PIT SLOPE MANUAL

supplement 2-1

DISCODAT PROGRAM PACKAGE

This supplement has been prepared as part of the

PIT SLOPE PROJECT

of the
Mining Research Laboratories
Canada Centre for Mineral and Energy Technology
Energy, Mines and Resources Canada

MINERALS RESEARCH PROGRAM
MINING RESEARCH LABORATORIES
CANMET REPORT 77-18

Available by mail from:

Printing and Publishing
Supply and Services Canada,
Ottawa, Canada K1A 0S9

CANMET
Energy, Mines and Resources Canada,
555 Booth St.
Ottawa, Canada K1A 0G1

or through your bookseller.

Catalogue No. M38-14/2-1977-1
ISBN 0-660-00989-7

Price: Canada $3.50
Other countries: $4.20

Price subject to change without notice.

En vente par la poste:

Imprimerie et Édition
Approvisionnements et Services Canada,
Ottawa, Canada K1A 0S9

CANMET
Énergie, Mines et Ressources Canada,
555, rue Booth
Ottawa, Canada K1A 0G1

ou chez votre libraire.

N° de catalogue M38-14/2-1977-1
ISBN 0-660-00989-7

Prix: Canada $3.50
Autres Pays: $4.20

Prix sujet à changement sans avis préalable.

THE PIT SLOPE MANUAL

The Pit Slope Manual consists of ten chapters, published separately. Most chapters have supplements, also published separately. The ten chapters are:

1. Summary
2. Structural Geology
3. Mechanical Properties
4. Groundwater
5. Design
6. Mechanical Support
7. Perimeter Blasting
8. Monitoring
9. Waste Embankments
10. Environmental Planning

The chapters and supplements can be obtained from the Publications Distribution Office, CANMET, Energy, Mines and Resources Canada, 555 Booth Street, Ottawa, Ontario, K1A 0G1, Canada.

Reference to this supplement should be quoted as follows:

Cruden, D., Ramsden, J. and Herget, G. Pit Slope Manual, Supplement 2-1 - DISCODAT Program Package; CANMET (Canada Centre for Mineral and Energy Technology, formerly Mines Branch, Energy, Mines and Resources Canada), CANMET REPORT 77-18; 62 p; Dec. 1977.

ACKNOWLEDGEMENTS

G. Herget was responsible for this supplement. Address enquiries to him at: 555 Booth Street, Ottawa, K1A 0G1, Canada.

This supplement was written largely by D. Cruden and J. Ramsden with assistance from R. Clements, R. Weingardt and P. Buttuls. Technical editing and consolidation was carried out by D. Patterson and G. Herget.

Contractors: University of Alberta, Edmonton

The Pit Slope Manual is the result of five years of research and development, cooperatively funded by the Canadian mining industry and the Federal Government of Canada.

The Pit Slope Group consisted of D.F. Coates*, M. Gyenge*, G. Herget, B. Hoare, G. Larocque, D.R. Murray, R. Sage* and M. Service.

*Pit Slope Group Leaders.

CONTENTS

FIGURES

TABLES

SUMMARY

This Supplement is intended to serve as an introduction to the DISCODAT computer programs for the storage, retrieval, analysis and display of structural data. Card decks are supplied separately. Documentation of the computer programs that comprise the system has been carried out to ASCE standards (ASCE Proceedings, 99, SM3, 1973, pp 249-266).

Seven computer programs are documented. They are:

1) CONVRT, a program for building a data file from field observations. The field observations are recorded on a form and all the following programs use the output from CONVRT as input.
2) KEY4, a program for searching and making retrievals from the data file.
3) WNDPLT, a program for producing orientation diagrams on a line printer for weighted, non-directed orientations.
4) WNSTAT, a program to calculate orientation statistics for weighted non-directed data.
5) DIPS, a program to produce histograms of relative and cumulative frequencies of numerical data.
6) HIST2V, a program to produce histograms of discontinuity size and spacing.
7) TRAVMP, a program to produce a map of survey traverses.

System documentation is duplicated for each program to make the description of each program self-contained.

INTRODUCTION

1. This Supplement documents the major part of a system - DISCODAT - for analyzing observations of the characteristics of discontinuities in a rock mass. A second part of the system is concerned with areal variations in discontinuity characteristics and forms the subject of Supplement 2-2. Techniques of describing discontinuities are discussed in the structural geology chapter of the pit slope manual to which the DISCODAT field guide forms an appendix.

2. This Supplement follows, as far as possible, the usual sequence of operations involved in processing field observations into a form immediately useful for design. Observations are first stored (para 14, program CONVRT), then retrieved from storage (para 41, KEY4) and displayed as structural diagrams (para 73, program WNDPLT) or as histograms (para 111, program DIPS). Individual elements of the rock fabric, single joint sets, for instance, can be recognized from the

displays and retrieved using KEY4 from the data bank built up by CONVRT. Input to the data analysis program, WNSTAT, a package of statistical routines documented in para 99, will produce estimates of the mean orientation of the fabric element and descriptions of the variation of observations about that mean orientation.

3. A statistical description of the size and spacing of a fabric element may be necessary for some design purposes. The program HIST2V documented in para 118 provides this. The program TRAVMP (para 135) provides a map of traverses undertaken during mapping of the mine. This display is useful in planning retrievals from the data bank to examine areal variation of a fabric element.

4. It would be unrealistic to suggest that the system cannot be improved or modified to better adapt it to particular computer systems or local requirements. To allow piece-by-piece adaptation, the programs making up the system have been written to be as independent of one another as possible.

5. The main body of the report consists of documentation of computer programs. System information has been duplicated with each program for user convenience and the documentation of each program is self contained.

DATA CAPTURE

6. Processing of the field forms is described after they have been completed by the geologist at the mine site to the stage where the data collected in the field survey are ready for computer input.

7. The procedures described are those developed by a University of Alberta group during the first three years of the Pit Slope Project. The development of data management technology is particularly rapid and procedures are quickly overtaken by advances in hardware. Recent developments in direct entry of data, without card punching, promise economies in the procedures outlined.

8. The geologist completing the field forms should allow himself time at the end of the day to check his work.

9. Information is punched onto computer cards directly from the field forms. To make the completed forms as legible and unambiguous as possible, the geologist should check that:

a. all letters on the forms are capitals;
b. all numbers are right-hand justified in their fields;
c. no decimal points have been inserted;
d. the letter "O" has been written as Ø to distinguish it from the number zero.
e. the number one has been indicated by a vertical line and distinguished from the letter "I" by horizontal bars on the top and bottom of the letter "I";
f. the letter "Z" has been written as Ƶ to distinguish it from the number 2.

10. Obvious redundant information, such as duplication of the identification on both type 1 forms, containing information common to the whole traverse, and type 2 forms which contain information about single observations, is necessary. The duplication of identification allows the field forms and punch cards to be correctly reassembled if for any reason they become out-of-sequence.

11. Much of the information on the type 1 form may be useful in subsequent sorting of the data. Information on domains, formations, pit bench levels and locations falls into this category. Programming generally follows the rule that a blank record indicates lack of information (not zero). Strikes that are due north should be recorded as 360 degrees.

12. Program cards for the IBM 029 card punch for one particular data collection are shown in Table 1. An experienced key punch operator will wish to devise program cards for each data collection. For instance, if information on the

Table 1: PROGRAM CARDS for IBM 029 Card Punch (or equivalent)

```
TRAVERSE DATA CARD (TYPE 1 CARD)
Column 1
↓
/  &&&/  1A  &&&/ &&&&/  &/1AA/  &&&/  &&/  &&&&&/  &&&&&
continued
Column 50
↓
/  &&/  &&/  &&&/  /  &/  &&/  -&&&&&&&
DISCONTINUITY DATA CARD (TYPE 2 CARD)
Column 1
↓
/  &&&/   &&&/ 1A/  &&/  &&/1/1/1/-/-/-&&/-&/-&&/1AA/1
continued
Column 50
↓
/ -&&&&&&&&&&&&&&&&&&&&&  1A  &&&
```

domain has consistently been omitted, the program card will differ from that for a data collection in which the domain had consistently been included.

13. Placing of individual fields from the traverse data card and the discontinuity data card (Table 2) is indicated in Table 3. A listing of typical input data cards forms the input to CONVRT, and is shown in the documentation of this program.

Table 2: Traverse line and discontinuity data sheet

LINE MAPPING SHEET

TRAVERSE LINE

Card Type	Identification	Elevation	Dmn.	Formatn.	Pit Bench Level	Pit Bench Location
0 1 0 1						

Local Grid Northing	Local Grid Easting	Traverse Trend	Traverse Plunge	Traverse Length	Sc.	Nos. Obs.	Reference Direction

Remarks:

DISCONTINUITY DATA

*0 2 Card Type Identification

No.	Discontinuity Distance	Discontinuity Type	Discontinuity Strike	Discontinuity Dip	Size	Spg.	Flng. 1	Flng. 2	Water	Waves I L A	Line Type	Line Pitch	Lthlgy.	Hrdn.

Table 3: Traverse and discontinuity data card

TRAVERSE DATA CARD (TYPE 1 CARD)

```
*0101/3HVO036/    /  /    /    /    /     /     /   /   / / /  /
   identificat. elevation dom formn bch-lvl bch-loc easting northing tr-trnd tr-plun tr-lgth scle nobs ref-dir
```

DISCONTINUITY DATA CARD (TYPE 2 CARD)

```
*02 /   /  /  /  / / / / / /  /  /  /  /  /                              3HVO036
    dist. type azimuth dip size spc fl fl wtr waves line- pitch lithology hdn   identifica-
                                    1  2              type                      tion
```

DATA STORAGE

14. This Section documents a computer program that converts observations of the characteristics of discontinuities collected according to the DISCODAT field guide into line file with the characteristics of each discontinuity occupying one line.

15. While there is some duplication of information about each discontinuity which is common to the whole traverse, the storage format chosen results in simple retrievals of all the information necessary for subsequent data analysis. Retrieval is accomplished by the program KEY4.

16. Input to the program is from computer cards punched to the format set out in Table 3. This format is compatible with input requirements for SAFRAS (Sutterlin, P.G., 1974, U.W.O. SAFRAS system, Geol. Surv, Canada, Paper 74-63, pp 62-67).

DATA CONVERSION WITH CONVRT

17. The program converts field data on discontinuities in a rock mass collected on straight line traverses to a form suitable for subsequent processing.

18. The data is collected as described in the DISCODAT Field Guide (Chapter 2 - Structural Geology, Appendix A). For each traverse, the program merges the information recorded on the type 1 form with that recorded on the type 2 form to produce a set of identical-format data records, one for each observed discontinuity. A flow diagram for program CONVRT is shown in Fig 1(a) and Fig 1(b).

Traverse Line

19. The information from the type 1 input record is placed in the output record unchanged, with the following exceptions: the grid coordinates and elevation of the traverse origin are converted to the basic unit of the survey using the scale read from the parameter card; they are not placed in the output record; and the reference direction is not placed in the output record.

Discontinuity Data

Sequence Number

20. The observations in a traverse are numbered sequentially in the order of input. This number is placed in the output record.

Distance

21. The distance of the point of observation from the traverse origin is converted to the basic unit of the survey using the traverse scale factor. The converted distance is placed in the

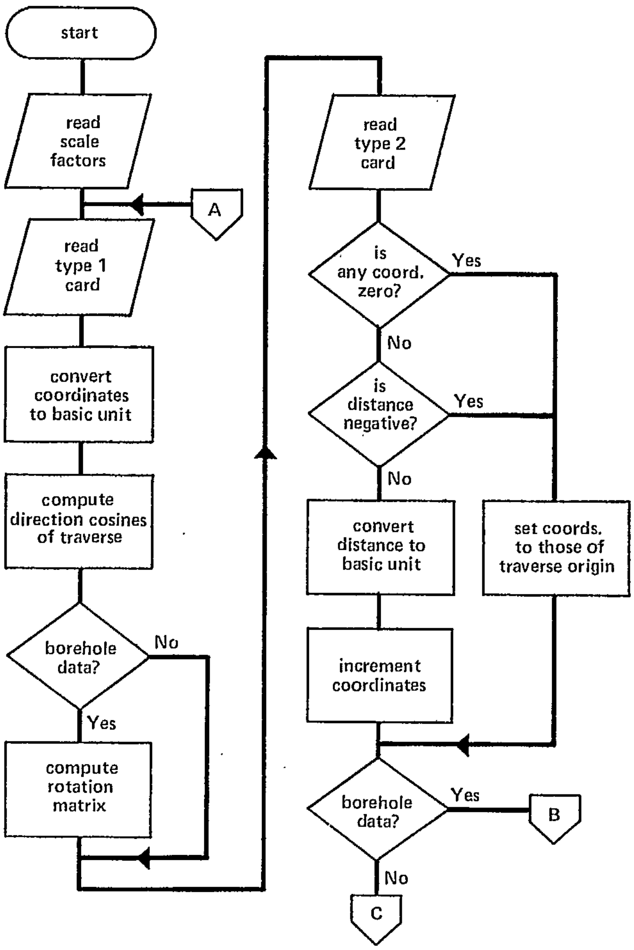

Fig 1(a) - Flow diagram for program CONVRT - Part 1.

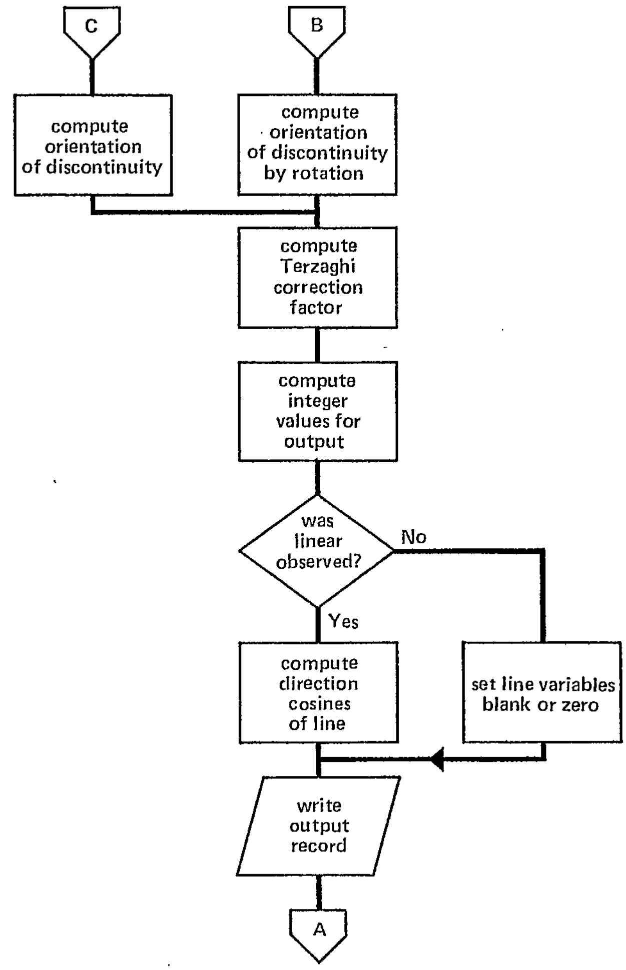

Fig 1(b) - Flow diagram for program CONVRT - Part 2.

output record.

Grid Co-ordinates and Elevation

22. The program computes the grid co-ordinates and elevation of each point of observation as follows. The distance from the traverse origin to the point of observation is resolved into northward, eastward, and upward components, which are added to the co-ordinates and elevations of the traverse origin. The three co-ordinates are rounded to the nearest integer for output.

Orientation

23. The program computes the dip direction of the discontinuity by adding the recorded azimuth to the reference direction. Direction cosines of the downward-directed normal to the discontinuity are computed, multiplied by 10000 and rounded to the nearest integer for output. An upper limit of 9999 is imposed.

Terzaghi Correction Factor

24. The Terzaghi correction factor compensates for bias due to the orientation of the traverse. Its value is the reciprocal of the cosine of the angle between the traverse and the normal to the discontinuity. It is truncated at an angle of 85 degrees, and to provide integer output, it is multiplied by ten and rounded to the nearest integer. The number placed in the output record thus has a range of from 10 to 115. This factor may be used as a weighting factor in subsequent

processing.

Linears

25. Direction cosines of a linear feature are computed from the dip direction and dip of the discontinuity and the pitch of the linear as follows: if the strike vector, S, is rotated counterclockwise about the upward-directed normal to the discontinuity through an angle equal to the pitch of the line, L, it will then coincide with that line. The matrix, RM, that will perform this rotation is obtained by calling the subroutine, ROTMAX. The direction cosines of the line are then found by the matrix multiplication,

$$RM \times S \rightarrow L$$

which is performed by the SSP subroutine, GMPRD, which has been added to the subroutine collection in each of these programs.

Borehole Data

26. The orientations of discontinuities observed in drill cores are computed as follows. Mutually perpendicular axes U, V, W are defined with respect to the core, so that U, V, W are a righthanded set; W is parallel to the core axis and is positive in the direction of drilling; U lies in the vertical plane containing W and is positive upward. If W is vertical, the direction of U is arbitrary.

27. The orientations of the discontinuities in the core are measured with the core oriented so that U, V, W are parallel to the standard reference axes: north, east, down, respectively. To obtain the original orientations of the discontinuities, rotation that will bring the standard reference axes into concidence with the original orientations of U, V, W must be applied to the recorded orientations. To be able to perform this rotation, it is necessary only to know the orientation of the borehole and the direction of drilling.

28. If the trend and plunge of the boreholes are T and P, then the original directions of U, V, W are:

	U	V	W
Trend	T	T+90°	T
Plunge	P-90°	0	P
Direction cosines			
1	cosT sinP	-sinT	cosT cosP
2	sinT sinP	cosT	sinT cosP
3	-cosP	0	sinP

29. If a 3 x 3 identity matrix, whose columns are the reference axes, is pre-multiplied by M, the direction cosine matrix in the above table whose columns are U, V, W, will be M. Thus, pre-multiplication of a column vector by M performs the required rotation. For each discontinuity, the program calls subroutine GMPRD to pre-multiply by M the column vector consisting of the direction consine of the normal to the discontinuity in its recorded orientation. The resulting column vector contains the direction cosines of the normal in its original orientation. From these are computed the original dip direction and dip.

Program Capabilities

30. Not more than 99 observations may be processed as one traverse. There is no limit to the number of traverses that may be processed in one run. All other data restrictions are described in the DISCODAT field guide.

Program Options

31. None.

Program Input and Output

32. The input data are fully described in the DISCODAT (1973) manual, except for the case of borehole data which should be recorded as follows.

33. A reference direction of 999 indicates borehole data. To record borehold data, draw a reference line along the side of the core on that part that was the top in its original orientation. Now orient the core so that its axis is vertical, drilling direction downward, and the reference line is on the north side. Measure the dip-direction and dip of each discontinuity in this

orientation, recording these in the 'azimuth' and 'dip' fields. Record the trend and plunge of the borehole in the 'traverse-trend' and 'traverse-plunge' fields.

34. If the drilling direction was vertically downward, draw the reference line on the original north side of the core, and enter zero in the 'traverse trend' field and 90 in the 'traverse plunge' field. If the direction was vertically upward, draw the reference line on the original south side of the core and enter zero in the 'traverse trend' field and -90 in the 'traverse plunge' field.

