Information Circular 8624

BuMines IC 8624

A Computer Program for Clustering Data Points on the Sphere

By Robert J. Shanley and M. Ashraf Mahtab
Denver Mining Research Center, Denver, Colo.

UNITED STATES DEPARTMENT OF THE INTERIOR
Rogers C. B. Morton, Secretary

BUREAU OF MINES
John D. Morgan, Jr., Acting Director

This publication has been cataloged as follows:

Shanley, Robert J

A computer program for clustering data points on the sphere, by Robert J. Shanley and M. Ashraf Mahtab. [Washington] U.S. Bureau of Mines [1974]

58 p. illus. (U.S. Bureau of Mines. Information circular 8624)

Includes bibliography.

1. Cluster analysis--Computer programs. I. U.S. Bureau of Mines. II. Mahtab, M. Ashraf, jt. auth. III. Title. (Series)

TN23.U71 no. 8624 622.06173

U.S. Dept. of the Int. Library

CONTENTS

ILLUSTRATIONS

A COMPUTER PROGRAM FOR CLUSTERING DATA POINTS ON THE SPHERE

by

Robert J. Shanley [1] and M. Ashraf Mahtab [2]

ABSTRACT

The computer program, HYBMODE, has been developed to meet the need for a fast and economical method for clustering data points on the sphere. The algorithm used in this development is an extension of D. Wishart's one-level mode analysis. The program can be used to cluster either vectorial or axial data; it is currently being used by the Bureau of Mines to cluster fracture orientations.

INTRODUCTION

An understanding of many phenomena necessitates the collection and analysis of orientation data by physical and earth scientists. Some examples of the types of data collected are magnetic orientations, axes of glacial till pebbles, crystallographic axes, and fracture orientations. Since such measurements can be conveniently represented by unit vectors, it is natural that the methods of cluster analysis be applied to collections of points on the unit sphere. When the data are vectorial the entire sphere may be involved. However, when the data are axial, only one hemisphere is involved. The algorithm used in this report constitutes an improvement over the author's previous work (1)[3] which was concerned only with axial data, did not cluster the entire sample, and used a unit cell four times as large as the one currently employed.

SUMMARY OF ALGORITHM

One-Level Mode Analysis

The algorithm used in program HYBMODE is an adaptation of Wishart's one-level mode analysis (2). The purpose of one-level mode analysis is to divide a set of points in any coordinate space into a collection of distinct subsets representing meaningful classifications of the data.

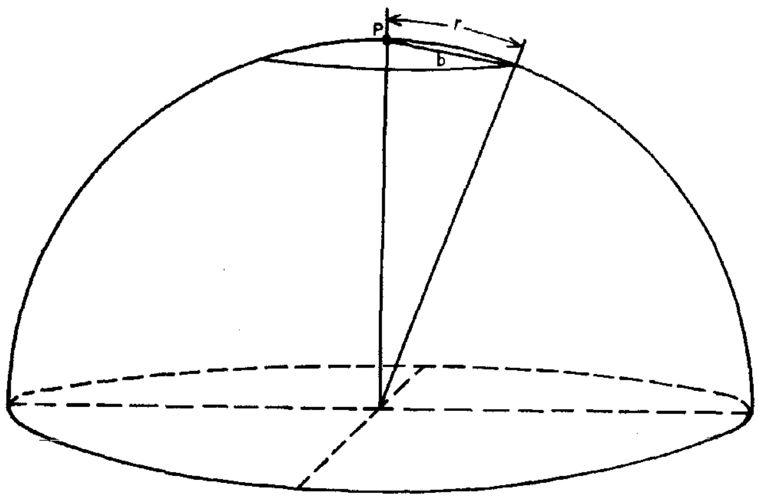
[1] Mathematician.
[2] Physical scientist.
[3] Underlined numbers in parentheses refer to items in the list of references at the end of this report.

To start a one-level mode analysis, two numbers must be chosen: r, a positive real number which is a distance and k, a positive integer. Spheres of radius r are constructed about every data point. The centers of those spheres containing k or more observations are called dense points. The dense points are then clustered by assigning two dense points separated by a distance less than or equal to r to the same cluster. At this stage, the algorithm has defined distinct sets of points, each point being dense. Next, the nondense points are allocated to clusters using some criterion; in this report, a nondense point is allocated to the cluster containing the closest dense point. Now, every point belongs to a cluster and the clusters are distinct. This marks the end of a one-level mode analysis.

