON 1332 PLOTTER
7780 REM   ----------------------------------------
7790 DIM Z(25),Z$(80)
7800 ON SRQ THEN 7990
```

```
7810 PRINT "BENSON ADDRESS ? ";
7820 INPUT Z(25)
7830 PRINT @Z(25),"76 1";
7840 Z$=CHR(34)
7850 PRINT @Z(25),Z$;"#$%&'()*+,-./0123456789:;<=>?";
7860 PRINT @Z(25),"@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]↑←";
7870 PRINT @Z(25), USING "A/L",Z$
7880 PRINT @Z(25),"76                                    ";
7890 PRINT @Z(25),"                                      ";
7900 PRINT @Z(25), USING "A/L",Z$
7910 Z(1)=0
7920 Z(2)=0
7930 Z(3)=200
7940 Z(5)=60
7950 Z(6)=0
7960 Z(13)=0
7970 Z(20)=Z(3)
7980 Z(21)=Z(3)
7990 RETURN
8000 REM
8010 REM   MOVING THE PEN
8020 REM   -------------
8030 Z(7)=INT(Z0*Z(20))
8040 Z(8)=INT(Z1*Z(21))
8050 Z(9)=Z(7)-Z(1)
8060 Z(10)=Z(8)-Z(2)
8070 Z(1)=Z(7)
8080 Z(2)=Z(8)
8090 Z(7)=ABS(Z(9))
8100 Z(8)=ABS(Z(10))
8110 IF Z(7)>16383 THEN 8200
8120 IF Z(8)>16383 THEN 8200
8130 Z(7)=0
8140 Z(11)=Z(9)
8150 Z(12)=Z(10)
8160 PRINT @Z(25),"PL ";Z(11),Z(12),Z2,
8170 PRINT @Z(25), USING "A/L",Z$
8180 IF Z(7)>1 THEN 8090
8190 RETURN
8200 Z(7)=INT((Z(7)+16383)/16384)
8210 Z(8)=INT((Z(8)+16383)/16384)
8220 IF Z(7)>Z(8) THEN 8240
8230 Z(7)=Z(8)
8240 Z(11)=INT(Z(9)/Z(7))
8250 Z(12)=INT(Z(10)/Z(7))
8260 Z(9)=Z(9)-Z(11)
8270 Z(10)=Z(10)-Z(12)
8280 GO TO 8160
8290 REM
8300 REM   CHANGING THE ORIGIN
8310 REM   -------------------
8320 Z(1)=INT(Z0*Z(20))
8330 Z(2)=INT(Z1*Z(21))
8340 RETURN
8350 REM
8360 REM   WRITING A STRING OF CHARACTERS
8370 REM   -------------------------------
8380 PRINT @Z(25),"TX ";Z3," ";Z$,
8390 PRINT @Z(25), USING "A/L",Z$
8400 Z(1)=Z(1)+Z(5)*Z3
8410 Z(2)=Z(2)+Z(6)*Z3
8420 PRINT @Z(25),"NT ";Z(5)*Z3," ";Z(6)*Z3,
8430 PRINT @Z(25), USING "A/L",Z$
8440 RETURN
8450 REM
8460 REM   INITIALIZATION OF CHARACTER TYPE AND ORIENTATION
8470 REM   -----------------------------------------------
8480 Z(5)=Z4*COS(Z9)
8490 Z(6)=Z4*SIN(Z9)
8500 Z(16)=INT(4*Z(5)*Z(20)/3)
8510 Z(17)=INT(-4*Z5*Z(20)*SIN(Z9)/3)
8520 Z(18)=INT(4*Z(6)*Z(21)/3)
8530 Z(19)=INT(4*Z5*Z(21)*COS(Z9)/3)
8540 PRINT @Z(25),"SZ ";Z(16)," ";Z(17)," ";Z(18)," ";Z(19),
8550 PRINT @Z(25), USING "A/L",Z$
8560 Z(5)=INT(Z(5)*Z(20))
8570 Z(6)=INT(Z(6)*Z(21))
8580 RETURN
8590 REM
8600 REM   CHANGING THE PEN
8610 REM   ----------------
8620 Z(14)=INT(ABS(Z6))-3*INT(INT(ABS(Z6))/3)
8630 IF Z(14)=Z(13) THEN 8670
8640 PRINT @Z(25),"NP ";Z(14),
8650 PRINT @Z(25), USING "A/L",Z$
8660 Z(13)=Z(14)
8670 RETURN
8680 REM
8690 REM   PRINTING OF NUMBERS
8700 REM   -------------------
8710 Z7=INT(Z7)
8720 Z(9)=ABS(INT(Z7))
8730 IF Z7>-2 THEN 8910
8740 IF Z(9)<10 THEN 8760
8750 Z(9)=9
8760 Z$=STR(Z(9))
8770 Z$=Z$&"E"
8780 Z(9)=Z(9)+7
8790 IF Z8>0 THEN 8830
8800 Z(9)=Z(9)+1
8810 PRINT @Z(25),"TX ";Z(9)," ";
8820 GO TO 8840
8830 PRINT @Z(25),"TX ";Z(9),
8840 PRINT @Z(25), USING Z$,Z8
8850 PRINT @Z(25), USING "A/L",Z$
8860 Z(1)=Z(1)+Z(9)*Z(5)
8870 Z(2)=Z(2)+Z(9)*Z(6)
8880 PRINT @Z(25),"NT ";Z(9)*Z(5)," ";Z(9)*Z(6),
8890 PRINT @Z(25), USING "A/L",Z$
8900 RETURN
8910 Z(7)=INT(LGT(ABS(Z8)+0.5))+1
8920 IF Z(7)>0 THEN 8940
8930 Z(7)=1
8940 IF Z8=>0 THEN 8960
8950 Z(7)=Z(7)+1
8960 Z$=STR(Z(7))
8970 Z$=Z$&"D"
8980 IF Z7>-1 THEN 9010
8990 Z(9)=Z(7)
9000 GO TO 8810
```