Input Data

35. Read from unit 1 - suggested device is a card reader. Traverse data has two types of records: A type 1 record for each traverse; a type 2 record for each observation. The records are ordered:

Type 1 record - 1^{st} traverse
Type 2 records - all observations for 1^{st} traverse
Type 1 record - 2^{nd} traverse
Type 2 records - all observations for 2^{nd} traverse
etc

Type 1 record: One per traverse

Columns	Variable name	Contents*
7-13	AID	traverse identification (A)
15-19	IELEY	elevation
21-22	ADOM	domain (A)
24-26	AFM	formation (A)
28-31	ABLVL	pit bench level (A)
33-35	ABLOC	pit bench location (A)
37-42	IEAST	local grid easting
44-49	INORTH	local grid northing
51-53	ITT	traverse trend
55-57	ITP	traverse plunge
59-62	ALEN	traverse length
64	DSCALE	traverse scale
66-67	NOBS	number of observation
69-71	IREF	reference direction

* Can be correlated with Table 3.

Special conditions: A reference direction of 999 indicates borehole data.

Type 2 record: One per observation

Columns	Variable name	Contents
7-10	IDIST	distance
12-13	ATYPE	type (a)
15-17	IAZ	azimuth
19-21	IDIP	dip
23	ASIZE	size A
25	ASPAC	spacing (A)
27	AFILL1	filling 1(A)
29	AFILL2	filling 2 (A)
31	AWAT	water (A)
33-35	AWAV	waves
37-38	LINTYP	line type (A)
40-42	IPITCH	line pitch
44-46	ALITH	lithology (A)
48-49	AHDN	hardness (A)

Special conditions: If 'distance' is blank, zero or negative, the distance is taken as zero. If any one of the co-ordinates and elevation fields on the type 1 input record is blank or zero, the co-ordinates of all observations on the traverse are set equal to those of the traverse origin.

36. Read from unit 2 - suggested device is a card reader.

Parameter card. One of these is required as the first card in the data deck. It contains scale factor for the grid co-ordinates and elevation.

Parameter card:

One only. This card is placed first in the data deck.

Columns	Contents
1	scale factor for grid co-ordinates
2	scale factor for elevation

These scale factors are determined in the same way as the traverse scale factors, as described in para 24 of the DISCODAT Field Guide (Appendix A, Chapter 2 - Structural Geology).

Output data

37. Written on unit 6 - suggested device is a line printer.

Listing of input. As each input card is read, it is listed for verification.

38. Written on unit 7 - suggested device is a tape disk drive.

Converted data. These converted data are used as input for other DISCODAT programs and for Domain Analysis programs. This is the basic data input file.

The record format is:

Position in record	Contents
18-20	pit bench location
21-23	traverse trend
24-26	traverse plunge
27-30	traverse length
31-32	number of observations
33	traverse scale
34-35	sequence number of observation within traverse
36-39	distance
40-45	easting
46-51	northing
52-56	elevation
57-58	type
59-61	dip direction
62-64	dip
65-69	a
70-74	b
75-79	g
80-82	weight
83	size
84	spacing
85	filling 1
86	filling 2
87	water
89-90	waves
91-92	linetype
93-97	l
98-102	m
103-107	n
108-110	lithology
111-112	hardness

a, b, g are the direction cosines of the downward-directed normal to the discontinuity. l, m, n are the direction cosines of the linear. If no linear was observed, l, m, and n are set to zero.

Variables

Input variables:

Parameter card:

GSCALE, ESCALE
: scale factors for grid coordinates and elevation respectively.

Type 1 card:

AID
: traverse number (A)

IELEV, IEAST, INORTH
: elevation, easting and northing of traverse origin

ADOM
: domain (A)

AFM
: formation (A)

ABLVL
: bench level (A)

ABLOC
: bench location (A)

ITT, ITP
: trend and plunge of traverse

ALEN
: length of traverse (A)

DSCALE
: scale factor for distance of traverse

NOBS
: number of observations on traverse

IREF
: reference direction.

Type 2 card:

IDIST
: distance along traverse

ATYPE
: discontinuity type (A)

IAZ, IDIP
: recorded azimuth and dip

ASIZE
: size code for discontinuity (A)

ASPAC
: spacing of discontinuities (A)

AFILL1, AFILL2

filling codes (A)

AWAT, AWAV

codes for water and waviness (A)

LINTYP

type of line, if any (A)

IPITCH

pitch of line

ALITH, AHDN

codes for lithology and hardness (A).

Output:

LOOP

counter for observations on each traverse

JEAST, JNORTH, JELEV

easting, northing and elevation of point of observation rounded to integers dip direction and dip of discontinuity

ICOSA, ICOSB, ICOSG

direction cosines of normal to discontinuity, times 10^4 and rounded to integers.

IW

Terzaghi correction factor, times 10 and rounded.

LINA, LINB, LING

direction cosines of linear, times 10^4 and rounded to integers.

Internal:

NCD

counter for number of cards read.

NOCOOR

flag for missing coordinates - input

RELEV, REAST, RNORTH

elevation, easting and northing of traverse origin.

T, P

trend and plunge of traverse - radians; trend and plunge of normal to discontinuity in radians; dip direction and dip of discontinuity in degrees.

ST, CT

sine and cosine of trend of traverse.

SP, CP

sine and cosine of plunge of traverse

X, Y, Z

direction cosines of traverse.

RM

rotation matrix to rotate reference axes north, east and down into the axes U, V and W, where W is the borehole direction and V is horizontal.

DIST

distance from traverse origin

SNORTH, SEAST, SELEV

northing, easting and elevation of point of observation

COSA, COSB, COSG

direction cosines of normal to discontinuity

VR

vector containing direction cosines of normal to discontinuity in recorded orientation; vector containing the direction cosines of the strike vector.

VT

vector containing direction cosines of normal to discontinuity - true orientation; vector containing direction cosines of linear.

CALPHA

cosine of angle between normal to discontinuity and traverse or borehole.

W

Terzaghi correction factor for discontinuity

TAX, PAX, ANG

trend and plunge of axis of rotation and angle of rotation, in radians, to rotate strike vector through pitch angle into coincidence with linear.

RM2

rotation matrix to rotate the strike vector through the pitch angle into coincidence with the linear.

Equivalences:

COSA, COSB and COSG occupy the same storage locations as the vector VT.

Subroutines and Function Subprograms

39. Function IDC(R): takes the direction cosine R, multiplies it by 10^4 and rounds it to the nearest integer. A maximum absolute value of 9999 is imposed. The result is returned as the function value.

Subroutine GMPRD: from the SSP subroutine package (IBM). Multiplies two matrices.

Calling sequence:

CALL GMPRD (A,B,C,I,J,K)

The J*K matrix B is premultiplied by the I*J matrix A and the result placed in the I*K matrix C.

Subroutine DCAN: included in subroutine library accompanying these programs. Calling sequence

```
CALL DCAN (A,B,G,I,T,P,J,K,N)
```

The vectors T and P of angles are computed from the vectors, A,B, and G of direction cosines. I is the spacing of direction cosines - A,B and G, and J the spacing of angles in T and P. For example, if J=2, output angles are placed in T(1), T(3), T(5), etc. If |K|=1, trends and plunges are computed. If K>0, angles are returned as radians; if K<0, angles are returned as degrees. N is the number of directions given.

Subroutine ROTMAT: Accompanying subroutine library. Computes a matrix for rotation through a given angle about a given axis.

Calling sequence:

```
CALL ROTMAT (A,B,C,R,)
```

A,B, and C are the trend and plunge of the axis of rotation and the angle of rotation in radians, R is the rotation matrix.

Data Structures

40. The input data file consists of unblocked 80-byte records. The program does not block output records. Output data records are 112 bytes long. One output data record is written for each type 2 input record.

Storage Requirements

Code	bytes:	words:
Main program	1045	549
	6144 (buffers)	3172
Function IDC	22	18
subroutine GMPRD	103	67
Subroutine DCAN	176	126
Subroutine ROTMAT	131	89
Total	7531	1021

DATA RETRIEVAL

41. This chapter documents a program, KEY4, that selectively retrieves records from identical record files. It is used to make conditional retrievals from the line files generated by the data storage program CONVRT.

42. The program plays a central role in the analysis of discontinuity data. It may first be required to retrieve the records of all discontinuities of a particular type for the whole mine. After display of these, displays of elements of the subfabric may be required for the characterization of a single joint set, say, among all the joints observed. Displays of all the members of the set from a particular part of the mine can be obtained if necessary. The program has therefore been written with the ability to operate on the subsets of the original data that it has retrieved from the data file. This leads to substantial economies in sequences of related retrievals and the user should consider storing the results of any particular retrieval for further use.

43. Output will generally be used as input to one of the data display programs - to the program, WNDPLT, if data to be displayed are orientations of discontinuities; to the program, HIST2V, if the data are sizes or spacings of discontinuities; or to the program, TRAVMP, if the output required is a traverse map.

SEARCH AND RETRIEVE WITH KEY4

44. This program searches a data file containing identical-format records for those records satisfying a given combination of conditions. It either copies those records into a separate file, or simply outputs the numbers of those records.

Solution

45. The program performs a series of discrete operations, each being initiated by a parameter card. These operations are of two kinds: those that involve a condition, SCAN - operations, and those that do not. If the operation is of the first kind, the program examines the data file and stores the numbers of those data records that satisfy the given condition, involving one field of the data record. Depending on the parameters, the program will examine either the entire data file or a subset of the data file identified in a previous operation.

46. Operations that do not involve a condition require some manipulation of the results of previous operation to produce a new set of record numbers. The data file is not examined.

47. On completion of any operation, it is possible to (1) save permanently the set of record numbers produced by the operation, and (2) actually retrieve the data records whose numbers

belong to that set and place them in a separate file. It is also possible to input sets of record numbers saved from previous runs. A flow diagram is given in Fig 2(a) to Fig 2(g).

Program Capabilities

48. The records of the data file must all have the same length and format.

49. The program is presently designed for a data file containing 112-byte records. The program CONVRT creates records of this type. The present limit on the size of a subset of the data file that can be defined is 5000 records. These limits may be changed as described below.

50. If the program is to be used on a file whose records are not 112 bytes long, the sizes of the vectors LINE and BLINE in subroutine NRETR must be changed, as well as format statement number 1 in the same subroutine.

51. To change the maximum size of a subset of a data file that can be handled, the dimensions of the variables KEY1, KEY2 and KEY3 in all program segments except POSN, RKEY, READ and WRITE must be changed to the new maximum plus 18. The dimension of KEY in subroutines READ and WRITE must be similarly changed.

52. If the sizes of these arrays are changed, the maximum length of unformatted records temporarily stored on the device assigned to I/O unit 2 will also change, and will be equal to the length of the array KEY3 plus one full word (plus words used by the system).

Program Options

53. There are 6 operation codes: SCAN, READ, COMM, PLUS, LESS, STOP.

SCAN

54. The scan - operation involves examining the data file or some subset of it and storing the numbers of those data records satisfying a given condition. label2 identifies the subset of the data to be examined. If label2 is blank, the entire data file is examined. field name is one of the field names defined on the field-definition cards. relational operator is one of the following:

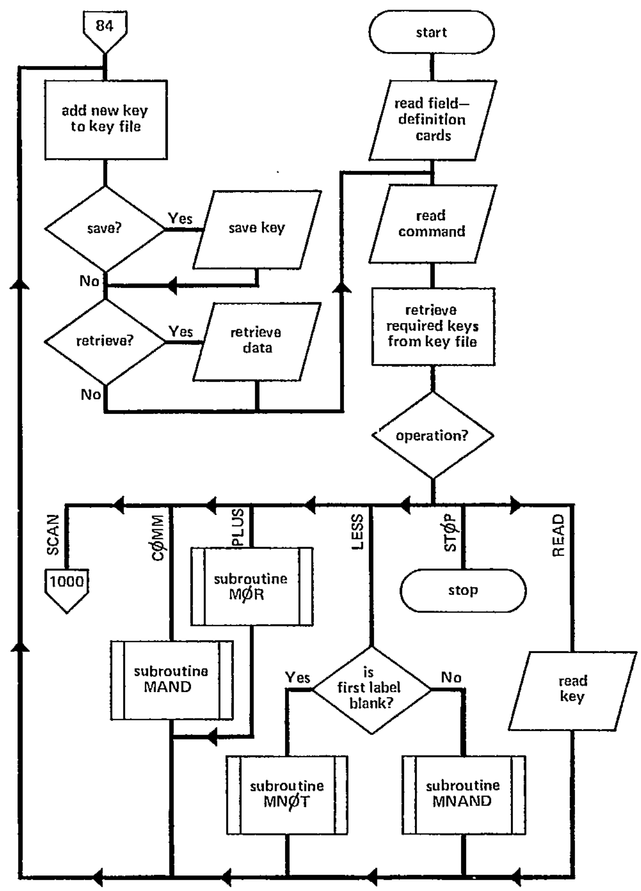

Fig 2(a) - Flow diagram for program KEY4 - Part 1.

EQ	(equal to)	NE	(not equal to)
LT	(less than)	NL	(not less than)
GT	(greater than)	NG	(not greater than)
GE	(greater than or equal to)	LE	(less than or equal to)
IR	(in range)	OR	(outside range).

55. If field name was defined with an A format code, a character string is to be placed in parameter field 6, and relational operators other than EQ and NE should not be used. If field name was defined with an I format code, then a single integer is to be placed in parameter field 7 for all relational operators other than IR and OR, while for these relational operators the lower and upper limits of the range (respectively) are to be placed in parameter fields 7 and 8. The range is

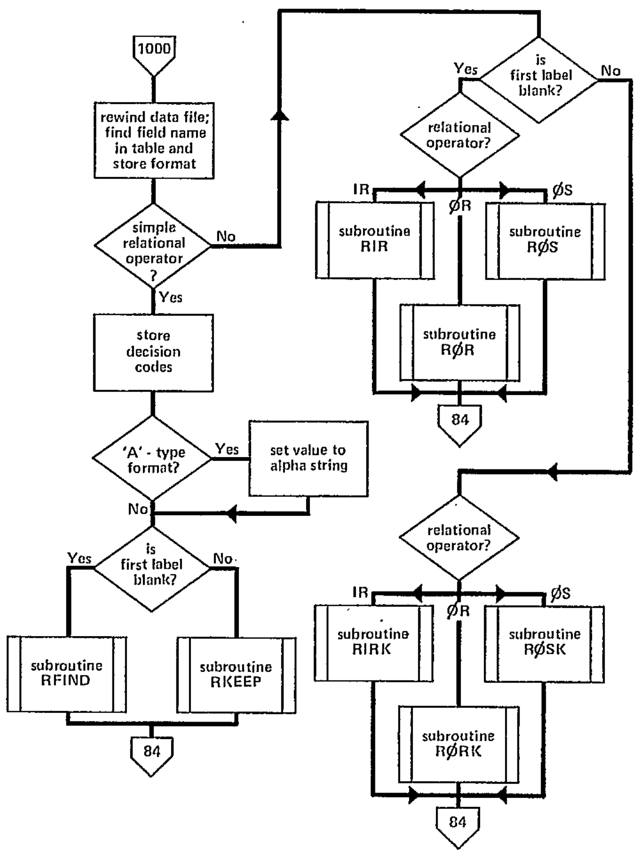

Fig 2(b) - Flow diagram for program KEY4 - Part 2.

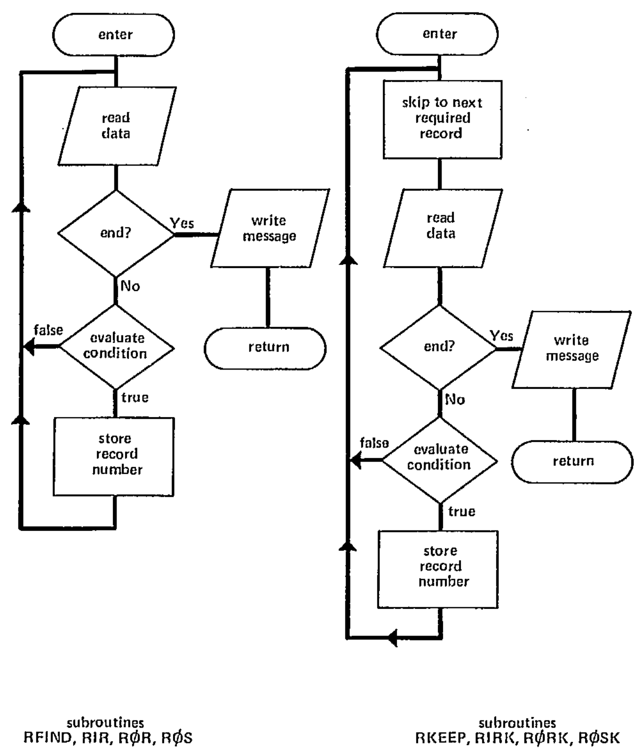

Fig 2(c) - Flow diagram for program KEY4 - Part 3.

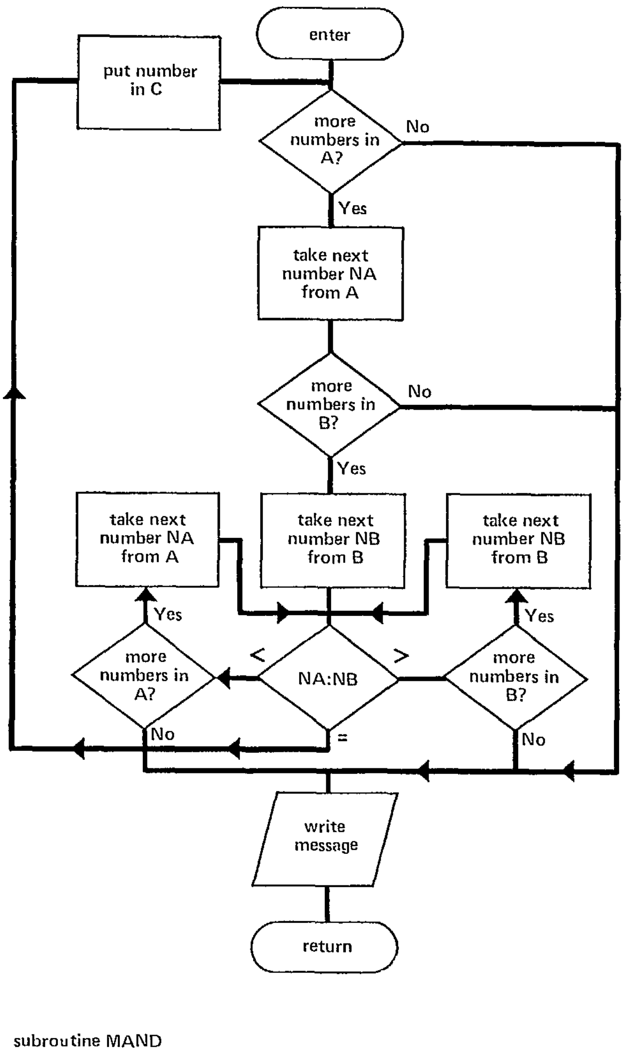

Fig 2(d) - Flow diagram for program KEY4 - Part 4.