The Hybrid Mode Analysis

A major drawback of one-level mode analysis is that the user must choose both r and k. To make the procedure less arbitrary an additional step is introduced. A value is chosen for r but a test for randomness is used to determine the value of k, namely, a point is defined to be dense if there is a 95 percent probability that the observations within a sphere of radius r about that point do not consitute a random grouping. The randomness test is based on the Poisson distribution (1, p.7).

The addition of the above step to one-level mode analysis achieves two advantages: (1) the user makes only one decision, namely, in choosing the size of r and (2) there is at least a 95-percent probability that each cluster contains a nonrandom grouping. The latter advantage results in dividing the data into subsets that represent meaningful classiciations.



NOTE: Percent of hemispherical surface contained in sphere of radius $b = 2 \sin(\frac{r}{2})$ about the point P is $(1 - \cos r)$.

FIGURE 1. - Spherical counting cell.

Application to the Sphere

For vectorial data, the sphere is divided into 800 equal area cells and for axial data, the hemisphere is divided into 400 equal area cells. The user must specify whether his data are vectorial or axial, since this information influences the randomness test described earlier. A value must be chosen for r in degrees and the value of k is determined by the randomness test. Then, for each point, the observations contained in a sphere of radius $b = 2 \sin\left(\frac{r}{2}\right)$ about that point are counted (see fig. 1). Each point for which this count is k or more is called dense. However, no cell may contain more than one dense point. This restriction on the number of dense points is introduced to minimize the computational effort involved in the analysis of large samples. Points in a cell containing a dense point belong to the cluster which contains that dense point. From this point on, the algorithm proceeds as described in the previous two sections.

The value of r is given in degrees and then converted to an Euclidean distance to provide the user with a better idea of the size of the counting cell being used (see fig. 1).

The Objective Function

Different values of r produce different partitions of the data. In some cases knowledge of the structure of the partitions formed as r assumes different values may not be sufficient. It may be desirable to select that partition which is "best" in the sense that it minimizes or maximizes some variable associated with the partitions. For example, a minimum variance partition may be most desirable. HYBMODE evaluates the function:

$$\psi = \sum_{i=1}^{M} \sum_{j=1}^{N_i} \| C_i - X_i^j \|^2 + \sum_{i=1}^{M-1} \sum_{j=i+1}^{M} \| C_i - C_j \|^2$$

where M is the number of clusters in the partition,

N_i is the number of observations in the i^{th} cluster,

C_i is the mean of the i^{th} cluster,

X_i^j is the j^{th} datum of the i^{th} cluster,

and $\| a-b \|$ denotes the Euclidean distance from a to b.

Note that the definition of "best" employed here is subjectively determined and implies a prior knowledge of the distribution of the data. The function given above has, however, performed well for the several examples of actural and hypothetical data analyzed by the authors. The hypothetical data consist of several artifically generated hemispherically normal (1, appendix D) clusters. These results are provided in the program description under the heading "example output."

Application to Fracture Orientations

When the attitude of a rock fracture is measured, the values of azimuth of the dip and dip (see fig. 2) give the attitude of a plane which approximates the fracture surface. Now, given any plane in three-dimensional Euclidean space, there is a line perpendicular to that plane (a normal to the plane) which passes through the coordinate origin (0,0,0). This normal must intersect the unit sphere (that is, the set of all points (X,Y,Z) such that