```
9010 Z(8)=LEN(Z$)+2
9020 Z$=Z$&". 0"
9030 IF Z(9)<10 THEN 9050
9040 Z(9)=9
9050 GO TO Z(9)+1 OF 9060,9080,9100,9120,9140,9160,9180,9200,9220,9240
9060 Z$=REP(" ",Z(8)+1,1)
9070 GO TO 9250
9080 Z$=REP("1",Z(8),1)
9090 GO TO 9250
9100 Z$=REP("2",Z(8),1)
9110 GO TO 9250
9120 Z$=REP("3",Z(8),1)
9130 GO TO 9250
9140 Z$=REP("4",Z(8),1)
9150 GO TO 9250
9160 Z$=REP("5",Z(8),1)
9170 GO TO 9250
9180 Z$=REP("6",Z(8),1)
9190 GO TO 9250
9200 Z$=REP("7",Z(8),1)
9210 GO TO 9250
9220 Z$=REP("8",Z(8),1)
9230 GO TO 9250
9240 Z$=REP("9",Z(8),1)
9250 Z(9)=Z(9)+Z(7)+1
9260 GO TO 8810
```

C
★

CROSSECTIONS

PROGRAM TO DRAW EARTHQUAKE CROSSECTIONS
THE DATA FILE MUST HAVE ALL THE NUMERICAL DATA CONCERNING THE PLOT :
A TABLE A3 WITH A4 NUMBER OF CORNERS, READING THE LAT AND LONG OF EACH CORNER.
A TABLE A5 WITH A6 NUMBER OF PROJECTION LINES, READING THE LAT AND LONG OF THE ORIGIN POINT AND THE SECOND POINT.
A TABLE A7 READING THE MAXIMUM AND MINIMUM DISTANCES ON EACH PROJECTION LINES OF THE CORNERS.
A TABLE A WITH A0 EARTHQUAKES, READING THE MAGNITUDE, DEPTH, AND DISTANCES ON THE PROJECTION LINES.
THE SECTIONS WILL BE DRAWN ON A BENSON 1332 PLOTTER.
THERE WILL BE UP TO 2 PAGES (29.7*42 CM FORMAT). THE FIRST PAGE WILL HAVE THE TITLE, THE LEGEND, AND THE TWO FIRST SECTIONS. THE SECOND PAGE THE REST OF THE SECTIONS.
THE PLOTTER WILL USE 3 PENS :
    PEN 0 FOR THE LEGEND AND THE SECTIONS
    PEN 1 FOR THE TITLE AND SUBTITLES
    PEN 2 FOR THE MAIN COMMENTS
THE THREE PENS MAY BE OF THE SAME COLOR OR OF 3 DIFFERENT COLORS

ALL THE COMPUTATIONS PREVIOUS TO THIS PLOT HAVE BEEN MADE WITH VARIOUS PROGRAMS ON A ZENITH 100 COMPUTER.

WHAT IS THE NUMBER OF YOUR DATA FILE (IF ALREADY LOADED, TYPE 0) ? 2
IS THE TAPE READY (Y/N) ? Y

TITLE OF THE PLOT ? MIDDLE AMERICA TRENCH 6

THE TABLE HAS 1 PROJECTION LINES
  HOW MANY CROSSECTIONS DO YOU WANT TO PLOT ? 1
NUMBER OF THE CHOSEN CROSSECTION 1 ? 1
SUBTITLE OF THE CROSSECTION 1 : MAT 6

THE EXTREME MINIMUM AND MAXIMUM DISTANCES OF THE WHOLE CROSSECTIONS ARE : -209 KM AND 619 KM
  WHAT MINIMUM AND MAXIMUM DISTANCES DO YOU WANT AS A FRAME (EACH SECTION WILL BE DRAWN WITH ITS REAL LIMITS WITHIN THE SAME FRAME TO ALLOW A COMMON SCALE) ? -300,700

THE DEEPEST EARTHQUAKE IS AT 163 KM
  GIVE THE MAXIMUM DEPTH OF THE PLOT ? 350

GIVE THE STEP TO DRAW THE CONTOUR OF THE SECTION
(AN INTEGER NUMBER OF KM) : 5

DISTANCES IN ROUND KM BETWEEN TICK MARKS ALONG BOTH AXES (A MULTIPLE OF THE DRAWING STEP) ? 20
DISTANCES IN ROUND KM BETWEEN LABELS ALONG BOTH AXES (A MULTIPLE OF THE DISTANCE BETWEEN TICK MARKS) ? 100

THE MINIMUM AND MAXIMUM MAGNITUDES ARE : 3.5 AND 7.13
  HOW MANY CLASSES OF MAGNITUDE DO YOU WANT (UP TO 10) ? 4