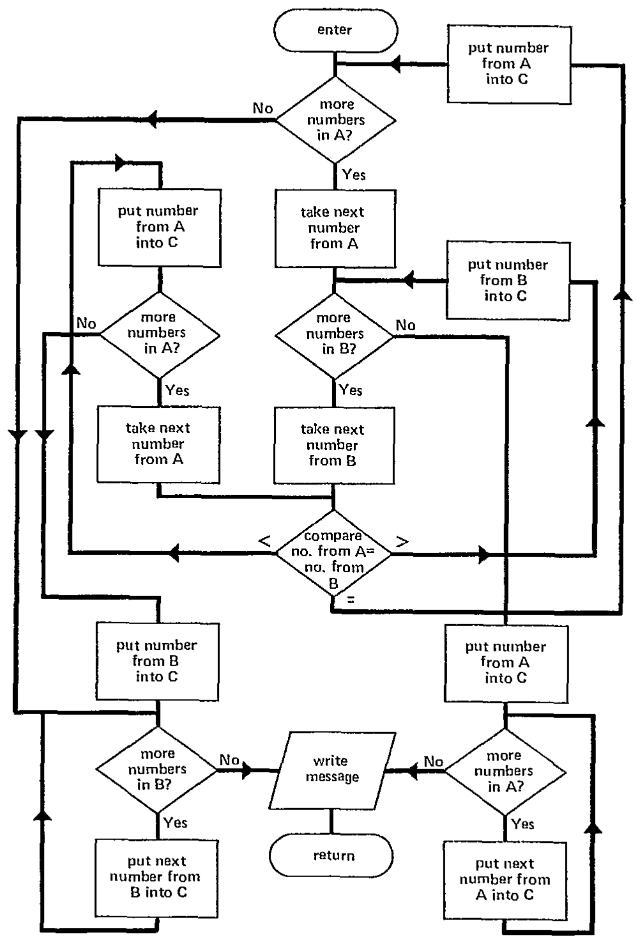

Fig 2(e) - Flow diagram for program KEY4 - Part 5.

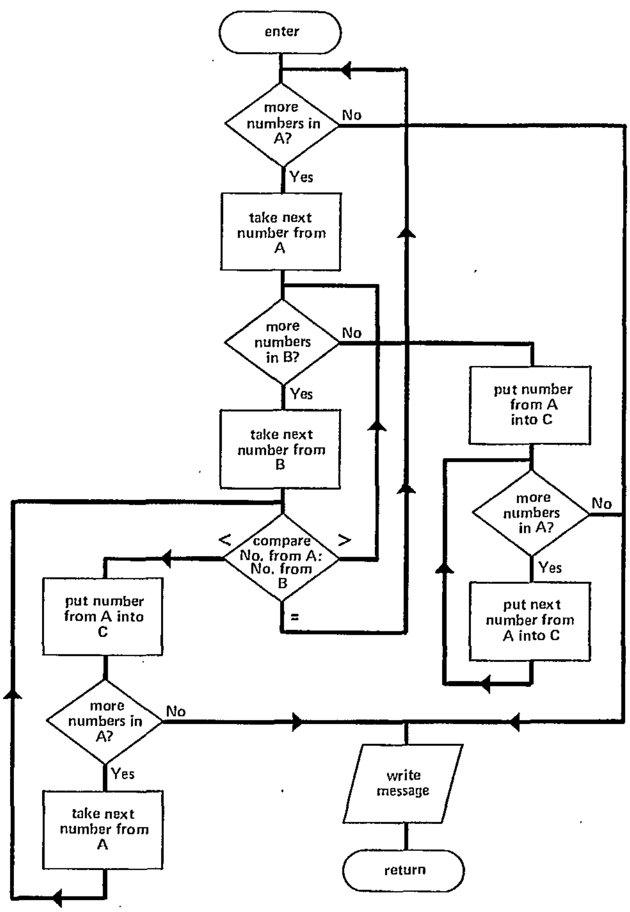

Fig 2(f) - Flow diagram for program KEY4 - Part 6.

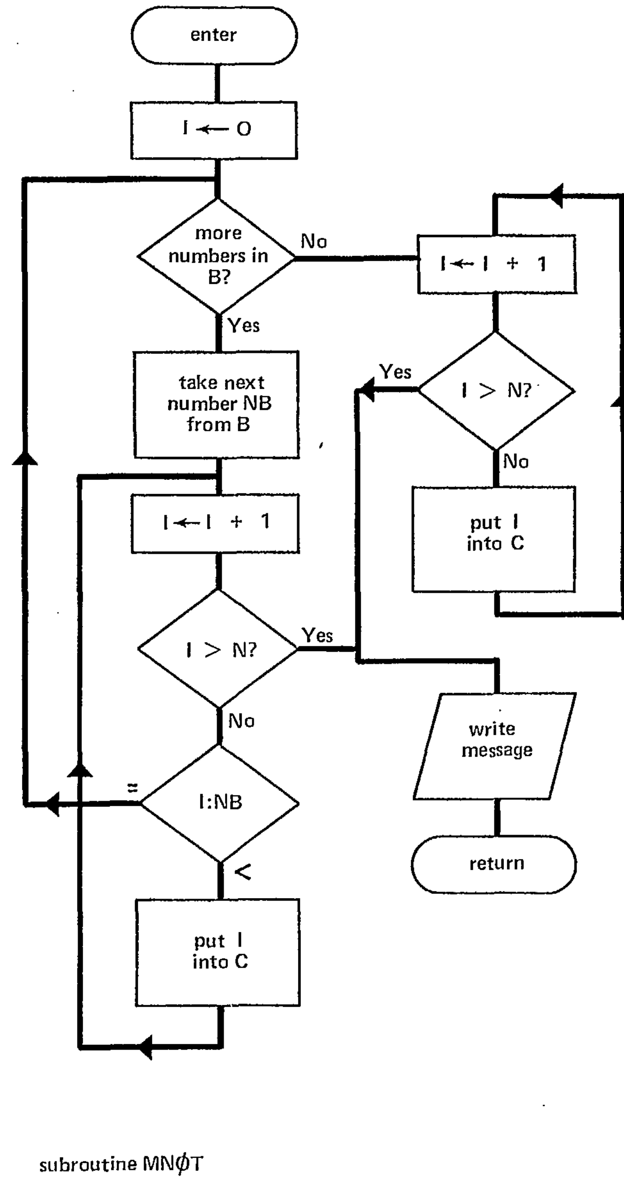

Fig 2(g) - Flow diagram for program KEY4 - Part 7.

considered to include the lower limit and exclude the upper limit.

READ

56. This operation code causes the program to read from I/O unit 4 a set of record numbers (saved from a previous run).

COMM

57. This code generates a set of record numbers consisting of those that belong to both of the sets identified by label 1 and label2.

PLUS

58. This code generates a set of record numbers consisting of those that belong to either of the two sets identified by label 1 and label2.

LESS

59. This code generates a set of record numbers consisting of those that belong to the set identified by label 1 but do not belong to the set identified by label2.

STOP

60. This code causes the program to stop.

Program Input and Output

61. Read from unit 1 - tape or disk.

Data file. This is the file created on unit 7 of CONVRT. Under Program Capabilities there is a description of how to change the format to accept another format of data. The Field Definition cards on unit 3 (below) can be used to redefine the input data fields.
The default is set to handle CONVERT output, the information selected is controlled by the parameters from unit 5.

62. Read from unit 3 - card reader.
Field definition. This is the first set of cards in the data deck. The first card contains a two digit number which is right justified giving the number of field definition cards that follow. The maximum is 99.

63. The field definition cards are of the form:

Card column	Contents
1-8	Field name of a field to be selected. See the output record description under CONVRT.
9-34	blank
35-43	A FORTRAN format defining the field's position in the record, its length and type (only I or A permitted).

Example:

(51X,I5)

field is integer, and 5 characters long.

skip 51 characters.

64. Read from unit 4 - card reader.
Input set record numbers, keys. These are used only when a READ parameter is used. If these cards appear, they follow a READ parameter card read from unit 5. These cards are the output on unit 7, and would be saved from a previous run. If the cards are not in order, an error message will be generated.

65. Read from unit 5 - card reader.
Parameter cards. These follow the cards read from unit 3. These cards contain any of the commands described under Program Options. The format of all the parameter cards is:

field number	variable name	contents	columns
1	LBL1	label 1	1-8
2	OPER	operation	10-13
3	LBL2	label 2	15-22
4	DNAME	field name	24-31
5	REL	relational operator	33-34
6	ALPH	alpha value	36-39
7	NUMA	first numerical value	41-48(right justified)
8	NUMB	second numeric	

		value	50-57(right justified)
9	LBL	label 3	59-66
10	KSAVE	'save'character	68
11	KRETR	'retrieve'	

Note: If I/O units 3, 4, and 5 are all assigned to the same input device, the order of input must be as follows: field-definition cards first, followed by parameter cards. One and only one previously saved set of record numbers must immediately follow each parameter card that specifies a READ - operation.

66. Written on unit 6 - line printer.
The program prints a message as each operation is completed.

67. Written on unit 7 - suggested device is a card punch.
Saved set of record numbers. These records can be used as input on a later run. See unit 4 and READ parameter.

68. Each set of record numbers saved is written out in 80-byte lines, numbered sequentially in columns 78-80. The first two lines contain the label, the number of records in the data file, and the number of record numbers in the set. Each subsequent line contains record numbers (keys) in 18 adjacent 4-column fields in columns 1 to 72. Unused fields on the last line contain zeros.

69. Written on unit 8 - suggested device is a tape.
Output data file. Depending on the parameter cards, the output file will contain an edited collection of input records. This permits a subset of records to be used as input to the subsequent programs of the series.
For each subset of the data file retrieved, the program writes one record containing the associated label followed by the retrieved data records, followed by one blank record.

Variables and Subroutines

Input variables:

LENTAB
the number of lines in the table

TABLE
names of data fields and format codes

LBL1
label of first subset

ØPER
operation code

LBL2
label of second subset

DNAME
name of data field to be examined

REL
relational operator

ALPH
alphameric value

NUMA
first numeric value

NUMB
second numeric value

LBL3
label to be associated with new subset.

KSAVE
'save' option control character

KRETR
'retrieve' option control character.

Output variables:

LBL
label associated with retrieved subset

LINE
data record

BLINE
blank line.

Subroutines:

POSN
position a file at a given record. Calling sequence:

CALL POSN(IOU,NCUREC,IPOSN)
IOU= input/output unit of file
NCUREC= current record
IPOSN= desired record

ROS
not used

RIR
Searches entire data file for values inside the given range. Calling sequence: CALL RIR

ROSK
not used

MNOT
not used

NRETR

retrieves the subset of the data file given by the key just obtained. Calling sequence:

```
CALL NRETR(IOU,LBL)
     IOU= input/output unit
     LBL= label of subset of data file
```

NSAVE

writes out the key just obtained on the printer and punch units. Calling sequence:

```
CALL NSAVE(LBL)
     LBL= label of subset of data file
```

RIRK

Searches a subset of the data file for values inside the given range. Calling sequence:

```
CALL RIRK
```

MAND

Puts into KEY3 these numbers that appear in both KEY1 and KEY2. Calling sequence:

```
CALL MAND
     KEY1= scratch matrix
     KEY2= scratch matrix
     KEY3=
```

ROR

searches the entire data file for values outside the given range. Calling sequence:

```
CALL ROR
```

RKEY

positions the input file according to the key. Successive calls to next (alternate entry point of this routine) position the file before the next records specified in the key. Calling sequence:

```
CALL RKEY(KEY,LENKEY,IDATA)
     KEY=
     LENKEY= the key length
     IDATA= input data I/O unit
```

RFIND

searches the entire data file for those records satisfying the given condition. Calling sequence:

```
CALL RFIND
```

RKEEP

searches a subset of the data file for those records satisfying the given condition. Calling sequence:

```
CALL RKEEP
```

RORK

searches a subset of the data file for values outside the given range. Calling sequence:

```
CALL RORK
```

MOR

puts into KEY3 those numbers that appear in either KEY1 or KEY2. Calling sequence:

```
CALL MOR
```

MNAND

puts into KEY3 those numbers that appear in KEY1 but not in KEY2. Calling sequence:

```
CALL MNAND
```

READ

reads a key. Used to read input from unit 4. Calling sequence:

```
CALL READ(IOU,TITLE,LENFIL,LENKEY,KEY)
     KEY = KEY3
```

WRITE

writes a key. used to write the keys on the line printer. These keys are as read by the subroutine READ used in KEY4 and WNDPLT. Calling sequence:

```
CALL WRITE(IOU,TITLE,LENFIL,LENKEY,KEY).
```

Data Structures

70. Structure of the input file is described above. The program does no blocking or deblocking. Present record length for the data file is 112 bytes.

71. Retrieved data file consists of title records, and one blank record as a delimiter. This sequence is repeated if more than one subset of data is retrieved. No end-of-file marks are written in the retrieved data file.

72. Sets of record numbers are output as 80-byte records. Each set consists of two title records followed by records containing the numbers. All the records in each set including the two title records are numbered sequentially in columns 78-80.

Storage Requirements

Code	bytes:	words:
KEY4	2757	1519
	14307 (buffers)	6343
PØSN	31	25
RØS	3	3
RIR	65	53

RØSK	3	3
MNØT	3	3
NRETR	127	87
NSAVE	50	40
RIRK	111	73
MAND	53	43
RØR	66	54
RKEY	52	42
RFIND	110	72
RKEEP	134	92
RØRK	112	74
MØR	122	82
MNAND	60	48
READ	144	100
WRITE	100	64
common blocks: ØNE	35417	15119
Total storage required:	40828	23883

Sample Run

Table of data names and formats:

37

YEAR	(001X,I1)
GEOL	(002X,A2)
TRAVERSE	(004X,I4)
DOMAIN	(008X,A2)
FORMATIO	(010X,A3)
LEVEL	(013X,I4)
LOCATION	(017X,A3)
TREND	(020X,I3)
PLUNGE	(023X,I3)
LENGTH	(026X,I4)
NOBS	(030X,I2)
SCALE	(032X,I1)
SEQNO	(033X,I2)
DISTANCE	(035X,I4)
EASTING	(039X,I6)
NORTHING	(045X,I6)
ELEVATIO	(051X,I5)
TYPE	(056X,A2)
DIPDIR	(058X,I3)
DIP	(061X,I3)
COSA	(064X,I5)
COSB	(069X,I5)
COSG	(074X,I5)
WEIGHT	(079X,I3)
SIZE	(082X,A1)
SPACING	(083X,A1)
FILLING1	(084X,A1)
FILLING2	(085X,A1)
WATER	(086X,A1)
WAVES	(087X,I3)
LINETYPE	(090X,A2)
LINA	(082X,I5)
LINB	(097X,I5)
LING	(102X,I5)
LITHOLOG	(107X,A3)
HARDNESS	(110X,A2)
MINE.LEV	(113,I2)
↑	↑
1	35

Search and retrieval instructions:

```
SCAN           LINETYPE NE                 LINEARS  XX
SCAN LINEARS   DIP      IR      80     90  SET1     XX
STOP
↑    ↑         ↑        ↑       ↑      ↑   ↑        ↑
10   15        24       33      38     56  59       68
```

Input data file:

```
TXFILEVJN
D2DM0002010R45920    222 -21001680 1   2478381496430259422JN 82 60-1205-8576 5000 15J   3171FD-9067 3001 2962SDSR5
D2DMD002010R45920    222 -21001680 2   2878381196429959422JN352 87-9889 1390  523 16J   3164FD 1480 8944 4220SDSR5
D2DM0002010R45920    222 -21001680 4   7478378196426559424JN 95 80  858-9811 1736 17J   3163FD -151 1730 9848SDSR5
D2DM0002010R45920    222 -21001680 9 1237837489542295942 5JN105 85 2578-9623  872 22J   3           0    0    0SDSR5
D2DM0002010R45920    222 -2100168010 1557837269642055942 6JN102 85 2071-9744  872 20J   3172FD 3175 1512 9361SDSR5
D2DM0002010R45920    222 -2100168011 1807837109641865942 7JN127 80 5927-7865 1736115J F 3174FD-7862-5181 3368SDSR5
D2DM0002010R45920    222 -2100168023 3877835719640335943 5JN102 85 2071-9744  872 20K   3           0    0    0SDSR5
D2DM0002010R45920    222 -2100168025 4167835529640115943 6JN 91 69  163-9334 3584 17K   3170FD 9955  486  814SDSR5
D2DM0002010R45920    222 -210D168026 4177835519640105943 6JN104 82 2396-9609 1392 22J F 3166FD-9695-2292  863SDSR5
D2DM0002010R45920    222 -2100168027 4347835409639985943 6JN102 74 1999-9403 2756 21J F 3162FD  453 2899 9560SDSR5
................     ... . ........  .................... .. ..  ........ .... .. .          .    .   ......
D2DM0003010R45920    227  0 21124024   0783140963630595 38JN 25 90-9063-4226    0 11 I           0    0    0SDSR5
```

Printed output:

```
           SCAN          LINETYPE NE               0        0 LINEARS  XX

END OF DATA

   38 LINES READ        23 LINE NUMBERS IN KEY
NEW KEY ADDED TO FILE
LINEARS  HAS BEEN SAVED
LINEARS  HAS BEEN RETRIEVED


           SCAN LINEARS  DIP       IR            80         90 SET1      XX

END OF DATA

OLD KEY CONTAINED   23 LINE NUMBERS     NEW KEY CONTAINS  12 LINE NUMBERS
NEW KEY ADDED TO FILE
SET1     HAS BEEN SAVED
SET1     HAS BEEN RETRIEVED

           STOP                               0         0
```

Output "KEYS":

```
LINEARS                                                                1
                                                   38      23          2
   1   2   3   5   6   8   9  10  12  13  14  15  16  17  18  19  20  22   3
  23  28  29  33  34   0   0   0   0   0   0   0   0   0   0   0   0   0   4
SET1                                                                   1
                                                   38      12          2
   2   3   5   6   9  15  16  17  20  23  28  33   0   0   0   0   0   0   3
```

Output retrieved data file:

```
TLINEARS
D2DM0002010R45920   222 -21001680 1   247838149643025942 2JN 82 60-1205-8576 5000 15J    3171FD-9067 3001 2962SDSR5
D2DM0002010R45920   222 -21001680 2   2878361196429959422JN352 87-9889 1390  523 16J    3164FD 1480 8944 4220SDSR5
D2DM0002010R45920   222 -21001680 4   7478378196426559424JN 95 80  858-9811 1736 17J    3163FD -151 1730 9845SDSR5
D2DM0002010R45920   222 -2100168010 1557837269642055942 6JN102 85 2071-9744  872 20J    3172FD 3175 1512 9361SDSR5
D2DM0002010R45920   222 -2100168011 1807837109641865942 7JN127 80 5927-7865 1736115J F 3174FD-7862-5181 3368SDSR5
D2DM0002010R45920   222 -2100168025 4167835529640115943 6JN 91 69  163-9334 3584 17K    3170FD 9955  486  814SDSR5
D2DM0002010R45920   222 -2100168026 4177835519640105943 6JN104 82 2396-9609 1392 22J f 3166FD-9695-229   863SDSR5
D2DM0002010R45920   222 -2100168027 4347835409639985943 6JN102 74 1999-9403 2756 21J F 3162FD  453 2899 9560SDSR5
D2DM0002010R45920   222 -2100168032 5977834319638775944 2JN 79 54-1544-7942 5878 16J MB4178FD 9871-1501  564SDSR5
D2DM0002010R45920   222 -2100168033 5857834399638865944 1JN107 76 2837-9279 2419 25K    3166FD 9335 3250 1518SDSR5
.................   ... . ........  .................... .. ........ .... .. .    ..... ....  ... .........
D2DM0003010R45920   227  0 21124019   0783140963630595 38JN 42 90-7431-6691    0 10 I   15FD-6399 7107   2924SDSR5

TSET1
D2DM0002010R45920   222 -21001680 2   2878381196429959422JN352 87-9889 1390  523 16J    3164FD 1480 8944 4220SDSR5
D2DM0002010R45920   222 -21001680 4   7478378196426559424JN 95 80  858-9811 1736 17J    3163FD -151 1730 9848SDSR5
D2DM0002010R45920   222 -210016B010 1557837269642055942 6JN102 85 2071-9744  872 20J    3172FD 3175 1512 9361SDSR5
D2DM0002010R45920   222 -2100168011 1807837109641865942 7JN127 80 5927-7865 1736115J F 3174FD-7862-5181 3368SDSR5
D2DM0002010R45920   222 -2100168026 4177835519640105943 6JN104 82 2396-9609 1392 22J F 3166FD-9695-2292  863SDSR5
D2DM0002010R45920   222 -2100168036 6607833899638305944 4JN 94 80  687-9824 1736 17K   4167FD-9976 -698     0SDSR5
D2DM0002010R45920   222 -2100168037 6697833839638235944 4JN164 85 9576-2746  872 19K   3160FD-2860-9425 1730SDSR5
D2DM0002010R45920   222 -2100168038 7147833539637905944 6JN101 87 1905-9803  523 20K   3171FD-9788-1856  870SDSR5
D2DM0002010R45920   222 -2100168049 8677832509636765945 1JN150 80 8529-4924 1736 32K   3159FD-3267-2439 9131SDSR5
D2DM0002010R45920   222 -2100168054 9307832089636295945 3JN157 85 9170-3892  872 24J   3175FD-3982-8803 2578SDSR5
D2DM0002010R45920   222 -2100168060 9827831739635915945 5JN207 85 8876 4523  872 10J   3170FD 4000-8508 3407SDSR5
D2DM0003010R45920   227  - 21124018   0783140963630595 38JN 91 83  173-9924 1219 14 I   174FD 9652  484 2569SDSR5
```

DATA DISPLAY

73. This part documents a program, WNDPLT, that produces an orientation diagram on a 0.2 m diameter Schmidt equal-area projection of the sphere with a dense array of counting locations. Input are weighted non-directed orientations, output is on a conventional line-printer.