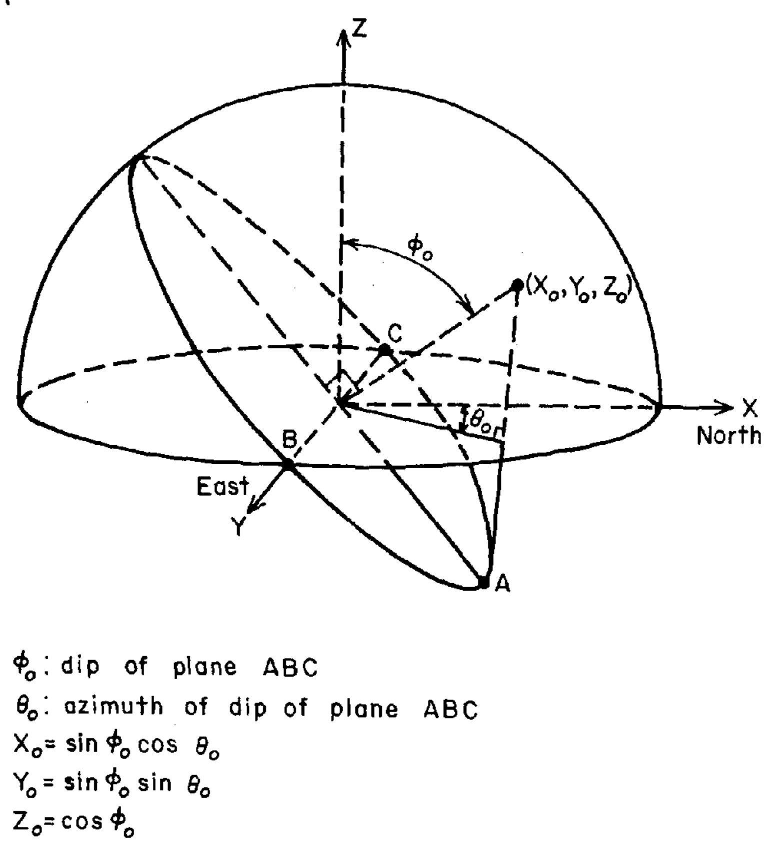


FIGURE 2. - Representation of fracture orientations by points on the unit sphere.

$X^2 + Y^2 + Z^2 = 1$) at two distinct points: (X_1, Y_1, Z_1), (X_2, Y_2, Z_2), where either $Z_1 \geq 0$ or $Z_2 \geq 0$.

For one of these two points, the following relationship holds: $X_1 = \sin\phi_1 \cos\theta_1$, $Y_1 = \sin\phi_1 \sin\theta_1$, $Z_1 = \cos\phi_1$, where ϕ_1 and θ_1 are the dip and azimuth of the dip respectively, of the plane of approximation (for axial data, $Z_1 \geq 0$). Since the set of all such points (X_1, Y_1, Z_1) comprises the sample that is actually analyzed, no distinction is made between the planes approximating fracture surfaces and the fracture surfaces themselves.

PROGRAM DESCRIPTION

Summary of Subroutines of HYBMODE

The program HYBMODE is written in Fortran IV and is designed to run on a CDC 6400 computer. It is 803 cards long. When dimensioned to handle 500 data points, $75{,}600_8$ words of memory are required to load. The main program flow chart is provided in figure 3. The two subroutines POLARC and SPRPAG have been adapted from another Bureau of Mines publication (3). Subroutine SORT has been adapted from a publication of the Kansas State Geological Survey (4).

HYBMODE Main Program

Calls input and initialization routines, sorts data into patches, and defines randomness test for spherical counting cell. Clusters dense points, allocates non-dense points, and outputs results with value of objective function

Subroutine CHANGE (M, N)

Points belonging to M^{th} dense point are reallocated to N^{th} dense point.

Subroutine ALLOC

Allocates each nondense point to the nearest dense point.