```
CLASS 1
    MINIMUM INCLUDED ? 3
    MAXIMUM EXCLUDED ? 4
    SIGN NUMBER CHOSEN AMONG THESE :
1 : +  ;  2 : #  ;  3 : *  ;  4 : o  ;  5 : .  ;
6 : 0  ;  7 : S  ;  8 : X  ;  9 : Z  ;  10 : H.
5
    IN WHICH COLOR (0 TO 2) ? 0

CLASS 2
    MINIMUM INCLUDED ? 4
    MAXIMUM EXCLUDED ? 4.5
    SIGN NUMBER CHOSEN AMONG THESE :
1 : +  ;  2 : #  ;  3 : *  ;  4 : o  ;  5 : .  ;
6 : 0  ;  7 : S  ;  8 : X  ;  9 : Z  ;  10 : H.
1
    IN WHICH COLOR (0 TO 2) ? 0

CLASS 3
    MINIMUM INCLUDED ? 4.5
    MAXIMUM EXCLUDED ? 5
    SIGN NUMBER CHOSEN AMONG THESE :
1 : +  ;  2 : #  ;  3 : *  ;  4 : o  ;  5 : .  ;
6 : 0  ;  7 : S  ;  8 : X  ;  9 : Z  ;  10 : H.
2
    IN WHICH COLOR (0 TO 2) ? 0

CLASS 4
    MINIMUM INCLUDED ? 5
    MAXIMUM EXCLUDED ? 8
    SIGN NUMBER CHOSEN AMONG THESE :
1 : +  ;  2 : #  ;  3 : *  ;  4 : o  ;  5 : .  ;
6 : 0  ;  7 : S  ;  8 : X  ;  9 : Z  ;  10 : H.
3
    IN WHICH COLOR (0 TO 2) ? 0
 PLACE THE PEN HOLDERS AT THE LEFT END OF THE PLOTTER
PLOTTER READY (Y/N) ? Y
BENSON ADDRESS ? 41
```

D MIDDLE AMERICA TRENCH 6

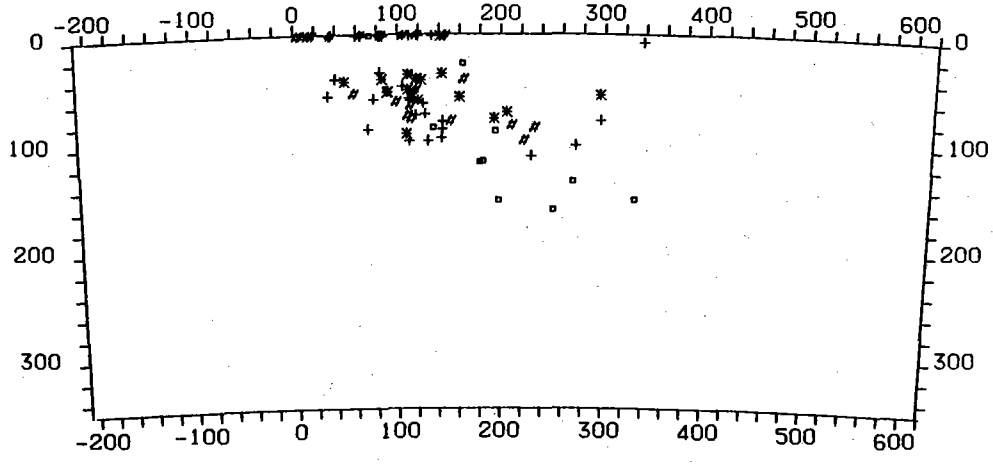
EARTHQUAKES CROSSECTIONS

NUMBER OF EARTHQUAKES : 82

CORNERS : N15.097-W101.33 N20.36-W96.413 N20.587-W97.501
N15.397-W102.35

MAT 6

PROJECTION LINE : ORIGIN N16.5-W100.68 SECOND POINT N20.155-W97.265



MAGNITUDE CLASSES :

o : [3, 4[+ : [4, 4.5[

: [4.5, 5[* : [5, 8[

ALL EARTHQUAKES PLOTTED AT THE SURFACE HAVE A NORMALIZED DEPTH (0 OR 33 KM)