74. The orientation diagram is the fundamental display in the analysis of orientation data. It is the routine first step in the analysis of a sub-fabric into elements. In a typical analysis, all joints observed in the mine or in a particular domain of the mine are displayed on an orientation diagram. The geologist then makes a trial division of the joint sub-fabric into elements and single joint sets, using personal judgment of the display. The limits placed on the strike and dip of the elements of the subfabric can be used to make selective retrievals of the elements using program KEY4 from the subset of the data file generated by the initial retrieval request.

75. Input to the program then would usually be from program KEY4. The next step in the data processing after this display is the statistical description of the data in programs WNSTAT and DIPS provided in para 96.

DISCONTINUITY ORIENTATION WITH WNDPLT

76. The program produces on the line printer, equal-area lower-hemisphere projections of point density for weighted, non-directed orientations. Each character position within the 20 cm projection serves as an independent counting location and contains a character indicating to the nearest one per cent the fraction of the sample lying within the counting circle centred at that location.

77. To map the density of points on a sphere, a density estimate is required at each of a number of pre-determined counting locations. For unweighted points, the density estimate at a particular location would be obtained by counting the number of points falling inside a pre-determined area or counting circle surrounding the location. The density estimate is usually obtained by expressing this count as a percentage of the total number of points. A density estimate for weighted points is obtained by summing the weights of the points falling inside the area and expressing this as a percentage of the total of the weights of all the data points.

Solution

78. A flow diagram for program WNDPLT is given in Fig 3(a) to Fig 3(c). This program employs a print matrix consisting of 3713 print positions: 47 lines with 79 print positions on each line. Printed six lines to the inch, it is approximately eight inches square. The equal area projection is a circle with a 10 centimeter radius centered on the print matrix. Every print position whose

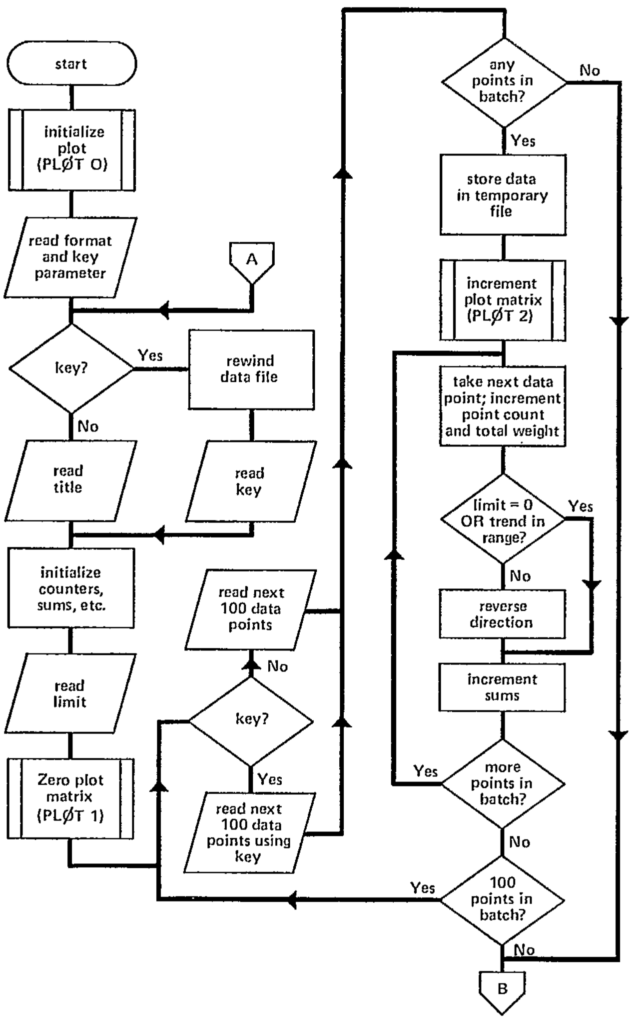

Fig 3(a) - Flow diagram for program WNDPLT - Part 1.

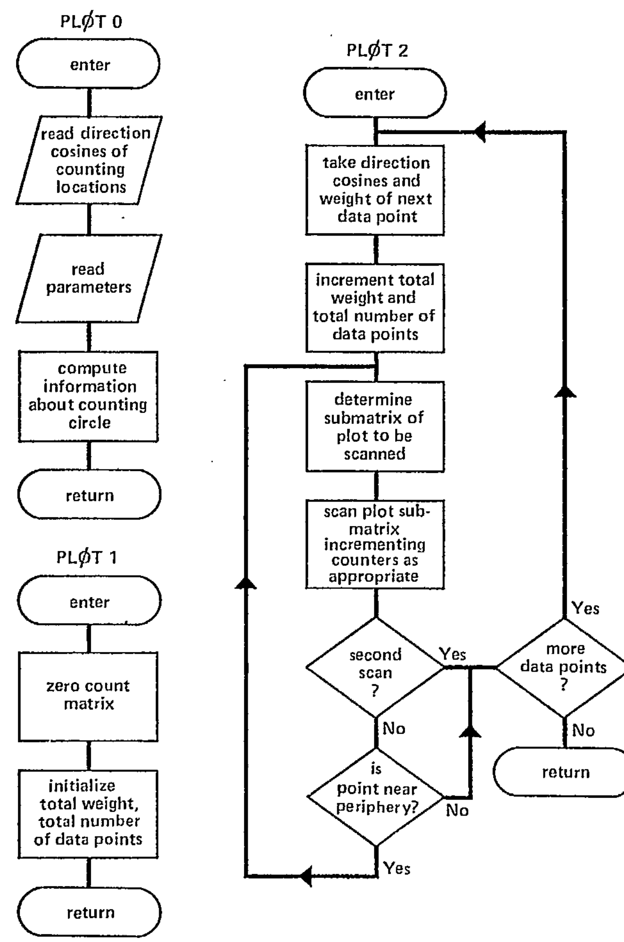

Fig 3(b) - Flow diagram for program WNDPLT - Part 2.

centre lies within the circle represents a counting location of which there are 2933. The counting locations are defined by direction cosines, read in by the program at the start of each run.

79. With such a large number of counting locations it is not feasible to compare every counting location with every data point in the counting process. The number of comparisons is greatly reduced by means of the following relatively simple procedure. For a particular data point, those counting locations that are to be incremented lie within a circle on the sphere whose projection approximates an ellipse that has its greatest size when the data point lies in the horizontal plane. Before beginning the counting procedure, the program computes the maximum dimension, M, for the ellipse in this case. For each data point, the program computes XP, YP, the X-Y co-ordinates on the print matrix of the projection of the data point, then determines the

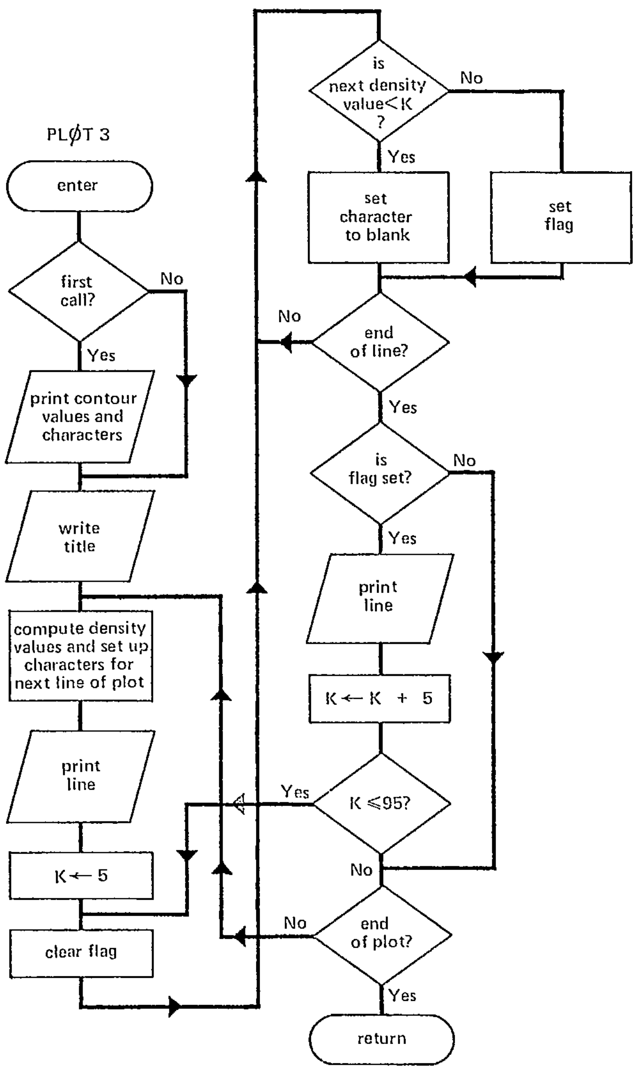

Fig 3(c) - Flow diagram for program WNDPLT - Part 3.

submatrix of the print matrix that lies inside the square bounded by

$$X = XP + M, X = XP - M, Y = YP + M, Y = YP - M.$$

This square wholly contains the projection of the counting circle centered on the data point. Each counting location represented in this submatrix is then tested by computing its angular distance from the data point and comparing this with the radius of the counting circle. This process is repeated in a slightly modified form if the data point is close enough to the horizontal to affect counting locations on the diametrically opposite part of the projection.

80. After the counting procedure is completed, the program constructs the plot, line by line, by expressing each count as a percentage of the total weight and then placing the appropriate character in the corresponding print position. A contour interval and a string of characters are supplied to the program on parameter cards. As an enhancement to the visual impact of the plot, characters representing higher densities are overprinted, the number of which increases with the density value.

Program Capabilities

81. There is no limit to the number of data points that can be plotted in one diagram. Given directions may be on either the upper or the lower hemisphere; all directions are converted to the lower hemisphere.

Program Options

82. The fifth parameter card specifies the data input mode. If the NOKEY - option is specified, every line of the input data file is used. If the KEY - option is specified, a set of line numbers is read from I/O unit 4, and those line numbers are used to determine which lines of the input data file are to be used.

Program Input and Output

83. Read from unit 1 - tape or disk. Orientations to be plotted.

84. The data file may be read with or without the use of a key as created by the program KEY4. If a key is used, each record of the data file contains the direction cosines and weight of one data point.

85. If no key is used, the data file must consist of a title record, followed by data records as above, followed by one blank record. This sequence may be repeated any number of times.

86. Read from unit 3 - tape or disk. Direction consines. A file containing the direction cosines is supplied with these programs. Its name is DIRECTIONCOSINE.

87. Read from unit 5 - card read. Parameters.

These parameters are optional, except for a blank card to indicate an end of file if no parameters are specified.

(1) One card containing the per cent size of the counting circle in columns 1 to 4 as a right-justified integer or with a decimal point.

(2) One card containing the contour interval, same format as (1).

(3) One card containing the plot characters in columns 2 to 41. The character for the first contour interval should be blank, since this character is placed in all positions in the print matrix that are outside the projection.

(4) One card containing the format of the data in columns 1 to 60. The list of variables read is as follows:

Variable	Type
Dir.cos.1	Real*4
Dir.cos.2	Real*4
Dir.cos.3	Real*4
Weight	Real*4
Unused variable	Integer*4

Thus the format would look like (F4.n,F4.n, F4.n,F4.n,I4) where n is the number of decimal places. If n is the same in all cases, say 2, (4F4.2,I4) could be used.

(5) One card containing the word 'KEY$\not{b}$' in columns 1 to 4 if a key is to be used, otherwise any other character sequence in columns 1 to 4 ($\not{b}$ represents a blank).

88. Read from unit 4 - card reader.

KEYS. This is optional. It is only specified if card 5 of the input on unit 5 is used. If the keys are to be used, they are the output from KEY4 and should be placed in the data immediately after the KEY card of unit 5 (card 5).

89. Inputs in order of reading are summarized below.

I/O Unit	Input	Format
3	Direction cosines of counting locations	
5	Per cent size of counting circle	(F4.0)
5	contour interval	(F4.0)
5	Plot characters	(IX,40A1)
5	Format	(15A4)
5	KEY/NOKEY	(A4)
4	Key (if used)	
1	Data	
4	Key (if used)	
1	Data	Repeated
	Etc.	

90. Written on units 6, 8 and 9. Preceding each plot, the program prints the title of the data and the parameter value for the run. At the beginning of each run a table is printed showing the plot characters and the percentages they represent.

Variables and Subroutines

91. Input variables:

FMT
(characters) the input format

KPAR
(characters) the input option keyword

ALOC, BLOC, GLOC
vectors of direction cosines of counting locations

PCA
per cent size of the counting circle

CONTI
contour interval

CHAR
(characters) the plot characters

TITLE
(characters) title of data

NFILE
number of records in the input file

LENK
number of record numbers in the "key"

KEY
record numbers of input file records to be used

DATA
direction cosines of the directions to be plotted

WQ
weights of the directions to be plotted

ITD
unused variable.

92. Output variables:

CHAR

plot characters

CONT

density value for printing table

ID

title of data

NTOT

total number of data points

SUMW

sum of the weights

PCA

per cent size of the counting circle

CI

contour interval

PLOT

one line of the plot.

93. Subroutines:

READ

Same as KEY4

R100

Reads up to 100 records of input file using the specified format. Calling sequence:

R100(IOU,FMT,NQ,DATA,WQ,SWQ,ITD)

IOU= input unit

FMT= the format stored in array FMT

NQ = the number of records actually read. This will be less than 100 if an end-of-file is encountered.

DATA= an array containing the direction cosines.

WQ = an array of weights associated with the direction cosines.

SWQ= the sum of the weights in WQ

ITD= an integer quantity that must be in the input format but may not be used by the calling program.

R100K

Similar to R100, but this subroutine reads under the control of a key. Calling sequence:

CALL R100K(IOU,FMT,NQ,DATA,WQ,SWQ,ITD,KEY,LENK,KEYPTR,NUM)

IOU,FMT,NQ,DATA,WQ,SWQ and ITD are described under subroutine R100.

KEY= a string of record numbers.

LENK= the number of record numbers in the key.

KEYPTR= at the time of a call to this routine, the work of KEYPTR is one more than the records already read. Initially or after a file rewind, KEYPTR is 1, set by the calling program.

NUM= at call time this is one more than the record number of the last record that was read. Initially or under a file record, NUM is 1, set by the calling program.

PLØTO

Calls GRIDIO to read direction cosines of counting locations, reads parameters, and computes information about the counting circle. Calling sequence:

CALL PLØTO(IØU)

IØU= Input Output Unit of Direction Cosines file.

PLOT2

Performs the counting procedure for the current batch of datapoints. AA, BB, GG. Calling sequence:

CALL PLOT2(AA,BB,GG,JS,N,W)

AA,BB,GG contain direction cosines

JS= the spacing of these variables in the arrays, eg, the first direction cosines might be in AA(1), AA(4), AA(7),... so JS would be 3.

N= the number of data points in the batch.

W= weights

GRIDIO

Reads the direction cosines of the counting locations. Calling sequence:

CALL GRIDIO

RANGEX

For the given data point, RANGEX determines the submatrix of the print matrix that wholly contains the projection of the counting circle centred on that data point. Calling sequence:

CALL RANGEX(ST,CT,G,R,I1,I2,J1,J2)

ST,CT= SIN and COS of the trend of the data point.

G= SIN of the plunge of the data point.

R= maximum possible radius of the counting circle on the projection.

I1,I2,J1,J2= variables used to return matrix indices defining the required printed submatrix.

PLOT1

Fills count array with zeroes. Calling

sequence:

CALL PLOT1

PLOT3

Prints the plot. Calling sequence:

CALL PLOT3(IOU,ID)

IOU= output I/O unit printer

ID = the title of the data

BLNK

Used to overprint the plot to give density information. Calling sequence:

CALL BLNK(NUM,PLOT,K)

NUM= a row of integers that are truncated density values.

PLOT= row of characters corresponding to NUM that will be one line of the plot.

K= an integer value used for comparison. For each number in NUM that is less than K the corresponding character in PLOT is replaced by a blank. If the entire line is blanked out, an alternate return is taken. This routine is called repeatedly for the same line, with increasing values for the same line, with increasing values of K. The line will be overprinted after successive calls; higher density positions will be overprinted more times than lower density positions.

Data Structures

94. The deck containing the direction cosines of the counting locations was constructed as described below. Of 3713 print positions in the print matrix, 2933 are inside the projection and are used as counting locations. The direction cosines of these were computed, multiplied by 10,000 and rounded to the nearest integer. These integers between 0 and 9999 were stored in 2-byte (10 bytes per word) memory locations. To complete the matrix, the print positions outside the projection were assigned the three numbers 2,2,2. Taking the print matrix row by row, and taking each row from left to right, these numbers were written onto cards using 'A' format code, each number thus occupying two card columns. Numbering the print positions in this sequence, the numbers of the first and the last print positions on each card were placed at the beginning of the card, also using the format 'A2'. Each card carries 38 numbers: two indices, plus 3 numbers for each of 12 print positions.

95. If a key is used, the input data file consists of any number of identical records, each containing three direction cosines and a weight for one data point. These may be in any format, the format for reading them in the above sequence being supplied to the program on a parameter card.