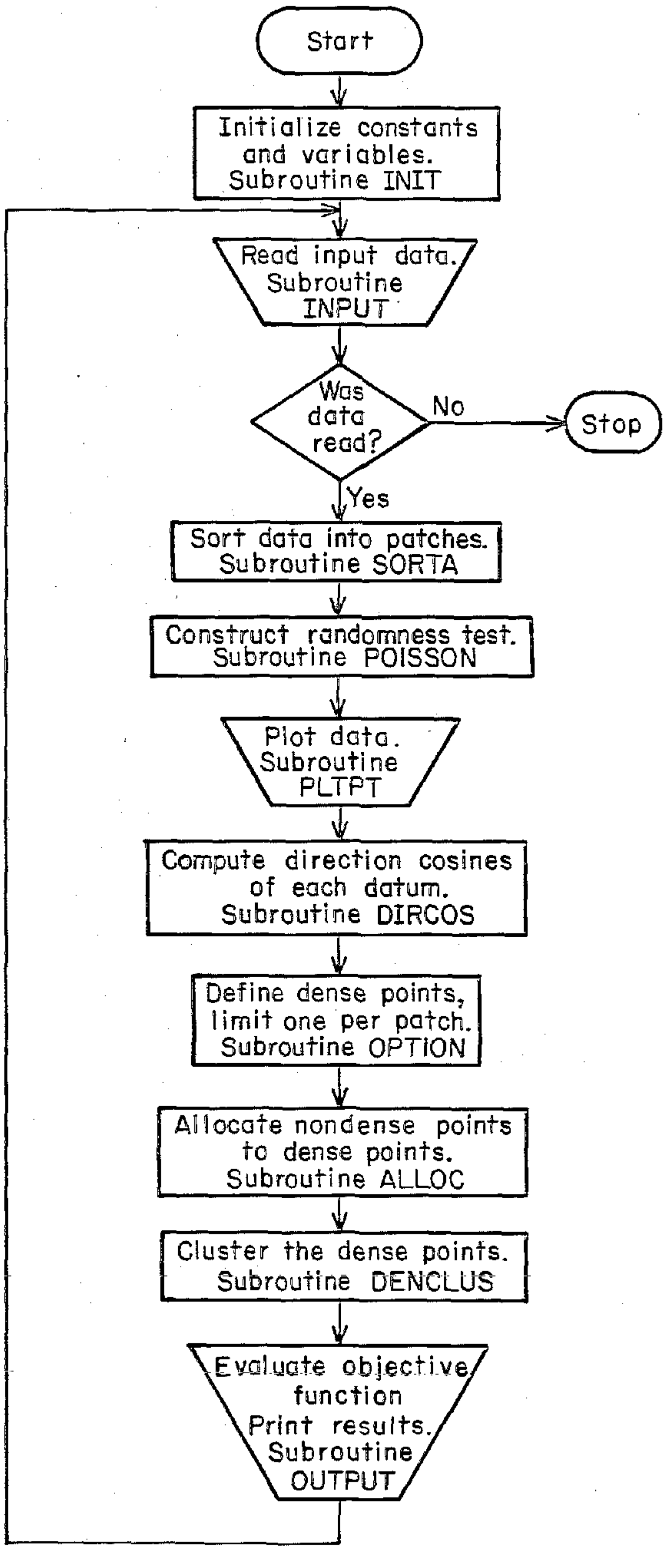


FIGURE 3. - Flow chart of main program.

Subroutine POLARC

Prints "*" at the center and at every 10° interval around the boundary of the output graph. Places N, E, S, W at 0°, 90°, 180°, 270°, respectively, and prints assembled page.

Subroutine SPRPAG (IX,IY,DIN,AZN)

Calculates page coordinates (IX,IY) of a point with spherical coordinates (1,AZN,DIN).

Funtion DIST (I, J)

Computes distance from I^{th} observation to J^{th} observation.

Function DIST2 (K, J)

Computes distance from K^{th} dense point to J^{th} dense point.

Function DIST1 (KK, JJ)

Computes distance from KK^{th} observation to JJ^{th} dense point.

Subroutine SORTA (KKL, LL3)

Sorts data into patches.

Subroutine OUTPUT

Prints contents of each cluster and evaluates the objective function.

Subroutine INPUT

Reads input data and program options.

Subroutine POISSON

Formulates randomness test for spherical counting cell.

Subroutine OPTION (KK1)

Defines dense points.

Subroutine BPLTPT (NN,K)

Loads arrays CU1, CU2 with observations belonging to NN^{th} dense point. There are K such points.

Subroutine PCTPLT (KK1)

Plots RLEVEL(J) at the projected (polar equal area) midpoint of patch J.

Subroutine DIRCOS (A,D,N,SL,SM,SN)

Calculates direction cosines of points, given azimuths of dip and dips.

Subroutine SPHRPT (X,Y,Z,D,A,R,)

Computes dip and azimuth of the dip of a point, given its (X,Y,Z) coordinates.

Subroutine DENCLUS

Clusters the dense points.

Subroutine PLTPT (A,D,N)

Given the azimuths and dips of a set of points, plots them (polar equal area projection).

Subroutine INIT (RADIUS)

Initializes constants and subscripted variables.

Subroutine SUMCOS

Computes (X,Y,Z) coordinates of resultant of cluster CU1, CU2.

Subroutine SORT (X,Y,N)

Sorts the N elements of X and Y into decreasing order of X.

Function DIST3(K, J)

Computes distance from the negative of the K^{th} point to the J^{th} point.

Subroutine XLEVEL (KK1)

Computes density of each cell.

Description of Output Produced by HYBMODE

The density (percent of sample) in each patch is computed and plotted (polar equal area projection) at the projected midpoint of the patch. Plots of the sample, the dense points, and the contents of each cluster are provided. The mean of each cluster is also computed. A summary of the above results and the values of the objective function corresponding to the various radii (RDIS) are also provided in the computer output.

Input Instructions

The data deck structure for program HYBMODE is given below:

Card No.	Columns	Contents	Format
1	1-80	HED	8A10
2	1-5	NOBS	
2	6-10	LANS	
2	11-15	ISPHER	
2	16-27	RDIS	
2	28-39	R1	
2	40-51	R2	3I5, 3F12.5
			12F5.0
3-NOBS/8=K-1	1-80	AZ(I), DIP(I) I=1, NOBS	
K	1-8	CONTINUE	A10
K+1	1-5	LANS	
K+1	6-17	RDIS	
K+1	18-29	R1	
K+1	30-41	R2	I5, 3F12.5
K+2	1-8	CONTINUE	A10
K+3	-	LANS	-
etc.	-	etc.	-

The user may select several option lists by use of the CONTINUE card.