Storage Requirements

Code	bytes:	words:
Main program	13120	5712
	12246	5286
Subroutines:		
READ	144	100
R100	54	44
R100K	76	62
GRID10	57	47
PLØT0	75	61
PLØT1	13	11
PLØT2	234	156
PLØT3	716	462
BLNK	30	24
RANGEX	76	62
Common blocks:		
ØNE		3779
	35115	
TWØ		11146
Total storage:	61956	26952

Sample Run

Parameter cards:

```
column 1
↓
01.0
01.0
0b123456789ABCDEFGHIJKLMNOPQRSTUVWXYZ****
(64X,3F5.4,F3.1,T59,I3)
```

```
TXFILEVJN
D2DM0002010R45920   222 -21001680 1  2478381496430259422JN 82 60-1205-8576 5000 15J    3171FD-9067 3001 2962SDSR5
D2DM0002010R45920   22  -21001680 2  2878381196429959422JN352 87-9889 1390  523 16J    3164FD 1480 8944 4220SDSR5
D2DM0002010R45920   222 -21001680 4  7478378196426559424JN 95 80  858-9811 1736 17J    3163FD -151 1730 9848SDSR5
D2DM0002010R45920   222 -21001680 9 12378374896422959425JN105 85 2578-9623  872 22J    3          0    0    0SDSR5
D2DM0002010R45920   222 -2100168010 15578372696420559426JN102 85 2071-9744  872 20J    3172FD 3175 1512 9361SDSR5
D2DM0002010R45920   222 -2100168011 18078371096418659427JN127 80 5927-7865 1736115J  F 3174FD-7862-5181 3368SDSR5
D2DM0002010R45920   222 -2100168023 38778357196403359435JN102 85 2071-9744  872 20K    3          0    0    0SDSR5
D2DM0002010R45920   222 -2100168025 41678355296401159435JN 91 69  163-9334 3584 17K    3170FD 9955  486  814SDSR5
D2DM0002010R45920   222 -2100168026 41778355196401059436JN104 82 2396-9609 1392 22J  F 3166FD-9695-2292  863SDSR5
D2DM0002010R45920   222 -2100168027 43478354096399859436JN102 74 1999-9403 2756 21J  F 3162FD  453 2899 9560SDSR5
.................   ... . ........   .................... .. ..  ........ .... .. .             .    .    ......
D2DM0003010R45920   227  0 21124024  078314096363059538JN 25 90-9063-4226    0 11 I               0    0    0SDSR5
```

CONTOUR VALUES AND CHARACTERS FOR THIS RUN ARE:

						I	>	18.0%	T	>	29.0%
						J	>	19.0%	U	>	30.0%
	>	0.0%	9	>	9.0%	K	>	20.0%	V	>	31.0%
1	>	1.0%	A	>	10.0%	L	>	21.0%	W	>	32.0%
2	>	2.0%	B	>	11.0%	M	>	22.0%	X	>	33.0%
3	>	3.0%	C	>	12.0%	N	>	23.0%	Y	>	34.0%
4	>	4.0%	D	>	13.0%	O	>	24.0%	Z	>	35.0%
5	>	5.0%	E	>	14.0%	P	>	25.0%	*	>	36.0%
6	>	6.0%	F	>	15.0%	Q	>	26.0%	*	>	37.0%
7	>	7.0%	G	>	16.0%	R	>	27.0%	*	>	38.0%
8	>	8.0%	H	>	17.0%	S	>	28.0%	*	>	39.0%

```
TXFILEVJN
  38 OBSERVATIONS WITH TOTAL WEIGHT OF     86.0
PERCENT OF TOTAL WEIGHT IN  1.0 PERCENT OF AREA
CONTOUR INTERVAL  1.0   CHARACTER SEQUENCE  123456789ABCDEFGHIJKLMNOPQRSTUVWXYZ****
                                        |
                                4441111
                          7AAA75552            11222323
                      555AAAAA7555            1111222553333
                    2999CCEA75522              11111233533322
                   33699966222                   111111111111111
                  3AAAA333                                  11  111
                 6AAAAAA                                         1111
                66666666                                          1111
           DDDDK6666666                                               1
          DDDDDD6666
         DDDDDDD
        DDDDDDD
        DDDDD


   22582222
   CCFA7522 33333
  CEEHHA755333333
  CERJHA758333333
  EKKNIA555333333                                                              1
 CKRPJD844 33333                                                              111
 CLNJFEA6631                                                                  135
 AGECBB93332                                                                  356
-AGGCBC73335317                         +                                     356-
 8CEEA955355333311                                                            138
 58A663665533333111                                                           11A
 333133355313333111                                                           14A
  1 333333311333111                                                           26
     333311  11111                                                            29
     111111                                                                   49
      1111                                                                    4
                                                                              2




                                                             777
                                                          7777777
           1                                             77777777
            1111                                         7777777
             1111                                         7777
               111
                 11
                    211111
                      333221                  11       2555
                          2221           1111111111 277
                                        111111114
                                        |
```

Fig 4 - Density diagram of discontinuity orientations from program WNDPLT.

DATA DESCRIPTION

96. This section documents two computer programs, WNSTAT and DIPS.

97. WNSTAT produces orientation statistics for weighted non directed data. Typically the program would be used to characterize a sample of a single element of a sub fabric; means and measures of the scatter about the mean are calculated in two different ways. The data displayed by WNDPLT are described both in their observed orientation and rotated so that the mean value lies at the centre of the projection. The rotated display allows the immediate appreciation of any asymmetry of the data distribution around their mean.

98. Statistical tests assess the significance of the asymmetry which may indicate small changes in the mean orientation of the fabric element across the domain. These changes may be examined by areal analysis techniques documented in Supplement 2-2.

99. DIPS produces histograms of relative and cumulative relative frequencies of observations. It is suitable for any numerical data but will find most use in the display of the distributions of dips.

DISCONTINUITY ORIENTATION WITH WNSTAT

100. Given a set of weighted non-directed orientations in three dimensions, the program describes the data by fitting two different theoretical distributions. One of these is Fisher's distribution; the other is the bipolar form of the Dimroth-Watson distribution. The fit of the data to the Fisher distribution is tested by means of chi-square tests of fit and the Rayleigh test for uniformity on the circle.

Solution

(1) The flow diagram for program WNSTAT is shown in Fig 5(a) and 5(b). Fitting the exp(K.cos theta) function. Watson and Irving (1957); Watson (1966).

Each data point is treated as a vector whose length is proportional to the associated weight, all weights being scaled so that their sum is equal to N.

(2) Test of fit of data to fitted Fisher function.

(a) Chi-square test of directions. Watson and Irving (1957).

Number of classes = J = N/5; minimum 4, maximum 12.

Degrees of freedom = J-3, since two estimated parameters are used (coordinates of mean).

Observed frequency for each class is the sum of the weights of the data points falling in each class, weights being scaled as above.

(b) Rayleigh test of directions. Durand and Greenwood (1958); Batschelet (1965).

Doubled angles are used, since the expected alternative to uniformity is a density function with a period of 180 degrees.

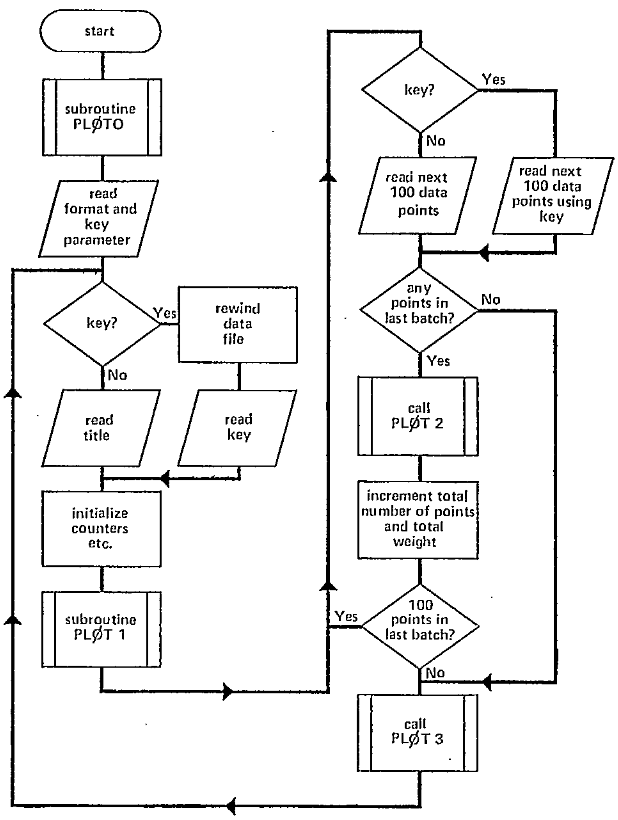

Fig 5(a) - Flow diagram for program WNSTAT - Part 1.

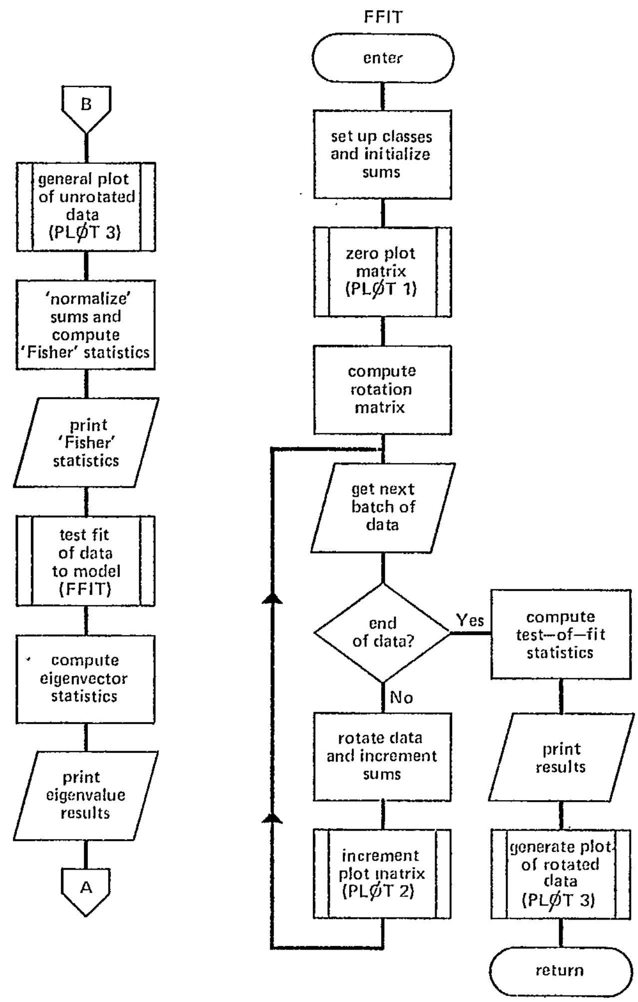

Fig 5(b) - Flow diagram for program WNSTAT - Part 2.

(This is because the expected alternatives to a single axially symmetric cluster are (1) two clusters, and (2) an elongated cluster; either alternative would give rise to a distribution about the computed mean having a 180-degree period.)

(c) Chi-square test of distance. Watson and Irving (1957).

Number of classes = 5

Degrees of freedom = 1, since all three estimated parameters are used in this test.

Observed frequencies computed as in chi-square test of directions.

(3) Fitting exp (K.cos-squared theta) function. Scheidegger (1965); Watson (1966).

Matrix used is:

SUM(XI*XI)	SUM(XI*YI)	SUM(XI*ZI)
SUM(XI*YI)	SUM*YI*YI)	SUM(YI*ZI)
SUM(XI*ZI)	SUM(YI*ZI)	SUM(ZI*ZI)

where ZI, YI, ZI represent the components of an individual observation vector whose length equals the associated weight, WI. The sum of the main diagonal elements of this matrix is sum (Wi). The matrix is multiplied by N/SUM(WI) before calculation of eigenvectors, so that its main diagonal elements (and therefore its eigenvalues) add up to N.

101. An azimuth limit, L, is specified for each data set processed. For each data point in the set, before the various sums are incremented, the azimuth read for the data point is compared with this limit. If the azimuth is between L and L+180, the direction is used as read; otherwise the direction cosines are multiplied by -1 and the new cosines used.

102. The program produces two equal-area diagrams of the data in each set - one projected onto the conventional horizontal plane, one projected onto the plane normal to the "Fisher" mean.

Program Capabilities

103. This program characterizes only single clusters of axes. The given directions may be on the upper or the lower hemisphere. If they do not form a single cluster as given, specifying a non-zero value of LIM causes the program to adjust the data so that all the azimuths lie within a single 180-degree sector.

Program Input and Output

104. Read from unit 1 - tape or disk. Input data file as generated by CONVRT, or a subset generated by KEY4. If the data files use the keys generated by KEY4, each record of the data file contains direction cosines and weight for each data point.

105. If no key is used, the data file must consist of a title record, followed by data records, as above, followed by one blank record. This sequence may be repeated any number of times.

106. Read from unit 2 - card reader. Plot parameters. First set of data records.

(1) One card containing the present size of the counting circle in columns 1-4 as a right-justified integer or with decimal point.

(2) One card containing the contour interval, same format as 1.

(3) One card containing the plot characters in columns 2-41. The character for the first contour interval should be blank, since this character is placed in all positions of the print matrix that are outside the projection.

107. Read from unit 3 - tape or disk. Direction cosines of counting locations. A file called DIRECTIONCOSINE is provided with those programs containing these cosines. Locations are written in A format.

108. Read from unit 4 - card reader. Keys as written by the program KEY4. Optional. If keys are being used, a KEY card in unit 5 input is needed. These cards are placed after the KEY card in the card deck (see order of input).

109. Read from unit 5 - card reader. Data parameters. Second set of data cards.

(1) One card containing the format of the data in columns 1 to 60. The list of variables read as follows.

Variable	Type
dir.cos.1	real*4
dir.cos.2	real*4
dir.cos.3	real*4
weight	real*4
azimuth	integer*4

(2) One card containing the word KEYb in columns 1 to 4 ("b" represents a blank) if a key is used, otherwise containing any other character sequences in columns 1 to 4.

(3) One card for each data set processed. Each card contains the azimuth limit for the corresponding data set as a right-justified integer in columns 1 to 3.

Variables and Subroutines

Subroutine GRID10: (See WNDPLT)

ALØC

vector of first direction cosines of counting locations.

BLØC

vector of second direction cosines of counting locations.

GLØC

vector of third direction cosines of counting locations.

Subroutine PLØTO: (See WNDPLT)

PCA

per cent size of counting circle

CØNTI

contour interval

ICØDE1

not used

CHAR

plot characters

WNSTAT

FMT

data input format

KPAR

data input option keyword

ID

data title.

Subroutine READ: (See KEY4)

NFILE

number of records in input data file

LENK

number of record numbers in key

KEY

key of record numbers

WNSTAT

LIM

azimuth limit to be used in assigning sense to axes.

Subroutines R100 and R100K: - See WNDPLT

DATA

direction cosines of orientations to be analyzed

W

weight of observation

WQ

vector of weights

ITD

vector of azimuths

Output variables:

WNSTAT

ID

data title

I

number of observations

SUMW

total of weights

AVG

average weight

MAX

maximum weight

MIN

minimum weight

RX,RY,RZ

components of resultant vector

R

length of resultant vector

RN

length of resultant vector divided by number of observations

XK

concentration parameter of best-fitting Fisher distribution

RT, RP

trend and plunge of resultant vector

DD, DP

dip direction and dip of plane normal to resultant vector

ALPH5, ALPH 1

95% and 99% confidence radii on mean

M(J)

eigenvalue

EN

eigenvalue divided by number of observations

T, P

trend and plunge of principal eigenvector

DD, DP

dip direction and dip of plane normal to principal eigenvector.

Subroutine CSDS1:

I

number of class

CØSTH

cosine of class limit

EF

expected frequency in class

ØF

observed frequency in class

SØF

sum of observed frequencies

CS

chi-square

IDF

degrees of freedom.

Subroutine CSDR1:

I

number of class

CLIM

class limit

EF

expected frequency in class

ØF

observed frequency in class

SØF

sum of observed frequencies

CS

chi-square

IDF

degrees of freedom.

Subroutine RYDR1:

Z

Rayleigh test statistic

Z95, Z99

95% and 99% points of the distribution of Z.

Subroutine PLØT3: - See WNDPLT

CHAR

plot characters

CØNT

contour value of plot character

ID

data file

NTØT

total number of data points

SUMW

sum of weights

PCA

per cent size of counting circle

CI

contour interval

CHAR

plot characters

PLØT

line of plot.

Subroutines for WNSTAT:

FFIT

Test if fit to a Fisher function. Calling sequence:

CALL FFIT(XK,RT,RP,XN,SUMW,ID)

XK = contration parameter of best fitting Fisher distribution.

RT,RP = trench and plunge of resultant vector

XN =

SUMW = total of weights

ID = data

EIGEN

Computes eigenvalues and eigenvectors of a real symmetric matrix. Diagonalization method originated by Jacobi and adopted by Von Neumann as presented by A. Ralston and H.S. Wilf (1962). Calling sequence:

CALL EIGEN(A,R,N,MV)

A = original matrix symmetric, destroyed in computation. Resultant eigenvalues are developed in diagonal of Matrix A in descending order

R = resultant matrix of Eigenvectors (stored columnwise, in same sequence as eigenvalues.)

N = order of matrices A and R

MV = input code

0 compute eigenvalues and eigenvectors

1 compute eigenvalues only (R need not be dimensioned but must still appear in calling sequence)

original matrix A must be real symmetric (storage Mode =1).

Matrix A cannot be in the same location as matrix R.

GTPRD

Premultiplies a general matrix by the transpose of another general matrix. Matrix transpose of A is not actually calculated. Instead, elements of Matrix A are taken by column rather than by row for postmultiplication by matrix B.

Calling sequence:

CALL GTRPD(A,B,R,N,M,L)

A = name of first input matrix

B = name of second input matrix

R = name of output matrix

N = number of rows in A and B

M = number of columns in A and rows in R

L = number of columns in B and R

Matrix R cannot be in the same location as matrix A.

Matrix R cannot be in the same location as matrix B.

All matrices must be stored as general matrices.

Data Structures

110. Structure of the input file is described under Input and Output for WNSTAT.

Storage Requirements

Code		bytes:	words:
Main program		13744	6166
		12246 (buffers)	5286
Subroutines	CSDS1	77	63
	CSDS2	105	69
	FFIT	2062	1074
	CSDS3	151	105
	RYDR1	4	4
	RYDR2	61	49
	LENGTH	16	14
	RYDR3	51	41
	DCTP	205	133
	R100	54	44
	PLØT0	75	61
	GRID10	57	47
	PLØT1	13	11
	BLNK	30	24
	CSDR1	44	36
	CSDR2	32	26
	SFIT	6	6
	CSDR3	151	105
	CØPY	54	44
	READ	145	101
	R100K	76	62
	PLØT2	234	156
	RANGEX	76	62
	PLØT3	716	462
	ROTMAT	131	89
common blocks:			
	ONE	35115	3779
	TWO	25612	11146
	CSDS	27	23
	CSDR	34	28
	RYDR	2	2
	EIGEN	432	282
	GMPRD	103	67
	GRPRD	100	64
Total storage required		66429	29681

Sample Run

Input:

column 1

↓

[direction cosines of counting locations for plots

1.0

1.0

bb123456789ABCDEFGHIJKLMNØPQRSTUVWXYZ****

(64X, 3F5.4, F3.1, T59, I3)

NØKEY

000

Input:

```
TTESTDATA
D2DM0002010R45920  222 -21001600 1  247838149643025942 2JN 82 60-1205-6576 5000 15J   3171FD-9067 3001 2962SDSR5
D2DM0002010R45920  222 -21001680 2  287838119642995942 2JN353 87-9889 1390  523 16J   3164FD 1480 8944 4220SDSR5
D2DM0002010R45920  222 -21001600 4  74783781964265595 24JN 95 80  658-9811 1736 17J   3163FD -151 1730 9848SDSR5
D2DM0002010R45920  222 -21001680 9 123783748964229594 25JN105 85 2578-9623  872 22J   3          0    0    0SDSR5
D2DM0002010R45920  222 -2100168010 155783726964205594 26JN102 85 2071-9744  872 20J   3172FD 3175 1512 9361SDSR5
D2DM0002010R45920  222 -2100168011 180783710964186594 27JN127 80 5927-7865 1736115J F 3174FD-7862-5181 3368SDSR5
D2DM0002010R45920  222 -2100168023 387783571964033594 35JN102 85 2071-9744  872 20K   3          0    0    0SDSR5
D2DM0002010R45920  222 -2100168025 416783552964011594 36JN 91 69  163-9334 3584 17K   3170FD 9955  486  814SDSR5
D2DM0002010R45920  222 -210168026  417783551964010594 36JN104 82 2396-9609 1392 22J F 3166FD-9695-2292  863SDSR5
D2DM0002010R45920  222 -2100168027 434783540963998594 36JN102 74 1999-9403 2756 21J F 3162FD  452 2899 9560SDSR5
.................  ... .  .......  .................... .. .. ........ .... .. .          .    .   ......
D2DM0003010R45920  227  0 21124024  078314096363059538JN 25 90-9063-4226    0 11 I              0    0    0SDSR5
```

Output:

(See WNDPLT: Contour values and characters for this run are:)

TTESTDATA (See Fig 6)

38 OBSERVATIONS WEIGHTS:- TOT: 86.0 AVG: 2.263 MAX:11.500 MIN: 1.000

DENSITY FUNCTION 1: EXPONENTIAL K COS THETA (FISHER)

RX	RY	RZ	R	R/N	K	TR	PL	DD	DP	95%	99%
9.41	-23.78	7.33	26.60	0.70008	3.2	291.6	16.0	111.6	74.0	15.4	19.4

CHI-SQUARE TEST OF DIRECTIONS

CLASS	DEGR	THEORET	ACTUAL
1	51.4	5.43	14.76
2	102.9	5.43	0.0
3	154.3	5.43	1.28
4	205.7	5.43	5.43
5	257.1	5.43	12.64
6	308.6	5.43	0.97
7	360.0	5.43	2.92
		TOTAL	38.00

CHI-SQUARE= 39.02 DF=4

RAYLEIGH TEST ON DOUBLED ANGLES

Z	95%	99%
11.71	2.98	4.53

CHI-SQUARE TEST OF DISTANCES

CLASS	COS	THEORET	ACTUAL
1	0.931	7.60	17.54
2	0.843	7.60	7.73
3	0.718	7.60	2.70
4	0.504	7.60	6.98
5	-1.000	7.60	3.05
		TOTAL	38.00

CHI-SQUARE= 18.95 DF=1

TTEST DATA - "FISHER" MEAN AT CENTRE (see Fig 7)

Fig 6 - Density diagram of discontinuity orientation from program WNSTAT.

Fig 7 - Density diagram of discontinuity orientation with Fisher mean at centre from program WNSTAT.

Output:

DENSITY FUNCTION @: EXPONENTIAL K COS-SQUARED THETA

EIGENVAL	EVAL/N	TR	PL	DD	DP
28.10	0.73935	296.7	9.6	116.7	80.4
8.68	0.22841	205.6	6.5	25.6	83.5
1.23	0.03225	82.3	78.4	262.3	11.6

DISCONTINUITY ORIENTATION WITH DIPS

111. From an input vector of observations, this program constructs a histogram of relative frequencies and a histogram of cumulative relative frequencies. This program draws histograms of the distribution of dips in a set of discontinuities, to be used in slope stability analysis.

Solution

112. A flow diagram is given in Fig. 8. Input consists of a vector of dips, the input format, and a title. Output consists of a vector of internal relative frequencies, a vector of cumulative relative frequencies, and the corresponding histograms.

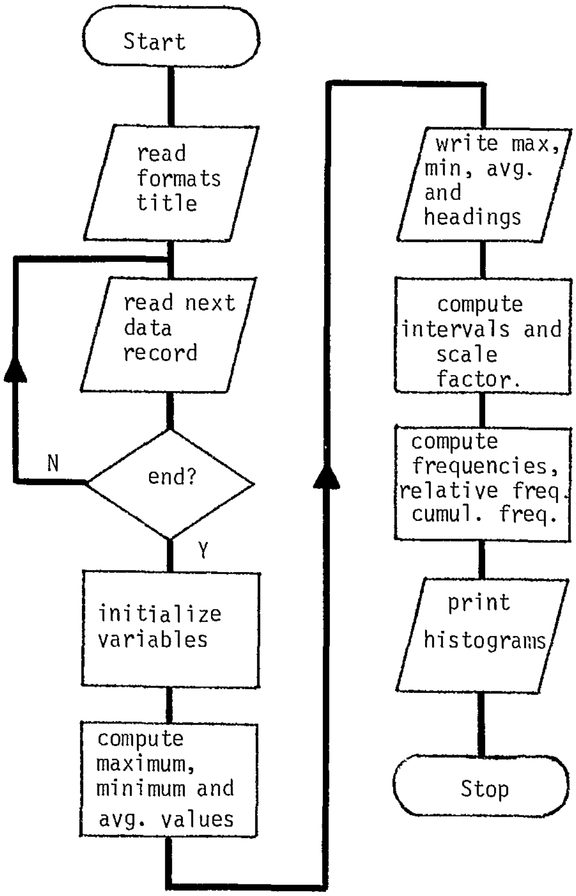

Fig 8 - Flow diagram for program DIPS.

Program Capabilities

113. Actual frequencies per cell should be 10,000. If all observations have the same value, a message is printed and the plot is ignored. If the values of the frequency scale cannot be printed in the format F7.3, the values are shifted by a power of 10 and this multiplier is printed at the bottom of the plot.

Program Input and Output

114. Read from unit 1 - tape or disk. Input data file. This is the data file created by CONVRT. The first record should be a title card; the test record should be blank.

115. Read from unit 5 - card reader. Format card. This is a single card containing a format for reading the dips from the input file. Typically this would be:

(61X,I3)
↑Column 1

Output:

116. Written on unit 6 - line printer. The output contains relative frequencies, cumulative relative frequencies, and the corresponding histograms.

Variables and Subroutines

Input variables:

FMT
Input format
TITLE
Title
X
Dips

Output variables:

TITLE
title
N
number of observations
XMIN
smallest dip
XMAX
largest dip
AVE
average dip
SZ
upper limit of first interval
IZ
zero
RZ
zero
ULIMIT
upper limit of interval
IFREQ
frequency in interval
REL
relative frequency in interval
CUM
cumulative relative frequency in interval
IREL
line of relative frequency histogram
ICUM
line of cumulative relative frequency histogram

SMULT

output scale factor

IC

number of observations used in histogram

Library and System Subroutines Required:

None.

Data Structure

117. The input file must contain a title in record 1, one dip in each succeeding record. All data records must have the same format.

Storage Requirements

code	bytes:	words:
Main program	2104	1092
	6144 (buffers)	3172
Subroutine HIST	1260	688
Total	9508	4952

Sample Run

Input:

col 1

↓

(61X,I3)

Sample Run Input

```
TTESTDATA
D2DM0002010R45920   222 -21001680 1  247838149643025942JN 82  60-1205-8576 50D0 15J   3171FD-9D67 3001 2962SDSR5
D2DM0002010R45920   222 -210D168D 2  287838119642995942 2JN352 87-9889 1390  523 16J   3164FD 1480 8944 4220SDSR5
D2DM0002010R45920   222 -21001680 4  747837819642655942 4JN 95 80  858-9811 1736 17J   3163FD -151 1730 9848SDSR5
D2DM0002010R45920   222 -21001680 9 123783748964229594 25JN105 85 2578-9623  872 22J   3          0    0   0SDSR5
D2DM0002010R45920   222 -2100168010 155783726964205594 26JN102 85 2071-9744  872 20J   3172FD 3175 1512 9361SDSR5
D2DM0002010R45920   222 -2100168011 180783710964186594 27JN127 80 5927-7865 1736115J F 3174FD-7862-5181 3368SDSR5
D2DM0002010R45920   222 -2100168023 387783571964033594 35JN102 85 2071-9744  872 20K   3          0    0   0SDSR5
D2DM0002010R45920   222 -2100168025 416783552964011594 36JN 91 69  163-9334 3584 17K   3170FD 9955  485  814SDSR5
D2DM0002010R45920   222 -2100168026 417783551964010594 36JN104 82 2396-9609 1392 22J F 3166FD-9695-2292  863SDSR5
D2DM0002010R45920   222 -2100168027 434783540963998594 36JN102 74 1999-9403 2756 21J F 3162FD  453 2899 9560SDSR5
.................   ...  . ........  .................... .. ..  ........ .... .. .           .    .   ......
D2DM0003010R45920   227  0 21124024  078314096363059538JN 25 90-9063-4226    0 11 I            0    0   0SDSR5
```

TTEST DATA (Fig 9)

```
TTESTDATA

NO. OF POINTS =     38

MINIMUM =   54.0000        MAXIMUM =   90.0000        MEAN =   79.8421

UPPER  FREQ   REL    CUM  |              RELATIVE FREQUENCY DIAGRAM               |          CUMULATIVE FREQUENCY DIAGRAM
LIMIT         FREQ   FREQ |                                                       |
52.000    0   0.0     0.0 |                                                       |
54.000    1   2.63    2.63|******                                                 |*
56.000    0   0.0     2.63|                                                       |*
58.000    0   0.0     2.63|                                                       |*
60.000    2   5.26    7.89|***********                                            |****
62.000    0   0.0     7.89|                                                       |****
64.000    0   0.0     7.89|                                                       |****
66.000    0   0.0     7.89|                                                       |****
68.000    1   2.63   10.53|******                                                 |*****
70.000    2   5.26   15.79|***********                                            |********
72.000    0   0.0    15.79|                                                       |********
74.000    1   2.63   18.42|******                                                 |*********
76.000    4  10.53   28.95|**********************                                 |***************
78.000    0   0.0    28.95|                                                       |***************
80.000    7  18.42   47.37|***************************************                |*************************
82.000    2   5.26   52.63|***********                                            |****************************
84.000    2   5.26   57.89|***********                                            |*******************************
86.000    9  23.68   81.58|*******************************************************|*******************************************
88.000    3   7.89   89.47|*****************                                      |***********************************************
90.000    4  10.53  100.00|**********************                                 |*****************************************************
```

Fig 9 - Relative frequency diagram and cumulative frequency diagram of discontinuity dips from program DIPS.

HISTOGRAMS OF SPACING AND SIZE

118. This section documents a program, HIST2V, which produces histograms of the size and spacing of discontinuities along a line traverse.

119. The distribution of size and spacing plays a part in theoretical models of the probability of occurrence of slides of a given size. These models are generally investigated when the fabric of the rock mass has been clearly established. It is therefore expected that the program will be used on data that have first been characterized by WNSTAT and shown to constitute a single fabric element.

120. A model for a dense random distribution of objects in three dimensions has yet to be obtained. However, the result of distributing objects at random along a line is a negative exponential distribution of the spaces between the objects. HIST2V tests the fit of both spacings and sizes to this distribution.

121. The program produces histograms of discontinuity size and spacing. The format of the data file is variable and must be specified by parameter cards. The variables size, weight, traverse number, traverse length and distance are extracted from a file assigned to logical unit 1. Within each traverse, distance increments are computed and spacings between observations in the traverse determined.

122. For each traverse in the file the following information is printed:

a. number of discontinuities

b. sum of weights ($\sum W_i$)

c. spacing of discontinuities ($L/\sum W_i$)

d. length of traverse (L)

e. trend of traverse

f. plunge of traverse

123. After all records of the file have been read, the mean values for size and spacing are determined. The size observations are assigned numeric values specified by parameter cards (see Data Inputs). All size observations are assumed to be at the midpoint of the size interval.

124. Size and spacing values are then tallied into frequency tables and plots are produced of the double natural logarithm of the reciprocal of % frequencies greater than a certain size (or spacing) against the size (or spacing). The total number of discontinuities and the total number of valid observations are printed.

125. Size and spacing of discontinuities are supposed to have a negative exponential distribution resulting from a Poisson process and a chi-square test is carried out for the significance of departures from the predicted distribution.

126. An additional test of fit is carried out according to Maguire (1951) for the spacings.

127. The program has the following options:

1. produce histograms and carry out test of fit for size only, spacing only, or both size and spacing
2. determine the mean spacing by dividing the sum of spacings by the number of measurements of spacing or by dividing the normal length by the total number of discontinuities in the file.

Solution

128. A flow diagram is given in Fig 10. The variable size is coded as an alphanumeric and is read with an A format. The variable spacing is numeric and is computed in the following steps.

1. Within each traverse, distance increments, d_i, are computed for observations 2 to N where N is the number of observations for a traverse.
2. The spacing is then computed as spacing, S_i, $= d_i/\text{weight}_i$, for observations 2 to N giving (N-1) spacings for each traverse.

129. Mean size is computed by dividing the sum of the size by the number of measurements of size. Mean spacing is computed as mean size or by dividing the normal length by the number of discontinuities in the file, depending on the value of the normal length specified in the first parameter card.

130. The negative exponential distribution has a cumulative distribution, exp ($-\lambda\chi$) where λ is the reciprocal of the mean of the distribution. The chi-square test of fit is carried out by solving the equation, P = exp ($-\lambda\chi$) for values of P = 0.75, 0.50, 0.25. 25% of the sizes (or spacings) should fall within each of the 4 classes defined by these three limits. The statistic

$$\sum_1^4 \frac{(fe - fo)^2}{fe},$$

where fe is 1/4 the total number of observations, fo is the number of observations in each of the 4 classes, can then be referred to tables of chi-square with 2 degrees of freedom.

131. The above test is carried out for both size and spacing. For spacing only, an additional test is carried out according to Maguire (1951). He gives

$$M = 2nk\ [\ln \tfrac{1}{k} \textstyle\sum T_i) - \tfrac{1}{k} \sum \ln T_i]$$

for individual spacings n = 1

$$\sum T_i = \sum (\text{spacings})$$
$$K = \text{total no. of spacings} = N$$

$$M = 2N\ [\ln (\tfrac{\sum T}{N}i) - \tfrac{1}{N} \textstyle\sum \ln T_i]$$

$$c = 1 + \frac{(N+1)}{(6N)} = \frac{7N+1}{6N}$$

$$\frac{M}{c} = (\frac{12\ N^2}{7N+1})\ [^{\ln}\ (\tfrac{\sum T}{N}i - \tfrac{1}{N} \textstyle\sum \ln\ T_i]$$

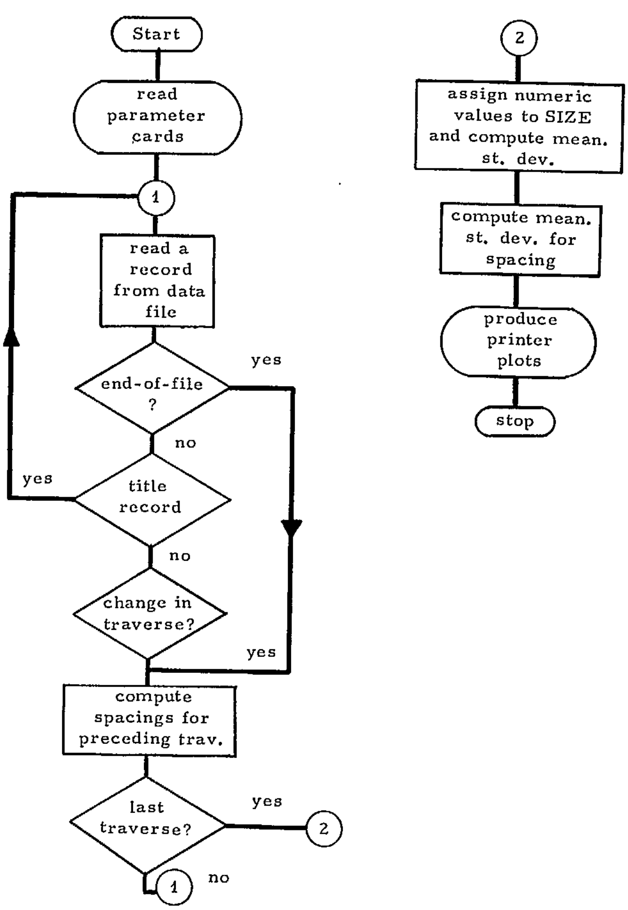

Fig 10 - Flow diagram for program HIST2V.

and this has χ^2 with (N-1) d.f.

Program Capabilities

Max no. of records per traverse = 100

Max no. of records per file = 3000

Max no. of frequency classes = 19

Max field width of each of the variables size, weight, traverse, scale, and distance = 8 characters

Program Options

1. Size only

   Spacing only

   Both size and spacing.

2. Mean spacing calculation

Program Input and Output

132. Read from unit 5 - card reader. Data input file. The format of this file is specified by the parameter cards below. The maximum size of each input variable is 8 character. The traverse identification and the size variable are assumed to be alphanumeric.

Format of parameter cards:

Card 1	Option card
Column	Information
1 - 7	Total normal strength
	≠ 0 used to calculate mean spacing
	= 0 not used, mean spacing is calculated as mean size
8 - 10	1OPT
	= 1 size only
	= 2 spacing only
	= 3 both size and spacing

Card 2	(optional, 1OPT=2 or 1OPT=3)
Column	Information
1 - 20	Name for size plot
21 - 23	Starting position in record of variable size
24 - 26	Field width of variable size
27 - 29	Number of decimal places in variable size
30 - 32	Number of frequency classes for variable size

Card 3	(optional, 1OPT=2 or 1OPT=3)
Column	Information
1 -20	Name for spacing plot
21 - 23	Starting position is record of variable weight
24 - 26	Field width of variable weight
27 - 29	Number of decimal places in variable weight.
31 - 32	Number of frequency classes for variable spacing

Card 4	(optional, 1OPT=2 or 1OPT=3)
Column	Information
21 - 23	Starting position in record of variable traverse no.
24 - 26	Field width of variable traverse no.
27 - 29	Number of decimal places in variable traverse no.

Card 5 (optional,1OPT=2 or 1OPT=3)

As card 3 for variable traverse length

Card 6 (optional,1OPT=2 or 1OPT=3)

As card 3 for variable distance

Card 7 (optional, 1OPT=2 or 1OPT=3)

As card 3 for variable trend

Card 8 (optional, 1OPT=2 or 1OPT=3)

As card 3 for variable plunge

Notes:

1. Maximum field width for all variables is 8.
2. Variable traverse is assumed to be alphanumeric (so the number of decimal places is ignored for this variable).

Card 9	(optional, 1OPT=1 or 1OPT=3)
Column	Information
1 - 4, 5-8, ... 53-56	Lower limits of frequency intervals for variable size

Card 10	(optional, 1OPT=1 or 1OPT=3)
Column	Information
1-4, 5-8, ... 57-60	Alphabetic frequency classes starting with name of class less than lowest numeric frequency interval for variable size

Note: Alphabetic names must be left adjusted in the four character field.

Card 11 (optional, 1OPT=2 or 1OPT=3)

As card 9 for variable spacing.

Card 12 (optional, 1OPT=2 or 1OPT=3)

As card for variable spacing.

133. Read from unit 1 - tape or disk. The format is specified by the parameter cards read by unit 5, with these restrictions:

a. the maximum record length is 255 characters

b. a title record may or may not be present. If it is, it must be the first record of the file; it is identified by a T in column 1. The title starts in column 2 and may be a maximum of 120 characters long

c. no decimal points are allowed in the data.

Written on unit 6 - line printer. Histograms plots and chi-square tables, F-statistic.