More than one data deck may be submitted at a time by assembling them in order in which they are to be run. They need not be separated by any special card.

All observations must be coded in the form (θ, ϕ) where θ denotes azimuth of the dip and ϕ denotes the dip. The spherical coordinates of this observation when plotted on the unit sphere are (R, θ, ϕ). The spherical coordinate system is presented in figure 2.

Glossary of Frequently Used Variables

Variable name	Definition
HED......................	Heading identifying data.
NOBS.....................	Number of observations in data deck.
ISPHER...................	=1, Vectorial data; $\neq$ 1, Axial data.
RDIS.....................	Radius of spherical counting cell.
AZ(I)....................	Azimuth of dip of I^{th} observation (degrees).
DIP(I)...................	Dip of I^{th} observation (degrees).
PCTSURF..................	Percentage of hemispherical (ISPHER=0) or spherical (ISPHER=1) surface contained in a spherical counting cell.
KPCUT....................	Randomness test value.
NDENSE...................	Number of points whose spherical counting cells contain KPCUT or more points.
SLD(I), SMD(I), SND(I)..	The (X,Y,Z) coordinates respectively of the I^{th} dense point.
B(J).....................	Index of the dense point to which the J^{th} observation belongs.
SL(K), SM(K), SN(K).....	(X,Y,Z) doordinates of the point whose spherical coordinates are (1, AZ(K), DIP(K)).
LPATCH(K)...............	Number of observations lying in patches 1 through K inclusive.
KBAND(K)................	Number of the band containing patch #K.
IBND(K).................	Number of patches in band #K.
DANG(K).................	Range of possible dip values for points in K^{th} band is: $DANG(K-L) \leq DIP < DANG(K)$.
ICUT....................	Number of points with Dip > 90°.
LANS....................	=1 An observation is counted among those points in sphere of radius RDIS about point Y if either the distance from X to Y or the distance from -X to Y is less than or equal to RDIS. $\neq$1 An observation X is counted among those points in sphere of radius RDIS about point Y if the distance from X to Y is less than or equal to Y.
R1......................	Radius of data plot and dense point plot. If R1 = 0, these plots are not provided.
R2......................	Radius of density plot. If R2=0, this plot is not provided.