Variables

Input variables:

Parameter cards:

Card 1

NORLEN

total normal length

IØPT

option for size, spacing or both

Card 2

NAME

name for size plot

NPØS

Starting position in record of data file of variable size

NWD

field width of variable size

NDC

number of decimal places in variable size

NR

number of frequency classes for variable size

Card 3

NAME

name for spacing plot

NPØS

starting position in record of data file of variable weight

NWD

field width of variable weight

NR

number of frequency classes for variable spacing

Card 4

NPØS

starting position in record of data file of variable traverse name

NWD

starting position in record of data file of variable traverse name

Card 5

NPØS

starting position in record of data file of variable traverse length

NWD

field width of variable scale

Card 6

NPØS

starting position in record of data file of variable distance

NWD

field width of variable distance

NDC

number of decimal places in variable distance

Card 7

NPØS

starting position in record of data file of variable traverse trend

NWD

field width of variable traverse trend

Card 8

NPØS

starting position in record of data file of variable traverse plunge

NWD

field width of variable traverse plunge

Card 9

TABLE

lower limits of frequency intervals for variable size

Card 10

ALFTAB

alphabetic codes for frequency classes for variable size starting with the code of the class less than lowest numeric frequency interval

Card 11

TABLE

lower limits of frequency intervals for variable spacing

Card 12

ALFTAB

alphabetic codes for frequency classes for variable spacing starting with the code of the class less than lowest numeric frequency interval.

Data File:

TITLE

title record

RECØRD

other records.

<u>Output</u>:

Please see the description of the parameters of subroutines PRITAB, CHISQR and TSTØFT.

<u>Subroutines and Function Subprograms</u>

<u>Subroutine CØNVRT</u>: extracts the variables <u>Size</u>, <u>Weight</u>, <u>Traverse no</u>., traverse length and distance from the record just read from the data file.

Calling Sequence:

CALL CØNVRT (VAR, NPØS, NWD, NDC, SIZE, WT, TRAV, LENGTH, DIST, TREND, PLUNGE).

Parameters:

VAR

Is the location of a LOGICAL*1 vector of length 255 giving the information read from the data file

NPØS

Is the location of an integer vector of length 5 giving the starting positions in the record of the five variables in the order specified above

NWD

Is the location of an integer vector of length 5 giving the field widths of the five variables

NDC

Is the location of an integer vector of length 5 giving the number of decimal places of the five variables

SIZE

Is the location of a single precision real variable in which the value for SIZE will be returned

WT

Is the location of a single precision real variable in which the value for <u>weight</u> will be returned

TRAV

Is the location of a double precision real variable in which the value of traverse name will be returned

LENGTH

Is the location of an integer variable in which the value of <u>traverse length</u> will be returned

DIST

Is the location of a single precision real variable in which the value of <u>distance</u> will be returned

TREND

Is the location of an integer variable in which the value of <u>traverse trend</u> will be returned

PLUNGE

Is the location of an integer variable in which the value of <u>traverse plunge</u> will be returned.

<u>Subroutine RDATA</u>:

A subroutine that generates formats to read the input data read on FORTRAN device 1 (TAPE 1).

Calling sequence:

CALL RDATA(NPOS,SWD,NDC,TYPE,ALFVAL,INTVAL, RELVAL)

Parameters:

NPOS

Is the location of the integer variable giving the starting position of the variable to be read

NED

Is the location of an integer variable giving the number of decimal places in the input variable

TYPE

Is the location of an integer variable giving the type of the input variable

ALFVAL

Is the location of a double precision variable in which alphanumeric data will be stored

INTVAL

Is the location of a double precision variable in which integer data will be stored

RELVAL

Is the location of a double precision variable in which real data will be stored.

Subroutine STATS:

Computes number of observations, mean and standard deviation for given frequency classes.

Calling sequence:

CALL STATS (F, NR, AVER, STDEV, ISF, CLASS)

Parameters:

F

Is the location of a single precision real vector giving the frequency counts for the classes

NR

Is the location of an integer variable giving the dimension of vectors F and CLASS

AVER

Is the location of a single precision real variable for the mean, returned by the routine

STDEV

Is the location of a single precision real variable for the standard deviation returned by the routine

ISF

Is the location of an integer variable for the number of observations retrieved by the routine

CLASS

Is the location of a single precision vector giving the arithmetic means of the class intervals.

Subroutine TALLY:

Tallies alphanumeric variables into a frequency table.

Calling sequence:

CALL TALLY (VAR, NVAR, N, NWD, TABLE, F, NC)

Parameters:

VAR

Is the location of a single precision real vector giving the alphanumeric variables to be tallied

NVAR

Is the location of an integer variable giving the dimension of vector VAR

N

Is the location of an integer variable giving the number of values in TABLE (max. 15)

NWD

Is the location of an integer variable containing the field width of variable VAR

TABLE

Is the location of a single precision real vector giving the alphanumeric table values

F

Is the location of a single precision real vector in which the resulting frequency counts will be stored

NC

Is the location of an integer variable in which the sequence numbers of the elements of vector TABLE will be stored

NDC

Not used.

Subroutine CUMPLT:

Sets up titles and arrays to be used by modified IMSL routine USPLX.

Subroutine CHISQR:

Computes and prints Chi-squared tables

Calling sequence:

CALL CHISQR (VAR, NVAR, ISF, QUART, TITLE, NAME)

Parameters:

VAR

Is the location of a real vector giving the values of the variable for which the Chi-squared value is to be determined

NVAR

Is the location of an integer variable containing the dimension of variable VAR

ISF

Is the location of an integer variable ($\leq$ NVAR) giving the number of observations of variable VAR

QUART

Is the location of a single precision vector of length 4 giving the lower limits of the four frequency classes

TITLE

Is the location of a LOGICAL*1 vector of length 120 giving the title of the data file

NAME

Is the location of a single precision real vector of length 5 giving the name of the variable as read from the parameter cards.

Subroutine ASSIZE:

Assigns numeric values to the alphabetically coded frequency classes and tallies the alphabetic values into a one-way frequency table.

Calling sequence:

CALL ASSIZE (SIZE, S, ITT, CLASS, TABLE, ALFSIZ, NR, F)

Parameters:

SIZE

Is the location of a single precision vector giving the alphabetically coded variables

S

Is the location of a single precision vector in which the numeric values for SIZE will be stored

ITT

Is the location of an integer variable giving the dimension of SIZE and S

CLASS

Is the location of a single precision vector in which the values of the midpoints of the frequency classes will be returned

TABLE

Is the location of a single precision vector giving the lower limits of the numeric class intervals

ALFSIZ

Is the location of a single precision vector containing the alphabetically coded frequency classes

NR

Is the location of an integer variable giving the dimension of CLASS, TABLE, ALFSIZE and F

F

Is the location of a single precision vector in which the frequency counts will be stored.

Subroutine YVALUES:

Computes modified cumulative frequencies (Y-values for histogram plots).

Calling sequence:

CALL YVALUES (Y, N, ITT, YGT, YLS)

Parameters:

Y

Is the location of a real vector of length, N. On input, Y contains simple frequencies; on output, it contains cumulative frequencies

N

Is the location of an integer variable giving the dimension of Y, YLS, YGT

ITT

Is the location of an integer variable giving the number of observations of Y

YLS

Is the location of a single precision vector in which the relative cumulative frequencies will be stored

YGT

Is the location of a single precision vector in which the double logarithm reciprocal of % frequencies greater than Y will be stored.

Subroutine USPLH: (entry point USPLX)

A modified IMSL routine to produce printer plots of up to ten functions. Its calling sequence is completely documented in its source listing.

Subroutine PRITAB:

Prints frequency table, etc.

Calling sequence:

CALL PRITAB (IØBS, ICØN, TABLE, ALFTAB, F, N, PCTLS, TITLE, NAME, AVER)

Parameters:

IØBS

Is the location of an integer variable giving the number of observations

ICØN

Is the location of an integer variable giving the number of discontinuities

TABLE

Is the location of a single precision real vector giving numeric frequency classes

ALFTAB

Is the location of a single precision real vector giving alphabetically coded frequency classes to corresponding TABLE

F

Is the location of a single precision real vector containing frequency counts

N

Is the location of an integer variable giving the dimention of TABLE, ALFTAB, F AND PCTLS

PCTLS

Is the location of a single precision real vector giving relative cumulative frequency counts

TITLE

Is the location of a LOGICAL*1 vector giving the title as read from the data file

NAME

Is the location of a single precision real vector giving the name as read from the parameter cards

AVER

Is the location of a single precision real variable giving the mean value.

Subroutine CSTDEV:

computes mean and standard for one variable.

Calling sequence:

CALL CSTDEV(X,NØBS,MIS,VMIS,NVALID,AVER,VAR,SD, IWRITE)

Parameters:

X

Is the location of a single precision vector giving the observations

NØBS

Is the location of an integer giving the number of observations

MIS

Is the location of an integer. If MIS = 0, all observations are included in the computation. If MIS = 1, all observations that are equal to VMIS are excluded.

VMIS

Is the location of a single precision real variable to be used in connection with MIS. IF MIS = 0, VMIS may have any value

NVALID

Is the location of an integer which will contain the number of observations

AVER

Is the location of a single precision real variable in which the mean will be stored

VAR

Is the location of a single precision real variable in which the variance will be stored

SD

Is the location of a single precision real variable in which the standard deviation will be stored

IWRITE

Is the location of an integer input variable

IWRITE = 0 - no printout

IWRITE = 1 - printout of mean, variance, standard deviation, number of observations

USMNMX is an IMSL routine required by USPLH. Their calling sequences are completely documented in their source listings.

Subroutine BDCØU1:

An IMSL routine to tally observations into a one-way frequency table. Its calling sequence is completely documented in its source listing.

Subroutine VSORTP:

An IMSL routine sorts arrays by algebraic values. Its calling sequence is compeltely documented in its source listing.

Subroutine TSTØFT:

Computes and prints F-statistic according to Maguire (1951). It also prints mean and standard deviation.

Calling sequence:

CALL TSTØFT (ITT, S, TITLE)

Parameters:

ITT

Is the location of an integer variable giving the number of observations of S

S

Is the location of a vector of length ITT for which the F-statistic is to be calculated

TITLE

Is the location of a LOGICAL*1 vector of length 120 giving the title as read from the data file.

Subroutine VERTST:

Error message generation.

Calling sequence:

CALL VERTST(IER,NAME)

IER = Error parameter. Type Tn when

Type = 128 - terminal error

64 - warning with fix

32 - warning

n = error code relevant to calling routine

NAME = name of the calling routine

Data Structure

133. The input file consists of unblocked records of a maximum record length of 255. An optional title record must start with a "T" in column 1 and it must not exceed 121 characters.

Storage requirements

code	bytes:	words:
HIST2V	31757	13295
	6144 (buffers)	3172
Subroutine CØNVRT	172	122
Subroutine STATS	42	34
Subroutine TALLY	112	74
Subroutine CUMPLT	12344	5348
Subroutine CHISQR	227	151
Subroutine ASSIZE	133	91
Subroutine YVALUS	125	85
Subroutine USPLH	1020	528
Subroutine PRITAB	162	114
Subroutine CSTDEV	121	81
Subroutine UERTST	104	68
Subroutine USMNMX	26	22
Subroutine BDCØU1	325	213
Subroutine VSRTPM	306	198
Subroutine TSTØFT	217	193
Subroutine RDATA	302	194
Total	53639	23933

Sample Run

Input:

Parameter cards

```
column 1
↓

 1583.1  2
SPACING             80  3  1 15
                     5  4
                    27  4
                    36  4
                    21  3
                    24  3
 0.6 1.1   2 3.5   6  11  20  35  60 110 200 350 60011002000
F       G       H       I       J       K       L       M   N
```

Output:

Fig 11

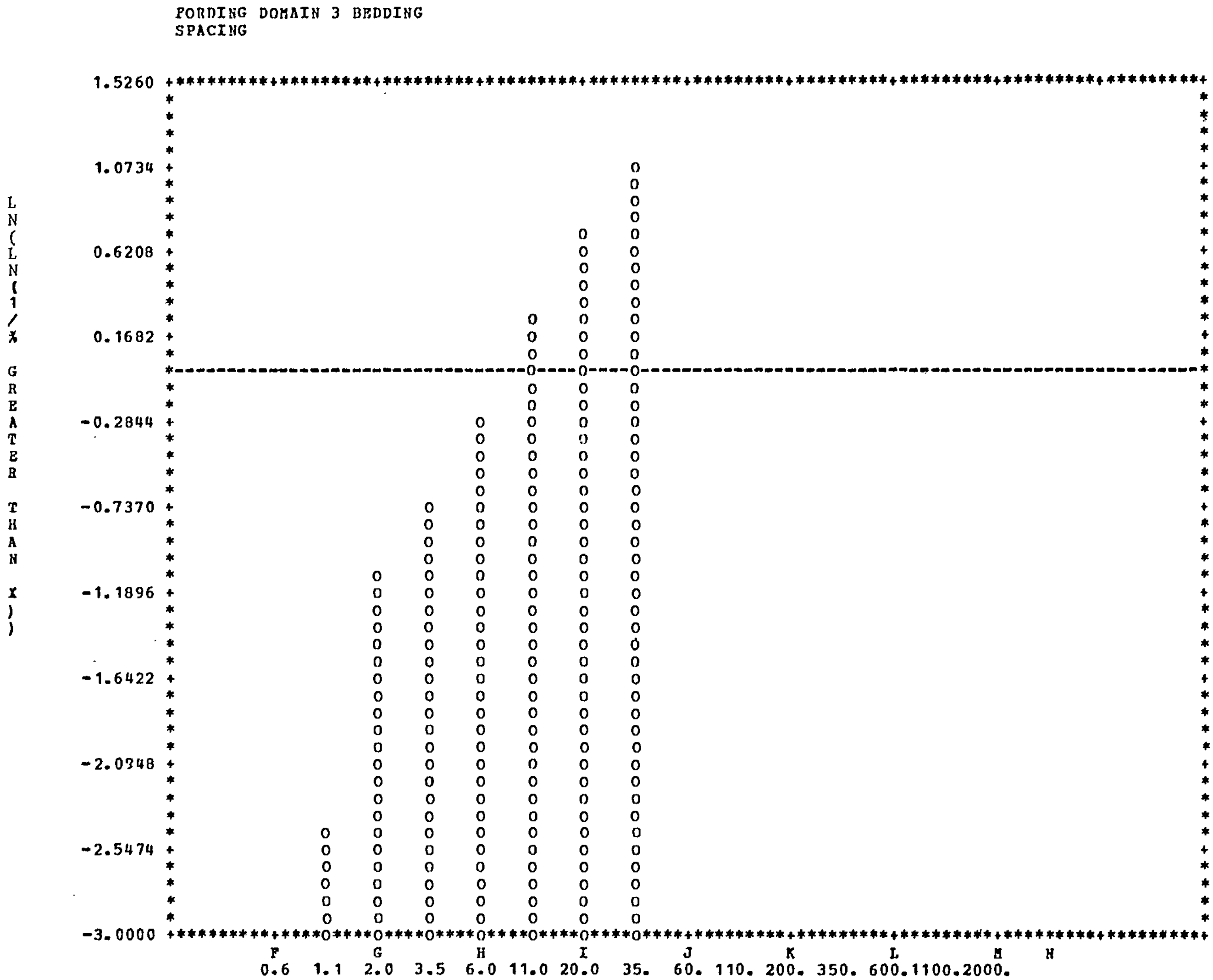

Fig 11 - Plot of discontinuity spacing from program HIST2V.

FORDING DOMAIN 3 BEDDING

TEST OF HOMOGENEITY OF SPACINGS F= 292.44
HAS CHI-SQUARED DISTRIBUTION WITH 253 DEGREES OF FREEDOM

MEAN OF OBSERVED SPACINGS IS 9.73
STANDARD DEVIATION OF OBSERVED SPACINGS IS 12.93
MEAN OF LOGARITHMS OF OBSERVED SPACINGS IS 1.60
STANDARD DEVIATION OF LOGARITHMS OF OBSERVED SPACINGS IS 1.23
20:35.24 7.6 RC=0

CHISQUARED TABLE FOR Z
LOG(Z)=LAMBDA*X, WHERE
LAMBDA=1/MEAN SPACING
X= SPACING

CLASS LIMITS	FE	FO	(FE-F))**2/FE
0.0 - 1.33	64.	57.	0.7
1.33- 2.00	64.	42.	7.3
2.00- 4.00	64.	53.	1.7
4.00 OVER	64.	102.	23.3

CHISQUARED= 33.02

FREQUENCY TABLE FOR VARIABLE SPACING
TOTAL NO. OF OBSERVATIONS= 254 TOTAL NO. OF DISCONTINUITIES= 297

	X	NO. LESS THAN X	% LESS THAN X
F	0.6	12.	4.72
	1.1	21.	8.27
G	2.0	70.	27.56
	3.5	95.	37.40
H	6.0	133.	52.36
	11.0	181.	71.26
I	20.0	221.	87.01
	35.0	240.	94.49
J	60.0	252.	99.21
	110.0	253.	99.61
K	200.0	254.	100.00
	350.0	254.	100.00
L	600.0	254.	100.00
	1100.0	254.	100.00
M	2000.0	254.	100.00

MEAN SPACING (1/LAMBDA) = 5.33

TOTAL NORMAL LENGTH OF TRAVERSES IS 1583.

TRAVERSE NUMBER	NUMBER OF DISCONTINUITIES	SUM OF WEIGHTS	SPACING OF DISCONTINUITIES	TRAVERSE LENGTH	TRAVERSE TREND	TRAVERSE PLUNGE
51	6	20.50	10.59	217	259	-5
52	7	18.50	15.14	280	262	5
54	8	28.70	16.45	472	258	0
55	28	141.60	2.08	294	252	-2
56	10	29.40	11.33	333	247	1
57	31	324.60	1.34	435	228	-14
65	6	17.40	21.55	375	259	7
67	4	16.60	18.73	311	246	1
68	6	53.40	7.49	400	250	-2
..	.	.....	....	...	...	.
89	7	54.50	2.57	140	241	-28

```
  1   TFDRDING DOMAIN 3 BEDDING
  2   D2DM  51 3 126967  259 -5 217150 3  538053989391006968 7BG 66 23-15B9-3570 9205 33L  1          0    0   OMDSR2
  3   D2DM  51 3 126967  259 -5 21715014  548053979391006968 7BG 46 17-2031-2103 9563 62 B            0    0   OMDSR2
  4   D2DM  51 3 126967  259 -5 217150 5  898053639390936969 0BG 83 27 -553-4506 8910 27L  2          0    0   OMDSR3
  5   D2DM  51 3 126967  259 -5 217150 6  920050009390936960 0BG 83 27 -553 4506 8910 27L  2          0    0   OCMSR1
  6   D2DM  51 3 126960  259 -5 21715015 163805291939079696 96BG 83 27 -553-4506 8910 27 B            0    0   OCMSR1
  7   D2DM  51 3 126967  259 -5 21715010 176805278939077696 97BG 62 26-2058-3871 8988 29L  1          0    0   OMDSR3
  8   D2DM  52 3 126967  262  5 280220 1   0804790938850696 92BG 96 15  271-2574 9659 30L Z 2172FD-9255 3587 1215MDSR2
  9   D2DM  52 3 126967  262  5 280220 3  358047559388456968 9BG 61 20-1658-2991 9397 25K Z 2         0    0   OMDSR2
 10   D2DM  52 3 126967  262  5 280220 8  858047069388386968 5BG 88 19 -114-3254 9455 25L Z 2         0    0   OMDSR2
 11   D2DM  52 3 126967  262  5 28022022 140804652938831696 80BG 65 16-1165-2498 9613 29 B   174FD 9896  527 1336MDSR2
...   .... .. . ......  ...... ...... . .................... .. ............ .... ...  .          .    .   ......
298   D2HM  89 3 116700  241-28 14023023 123805265941558674 06BG 61 18-1498-2703 9511 58 F            0    0   OCOLS4
```

DISCONTINUITY LOCATION WITH TRAVMP

135. This section documents program TRAVMP which uses a Calcomp drum plotter to produce a map of survey traverses. The map shows the start of each traverse, the horizontal projection of the traverse line and an identification number. An additional number may be plotted at each traverse origin, being usually the topographic height at the origin of the traverse.