PROGRAM LISTING

```
      PROGRAM HYBMODE(TAPE5,OUTPUT,TAPE6=OUTPUT,TAPE7)                  HYBMODE
      COMMON/B1/ AZ(500),DIP(500),NOBS                                  HYBMODE
      COMMON/B2/ NDENSE,KPCUT,IFLAG(500)                                HYBMODE
      COMMON/B3/ RDIS,ISPHER,LANS                                       HYBMODE
      COMMON/B8/ SL(500),SM(500),SN(500)                                HYBMODE
      COMMON/B21/ R1,R2                                                 HYBMODE
      COMMON/B22/ IOP                                                   HYBMODE
      COMMON/B23/ PCTSURF,RDEG                                          HYBMODE
      COMMON/B24/ MC,VARSUM4                                            HYBMODE
C*****                                                                  HYBMODE
C***** INITAILIZE CONSTANTS AND VARIABLES                               HYBMODE
C*****                                                                  HYBMODE
      CALL INIT                                                         HYBMODE
99997 CALL SECOND(T1)                                                   HYBMODE
C*****                                                                  HYBMODE
C***** READ INPUT DATA                                                  HYBMODE
C*****                                                                  HYBMODE
      CALL INPUT                                                        HYBMODE
C*****                                                                  HYBMODE
C***** CONSTRUCT TEST FOR RANDOMNESS                                    HYBMODE
C*****                                                                  HYBMODE
      CALL POISSON                                                      HYBMODE
C*****                                                                  HYBMODE
C***** SORT DATA INTO PATCHES                                           HYBMODE
C***** DETERMINE DENSE POINTS(MAXIMUM OF ONE PER PATCH)                 HYBMODE
C*****                                                                  HYBMODE
      IF(IOP) 3000,3000,3001                                            HYBMODE
 3000 IF(ISPHER - 1) 2,1,2                                              HYBMODE
    1 K1=1                                                              HYBMODE
      K2=1                                                              HYBMODE
      GO TO 3                                                           HYBMODE
    2 K1=19                                                             HYBMODE
      K2=401                                                            HYBMODE
    3 CALL SORTA(K1,K2)                                                 HYBMODE
      IF(R2-6.0) 49,50,50                                               HYBMODE
   50 CALL XLEVEL(K2)                                                   HYBMODE
      CALL INITR(R2)                                                    HYBMODE
      CALL PCTPLT(K2)                                                   HYBMODE
   49 IF(R1) 1000,1000,999                                              HYBMODE
  999 CALL INITR(R1)                                                    HYBMODE
      CALL PLTPT(AZ,DIP,NOBS)                                           HYBMODE
      IF(ICUT .GE. NOBS) GO TO 1001                                     HYBMODE
      CALL PLTPT1(AZ,DIP,NOBS)                                          HYBMODE
 1001 WRITE(6,900) NOBS                                                 HYBMODE
  900 FORMAT(//1X,*DATA PLOT(POLAR EQUAL AREA PROJECTION).*/            HYBMODE
     1 1X,*THERE ARE*,I5,1X,*POINTS IN THE SAMPLE.*)                    HYBMODE
 1000 WRITE(7,902)                                                      HYBMODE
  902 FORMAT(1H1,*SUMMARY OF RESULTS*//                                 HYBMODE
     1 30X,*RANDOMNESS*,3X,*NO OF*,4X,*OBJECTIVE*,2X,*TIME(SECONDS)*/   HYBMODE
     2 1X,*RDIS  ISPHER  LANS  PCTSURF*,5X,*TEST*,5X,*CLUSTERS*,        HYBMODE
     3     3X,*FUNCTION*,4X,*REQUIRED*///)                              HYBMODE
      CALL DIRCOS(AZ,DIP,NOBS,SL,SM,SN)                                 HYBMODE
 3001 CALL OPTION(K2)                                                   HYBMODE
C***** ALLOCATE EACH REMAINING POINT TO THE DENSE POINT NEAREST IT      HYBMODE
      IF(NDENSE) 99997,99997,91                                         HYBMODE
   91 CALL ALLOC                                                        HYBMODE
C***** CLUSTER THE DENSE POINTS                                         HYBMODE
      CALL DENCLUS                                                      HYBMODE
C***** COMPUTE VALUE OF OBJECTIVE FUNCTION AND OUTPUT CLUSTERS          HYBMODE
```

```
      CALL OUTPUT                                                        HYBMODE
      CALL SECOND(T2)                                                    HYBMODE
      TIME=T2-T1                                                         HYBMODE
      WRITE(6,901) NOBS,TIME                                             HYBMODE
  901 FORMAT(1X,*TIME REQUIRED TO ANALYZE*,I5,1X,*DATA POINTS*,F7.3,1X,  HYBMODE