136. Output from the program is a display and no analysis is performed. The display is useful for arranging retrievals from the data bank from an area with complex boundaries and in assessing and reporting coverage of the mine by the survey.

137. Given a list containing the coordinates, orientations and identification numbers of the survey traverses, the program produces on the Calcomp drum plotter a map showing the starting point of each traverse, the horizontal projection of the traverse and the identification number. An additional number may be plotted beside each traverse.

Solution

138. The program produces a Calcomp plotter map of survey traverses and a flow diagram is given in Fig 12.

139. The program determines maximum and minimum X and Y from data. Then 1/20 of the range of the data is taken and subtracted from the minimum X and Y, respectively, to obtain values corresponding to the lower borders of the map. These values are also added to the maximum X and Y to obtain the other two borders of the map. A frame is drawn coinciding with these border values and a title plotted below the lower border, using the first record of the traverse data. A grid is also drawn as determined by a starting grid value and interval read in by the program. The program then plots the individual points with elevation and identification (traverse number). Optimization of plotting is achieved as described below. Using the extreme X and Y values corresponding to the borders of the map, the program subdivides the whole rectangular region into a 60 x 60 grid, determines the number of observations within each grid square and also gives pointers to locate all observations within a grid square. This helps the program to do a faster scan on the 60 x 60 matrix to optimize plotting.

140. The size of the map is determined by a scale factor read from a parameter card. The scale factor is the reciprocal of the number of co-ordinate units to be represented by one inch on the map. For example, if the co-ordinates are in tenths of feet, and a scale of 100 feet to the inch is required, the scale factor will be 0.001.

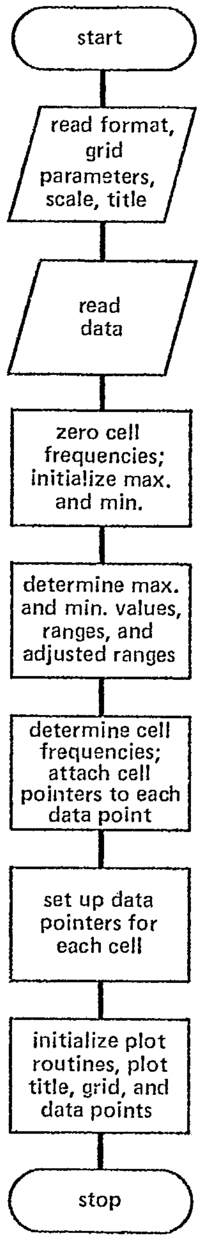

Fig 12 - Flow diagram for program TRAVMP.

141. The size of the traverse symbols is determined by a relative scale factor. The factor used for the symbols is the co-ordinate factor multiplied by the relative factor. In the above example, if the traverse lengths were given in feet and were also to be plotted to a scale of 100 feet to the inch, the relative scale factor would be 10.0.

Program Capabilities

142. Maximum number of traverses is 1500 at present. If the starting X or Y grid line value is less than minimum of data error, messages will be given and the program will stop. If the starting value is so high that another grid line below it and the frame of the map can be accommodated, this will be done.

Program Options

143. None.

Program Input and Output

144. Read from unit 1 - card reader. Input data. These cards follow the parameter cards read from unit 5.

145. First record is a title up to 80 characters. Subsequent records contain the following quantities, to be read using the format given on unit 5.

I = integer, A = alphanumeric

easting	I
northing	I
number to be printed beside traverse	I
character string (maximum 8 characters) to be printed beside traverse	A
traverse trend	I
unused field	I or A
traverse length	I

146. Read from-unit 5 - card reader. Parameters. These cards come before those read from unit 1.

(1) One card specifying format of data.

(2) One card specifying grid lines (all numbers right-justified integers).

Columns	Contents
5-10	starting X-value
15-20	X-interval
25-30	starting Y-value
35-40	Y-interval

(3) One card containing the co-ordinate scale factor.

(4) One card containing the relative scale factor. Both scale parameters must appear in columns 1 to 15, either as right-justified integers or with

decimal point.

147. Written on unit 6 - line printer. Listing of input.

148. Unit 9 is a scratch file - tape or disk.

149. Plotter output - unit number is user defined. This output is specified by the plot routines. These routines such as the Calcomp plot routines, are in a system library and may vary from those used in the example.

Variables and Subroutines

Input Variables:

FMT
data record format

MINGRX
starting value for grid lines, X-direction

INTGRX
number of data units between grid lines, X-direction

MINGRY
starting value for grid lines, Y-direction
number of data units between grid lines, Y-direction

SCALE
scale of map

SCALEL
scale of traverse

TITLE
title of map

IX
X-coordinates

IY
Y-coordinates

IZI
elevation (printed beside traverse)

IZ2
(characters) identifying label

IZ3
length of traverse

IDUM
not used

IZ4
trend of traverse.

External Subroutines:

150. The following system subroutines are called by the program. These subroutines are part of the standard Calcomp plotter subroutine package available from Calcomp.

151. Subroutine PLOT may be utilized to draw straight lines between 2 given points with the pen either up or down during the move.

152. One also has the choice of shifting the plot origin to the new pen position if so desired; this is usually convenient when going from one graph to another. Another function of PLOT is to complete a given plot by closing the file containing the plotter instructions.

PLOT (XPAGE,YPAGE,IC):

Moves the plotter pen in a straight line from its present position to the new position specified, with pen either up or down during the move. The coordinates of the new pen position, appearing in the argument list, are converted to plotter commands and put out on tape.

XPAGE, YPAGE

Page coordinates of the new position to which pen is to be moved (specified in inches from the current plot origin corresponding to XPAGE, YPAGE = 0.0).

IC

Integer whose most important function is to determine whether the pen is up or down, and if current plot origin is to be changed or not.

(a) if the absolute value of the units digit of IC is 2, the pen is lowered before it is moved to the new position specified by XPAGE, YPAGE; if it is 3, the pen is raised before moving.

(b) if in addition, IC is negative instead of positive, the plot origin will be reset to coincide with the new pen position. A block address is also written on the tape to permit searching of it for a specific plot, and this address is incremented by one each time PLOT is called with a negative IC (starting from 1).

(c) if IC=12 or 13, then subroutine OFFSET (page 20) should have been called previously and the page coordinates are then adjusted as follows: XPAGE = (XPAGE-XOFF) /XFAC; YPAGE = (YPAGE-YOFF)/YFAC.

(d) if IC≥25, then a block address of the value of IC is written on the tape which is then closed. (A negative IC in this case will override closing of the tape).

NOTES:

(a) IC values of 22 and 23 should not be used.

(b) For the purpose of terminating a plot and closing the tape IC=999 must be used.

To produce plot annotations consisting of one or more characters (or symbols), subroutine SYMBOL may be used. This routine may be used in two different ways:

(a) to plot a single character referenced by a number which will select that character from the set of the available ones.

(b) to plot a series of characters stored in some array.

SYMBOL (XPAGE, YPAGE, HEIGHT, BCD(I), THETA, NS)

YPAGE,YPAGE

This will be, for most of the characters, the location (page coordinates) of the lower left-hand corner of the grid containing the character to be plotted (often lower left-hand corner of character). However, for centered symbols this is the location of the grid center.

HEIGHT

height (inches) of characters to be plotted.

BCD(I)

Case (a) - Plotting one character at a time. Under this option BCD(I) is an integer which selects the appropriate single character to be plotted. Its value ranges from 0 to 127 since there are 128 characters available. For the centered characters this value + 128 will yield a character height of 4/7 the normal value.

Case (b) - Plotting a series of characters. Here BCD(I) specifies the Ith location in array BCD from which plotting of characters is to start. (Subscript I not needed if starting point is first location of BCD).

Characters may be put into BCD by means of a DATA or TYPE statement or read in from cards using A4 format if BCD is single precision, and A8 if double precision.

THETA

angle (degrees) at which the characters are to be plotted.

NS

In case (b) above, this parameter is positive and determines how many characters from array BCD are to be plotted, starting from the specified (by subscript I) location of that array. In case (a) above, NS should be set equal to -1 if the pen is to be up while moving the location (XPAGE,YPAGE) or -2 if the pen is to be down on the paper.

NOTE:

SYMBOL uses subroutine PLOT.

The subroutine NUMBER plots a given number at a specified location on plotting paper. This routine makes use of symbol after some pre-processing which consists of converting a real number (floating point) to its fixed-decimal equivalent.

NUMBER (XPAGE, YPAGE, HEIGHT, FPN, ANGLE, ± NN):

XPAGE, YPAGE, HEIGHT, ANGLE

these parameters are the same as the corresponding ones in SYMBOL.

FPN

the name of the floating point number to be converted and plotted.

±NN

controls precision of the conversion of the number FPN. If NN > 0, it specifies the number of digits to the right of the decimal point that are plotted, after rounding. For NN > 6, nothing is plotted. If NN = θ, only the integer portion of the number and a decimal point are plotted, after rounding. For NN = -1, only the integer portion of the number is plotted, with no decimal point. For NN <-1, |NN| -1 digits are truncated from the integer portion of FPN, after truncation.

NOTE: For FPN greater than 1,000,000.0 nothing is plotted. $FPN*10^{nn}$ should be less than 8,400,000.

The LINE subroutine may be used to display data values on a graph in either or both of two ways:

(a) by drawing straight line segments between points.

(b) by drawing a symbol at each data point with the option of joining points with straight line segments also.

X and Y values of data must be stored in two separate arrays and the two scale parameters discussed in the description of SCALE must follow the data points in each array. LINE is really the short way of plotting many points with one call to that routine instead of calling PLOT as many times as there are data points. Also the former routine accepts any values while in the latter all points are in page coordinates.

LINE (X,Y,NK,J.L):

X, Y

single precision arrays containing X and Y values to be plotted, and the two scaling parameters XMIN, XDELTA, YMIN, YDELTA (analogous to those in description of SCALE, AXIS)

N

number of data point to be displayed

K

repeat factor, same as in description of SCALE, (normally = 1)

J

parameter that determines which representation of data points is desired. J=0: straight line segments between points only. J>0 plot of a symbol at data points in addition to line segments. J=1 will produce a symbol at every data point. J=2 at every second point, etc. Value of L then determines symbol selected. J<0: plot of symbols only.

L

integer that occurs opposite the symbol selected. If only a line is drawn, L is still necessary and may be put equal to zero.

Subroutine LINE functions as follows:

For each pair of data values X(I), Y(I), $1 \leq I \leq N$, the conversion to page coordinates is made, ie,

XPAGE = (X(I) - XMIN) / XDELTA

YPAGE = (Y(I) - YMIN) / YDELTA

(ie, LINE draws a line such that values (XMIN, YMIN) coincide with the current plot origin of (XPAGE=0.0, YPAGE=0.0).

For J>0 and J=0 a call of PLOT (XPAGE,YPAGE,2) is made for each point except the first. For that point, the pen index is put equal to 3 instead of 2. This explains why the pen does not have to be moved to the position of the first point in pen up position before calling LINE. LINE is optimized internally so that if the current pen position happens to be near the end of a data line to be drawn, the line will then be drawn backwards. For J>0 and J<0 a call to SYMBOL is made using a symbol height of 0.08 inches.

NOTE:

Locations X(N*K+1), X(N*K+K+1), Y(N*K+1), Y(N*K+K+1) must be filled with the values of the scale parameters XMIN, XDELTA, YMIN, YDELTA before calling LINE, either automatically by a call to SCALE, or else by means of arithmetic assignment statements.

The plotter commands generated by subroutine PLOT require a work region of specified size. Subroutine PLOTS sets up this area and therefore a call to that routine should be the first statement of a plot program ie,

CALL PLOTS (BUFFER, NBUF):

BUFFER

The name of the single precision work area for the plotter commands; it must also appear in a DIMENSION statement with the number of locations = NBUF/4.

NBUF

The size (in bytes) of the above work area. The larger this number is, the more efficient plotter I/O operations are; however, if it is too large, the probability of an I/O error increases. A value of 16384 bytes is about the optimal value that may be used if enough storage is available. (The PLOTS program limit for NBUF is 2^{15}-1).

Subroutines for TRAVMP:

PPLOT determines minima, maxima, ranges, map borders, sets up 60 x 60 grid for optimization of plotting, then calls routines to do grid, frame and plot of all traverse information.

calling sequence:

CALL PPLOT(N,MINGRX,INTGRX,MINGRY,INTGRY,TITLE, SCALE)

N = number of data points

MINGRX = starting value for grid lines in X direction

INTGRX = number of data units between grid lines, X direction

MINGRY = starting value for grid lines, Y direction

INTGRY = number of data units between grid lines in Y direction

TITLE = title of the set of data

SCALE = scale of the map

PGRID draws frame and grid lines.

Calling sequence:

CALL PGRID(MINGRX,INTGRX,MINGRY,INTGRY,IXMIN, RANGE,IXMAX,SCALE)

MINGRX,INTGRX,SCALE,MINGRY,INTGRY - see plot

IXMIN = returned minimum value of X

RANGE = returned range

IXMAX = returned maximum value of X

PPTS plots all traverse locations, elevations, and numbers in grid.

Calling sequence:

CALL PPTS(I,J,IXMIN,SCALE)

I = index of starting value of X to be plotted (XMIN)

J = index of final value of X to be plotted

IXMIN = smallest value of X to be plotted

SCALE = scale of the map.

Data Structures

153. Structure of the input data file is described under "data inputs". Structure of the output plot description file depends on the system plotting subroutines. Internally, a system of pointers is used to optimize plotting. The plot is divided into a 60 x 60 matrix of cells. As each data point is assigned to a cell, the indices of this cell are placed in the appropriate locations in the vectors IS and JS, and the appropriate element of the frequency matrix L is incremented. The vector POINT2 is now used to contain pointers from each cell to the data points that fall in that cell. Using these pointers, the program can plot the traverses with a minimum of wasted pen movement.

Storage Requirements

		bytes:	words:
Main Program		351	233
		6144 (buffers)	3172
Subroutines:	VALID	51	41
	PPLØT	10,315	430
	PGRID	275	189
	PPTS	262	178
Common blocks:			
	MAP1	21,451	9001
	MAP2	21,989	9208
Total user program		60,838	22,452

Sample Run:

Input:

```
 (T37,I6,1X,I6,T15,I5,T7 ,A7,T51,I3,1X,I3,1X,I4)
    102000      5000     97000      5000     GRANISLE 73 TRAV MAP GRID
0.001
1.0
GRANISLE 1973 TRAVERSES
*0101/3HV0026/23429/  /   /   9/  W/106880/106680/ 51/+04/1000/0/25/ 90/
*0101/3HV0027/23436/  /   /   9/  W/105330/103240/ 10/-02/ 680/C/ 2/ 90/
*0101/3HV0028/23717/  /   /   8/  W/105570/107550/ 41/-04/ 871/0/14/ 90/
*0101/3HV0029/23809/  /   /   8/  W/108530/109540/ 45/-02/ 759/0/21/ 90/
*0101/3HV0030/23779/  /   /   8/  N/111460/110670/ 82/  0/ 997/0/30/ 90/
*0101/3HV0031/23118/  /   /  10/  W/107540/105370/ 46/+06/ 910/0/31/ 90/
*0101/3HV0032/22844/  /   /  11/  W/109600/107300/ 51/+02/ 992/0/21/ 90/
*0101/3HV0033/23760/  /   /   8/  S/106000/ 95990/290/  0/ 994/C/30/ 90/
*0101/3HV0034/23386/  /   /   9/  S/112040/ 98280/240/-02/ 914/0/44/ 90/
*0101/3HV0035/23407/  /   /   9/  S/110330/ 97390/250/  0/ 955/0/27/ 90/
...................   .   .  ...  .................. ... ......... ...
*0101/3PM0053/23042/  /   /  10/  S/117420/107452/159/ +7/ 680/0/09/ 90/
```

OUTPUT:

TRAVMP8 (Fig 13)

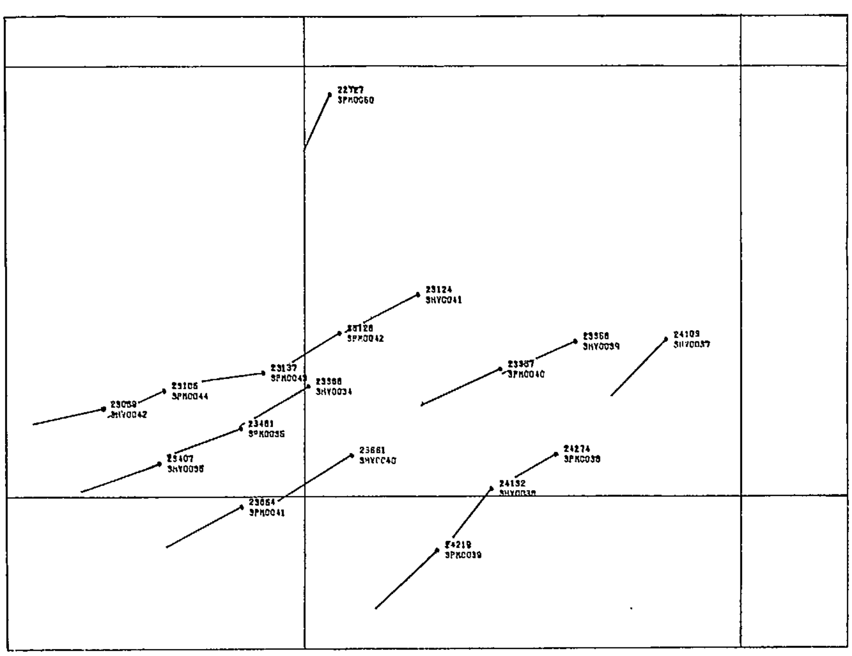

Fig 13 - Plot of traverse start, horizontal projection, and identification number with program TRAVMP.

REFERENCES

1. Batschelet, E. "Statistical methods for the analysis of problems in animal orientation and certain biological rhythms"; AIBS Monograph; 1965.

2. Durand, D and Greenwood, J.A. "Modifications of the Rayleigh test for uniformity in analysis of two-dimensional orientation data"; Jour. Geol.; 66; pp 229-238; 1958.

3. Maguire, B.A., Pearson, E.A., Wynn, A.H.A. "The time intervals between industrial accidents"; Biometrika, 37, pp 168-180; 1951.

4. Ralston, A. and Wilf, H.S. "Mathematical methods for digital computers"; John Wiley and Sons, New York, Chapter 7, 1962.

5. Scheidegger, A.E. "On the statistics of the orientation of bedding planes, grain axes, and similar sedimentological data; U.S.G.S. Prof. Pap. 525-C; pp 164-167; 1965.

6. Watson, G.S. and Irving, E. "Statistical methods in rock magnetism"; Mon. Not. Roy. Astron; Soc, Geophysical Supplement 7; pp 289-300; 1957.

7. Watson, G.S. "The statistics of orientation data"; Jour. Geol., 74; pp 786-797; 1966.