     1 *SECONDS*)                                                        HYBMODE
      WRITE(7,911) RDEG,ISPHER,LANS,PCTSURF,KPCUT,MC,VARSUM4,TIME        HYBMODE
  911 FORMAT(F5.1,I5,I8,F8.2,I10,I11,E14.3,F10.3)                        HYBMODE
      GO TO 99997                                                        HYBMODE
      END                                                                HYBMODE
      SUBROUTINE CHANGE(M,N)                                             CHANGE
C***** REALLOCATE POINTS BELONGING TO DENSE POINT NUMBER M TO DENSE      CHANGE
C***** POINT NUMBER N                                                    CHANGE
      COMMON/B1/ AZ(500),DIP(500),NOBS                                   CHANGE
      COMMON/B7/ B(500)                                                  CHANGE
      INTEGER B                                                          CHANGE
      DO 1 NN=1,NOBS                                                     CHANGE
      IF(B(NN) .EQ. M) B(NN)=N                                           CHANGE
    1 CONTINUE                                                           CHANGE
      RETURN $ END                                                       CHANGE
      SUBROUTINE ALLOC                                                   ALLOC
C***** ALLOCATE EACH POINT TO A DENSE POINT                              ALLOC
      COMMON/B1/ AZ(500),DIP(500),NOBS                                   ALLOC
      COMMON/B2/ NDENSE,KPCUT,IFLAG(500)                                 ALLOC
      COMMON/B7/ B(500)                                                  ALLOC
      INTEGER B                                                          ALLOC
      DIMENSION DIS(500)                                                 ALLOC
      DO 1 JJ=1,NOBS                                                     ALLOC
      IF(B(JJ)) 1,5,1                                                    ALLOC
    5 DO 2 KK=1,NDENSE                                                   ALLOC
    2 DIS(KK)=DIST1(JJ,KK)                                               ALLOC
      DS=100.                                                            ALLOC
      DO 3 NN=1,NDENSE                                                   ALLOC
      IF(DIS(NN) .GE. DS) GO TO 3                                        ALLOC
    4 DS=DIS(NN)                                                         ALLOC
      MINDEX=NN                                                          ALLOC
    3 CONTINUE                                                           ALLOC
      B(JJ)=MINDEX                                                       ALLOC
    1 CONTINUE                                                           ALLOC
      RETURN $ END                                                       ALLOC
      SUBROUTINE POLARC                                                  POLARC
C  POLARC * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * POLARC
C      DRAWS AND LABELS AXES AND CIRCLE SURROUNDING OUTPUT GRAPH.      * POLARC
C  PLACES AN  *  AT EVERY 10 DEGREES AROUND CIRCLE CIRCUMFERENCE,WITH * POLARC
C  N,E,S,W AT 0,90,180,270 DEGREES RESPECTIVELY.                       * POLARC
C * * CALLS- SPRPAG * * * * * * * * * *                                * POLARC
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * POLARC
      COMMON/B4/ NXMID,NYMID,NX,NY,XCONV,YCONV                           POLARC
      COMMON/B5/ LPAGE(130,78)                                           POLARC
      DIMENSION LDIR(4)                                                  POLARC
      DATA LASTER/1H*/,XINC/10.0/,LDIR/1HN,1HS,1HE,1HW/                  POLARC
      DATA LBLANK/1H /,Q90/90.0/                                         POLARC
      NX1=NX-1                                                           POLARC
      ANGL=0.0                                                           POLARC
      DO 200 I=1,36                                                      POLARC
      CALL SPRPAG(IX,IY,Q90,ANGL)                                        POLARC
       LPAGE(IX,IY)=LASTER                                               POLARC
  200 ANGL=ANGL+10.                                                      POLARC
      LPAGE(NXMID,NYMID)=LASTER                                          POLARC
```

```
C          LABEL N,S,E,W AT APPROPRIATE POSITIONS                       * POLARC
  600 LPAGE(NXMID,NY)=LDIR(1)                                            POLARC
      LPAGE(NXMID,1)=LDIR(2)                                             POLARC
      LPAGE(NX,NYMID)=LDIR(3)                                            POLARC
      LPAGE(1,NYMID)=LDIR(4)                                             POLARC
C          PRINT THE COMPLETE ASSEMBLED PAGE IMAGE,ROW BY ROW           * POLARC
C          **NOTE** THE INDEX K MUST GO BACKWARDS BECAUSE THE ASSUMED   * POLARC
C          COORDINATE ORIGIN IS AT LOWER LEFT HAND CORNER OF PAGE.      * POLARC
      DO 700 J=1,NY                                                      POLARC
      K=NY-J+1                                                           POLARC
700   WRITE(6,2000) (LPAGE(I,K),I=1,NX)                                  POLARC
 2000 FORMAT(121A1)                                                      POLARC
      DO 199 J1=1,73                                                     POLARC
      DO 199 J2=1,121                                                    POLARC
199   LPAGE(J2,J1)=LBLANK                                                POLARC
      RETURN                                                             POLARC
      END                                                                POLARC
      SUBROUTINE SPRPAG(IX,IY,DIN,AZN)                                   SPRPAG
C  SPRPAG * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * SPRPAG
C     GIVEN  HEMISPHERE COORDINATES (DIN,AZN),SPRPAG RETURNS NEAREST    * SPRPAG
C  PAGE COORDINATES (IX,IY). IF OUTSIDE PAGE,MOVES IX OR IY TO          * SPRPAG
C  CLOSEST POINT ON EDGE OF PAGE                                        * SPRPAG
      COMMON/B4/ NXMID,NYMID,NX,NY,XCONV,YCONV                           SPRPAG
      COMMON/B12/ DEGRAD,RADEG                                           SPRPAG
      ANGLR=AZN*DEGRAD                                                   SPRPAG
C          THE NEXT 3 STATEMENTS PERFORM EQUAL-AREA PROJECTION          * SPRPAG
  100 XYT=127.2798*SIN(DEGRAD*DIN/2.0)                                   SPRPAG
      NDUM=XYT* SIN(ANGLR)*XCONV+0.5                                     SPRPAG
      JDUM=XYT*COS(ANGLR)*YCONV +0.5                                     SPRPAG
  200 IX=NXMID+NDUM                                                      SPRPAG
C          BRING X INDEX IN RANGE IF IT IS OUTSIDE.                     * SPRPAG
      IF(IX.LT.1) IX=1                                                   SPRPAG
      IF(IX.GT.NX) IX=NX                                                 SPRPAG
      IY=NYMID+JDUM                                                      SPRPAG
C          BRING Y INDEX IN RANGE IF IT IS OUTSIDE.                     * SPRPAG
      IF(IY.LT.1) IY=1                                                   SPRPAG
      IF(IY.GT.NY) IY=NY                                                 SPRPAG
      RETURN                                                             SPRPAG
      END                                                                SPRPAG
      FUNCTION DIST(I,J)                                                 DIST
C***** DIST = DISTANCE FROM PT NUMBER I TO PT NUMBER J                   DIST
      COMMON/B8/ SL(500),SM(500),SN(500)                                 DIST
      DIST=(SL(I)-SL(J))**2 + (SM(I)-SM(J))**2 + (SN(I)-SN(J))**2        DIST
      RETURN $ END                                                       DIST
      FUNCTION DIST2(K,J)                                                DIST2
C***** DIST2 = DISTANCE FROM DENSE POINT NUMBER K TO DENSE POINT          DIST2
C***** NUMBER J                                                          DIST2
      COMMON/B6/ SLD(500),SMD(500),SND(500)                              DIST2
      DIST2=(SLD(K)-SLD(J))**2+(SMD(K)-SMD(J))**2+(SND(K)-SND(J))**2     DIST2
      RETURN $ END                                                       DIST2
      FUNCTION DIST1(KK,JJ)                                              DIST1
C***** DIST1 = DISTANCE FROM POINT NUMBER KK TO DENSE POINT NUMBER        DIST1
C***** JJ                                                                DIST1
      COMMON/B6/ SLD(500),SMD(500),SND(500)                              DIST1
      COMMON/B8/ SL(500),SM(500),SN(500)                                 DIST1
      DIST1=(SL(KK)-SLD(JJ))**2+(SM(KK)-SMD(JJ))**2+(SN(KK)-SND(JJ))**2 DIST1
      RETURN $ END                                                       DIST1
      SUBROUTINE SORTA(KK1,LL3)                                          SORTA
C***** SORTS DATA INTO PATCHES. IPATCH(J) IS THE LARGEST INDEX I         SORTA
```

```
C***** SUCH THAT (A(I),D(I)) IS IN PATCH NUMBER J. IF PATCH NUMBER
C***** J IS EMPTY, IPATCH(J) = IPATCH(J-1). IF(ISPHER = 1, THEN
C***** KK1=LL3=1. IF ISPHER NE 1, KK1=19, LL3=401, AND
C***** BANDS 19 THROUGH 36 AND CELLS 401 THROUGH 800 CONTAIN ALL
C***** THE DATA. IF ISPHER = 1 THEN THERE ARE DATA POINTS WITH
C***** DIP GREATER THAN 90 DEGREES IN WHICH CASE BANDS 1 THROUGH
C***** 19 AND PATCHES 1 THROUGH 400 CONTAIN DATA ALSO.
      COMMON/B1/ A(500),D(500),NOBS
      COMMON/B9/ IPATCH(800),KBAND(800)
      COMMON/B11/ IBND(36)
      COMMON/B16/ DANG(37)
      DATA NB/36/
      REAL KANG
      DO 9 I=1,800
    9 IPATCH(I)=0

C***** ARRANGE DATA IN DECREASING ORDER OF DIP

      NOBS1=NOBS-1
      CALL SORT(D,A,NOBS)
C

C***** ARRANGE THE POINTS IN EACH BAND IN INCREASING ORDER OF AZIMUTH

      M=1
      DO 93 L=KK1,NB
      IF(M .EQ. NOBS) GO TO 8888
   94 CONTINUE
      DO 92 J=M,NOBS1
      J1=J+1
      IF(D(J) .GE. DANG(L+1)) GO TO 8884
 8885 M=J
      GO TO 93
 8884 DO 90 K=J1,NOBS
      IF(D(K) .GE. DANG(L+1)) GO TO 8886
 8887 M=J+1
      GO TO 94
 8886 IF(A(J)-A(K)) 90,90,91
   91 TMP=A(J)
      A(J)=A(K)
      A(K)=TMP
      TMP=D(J)
      D(J)=D(K)
      D(K)=TMP
   90 CONTINUE
      M=M+1
   92 CONTINUE
   93 CONTINUE
 8888 CONTINUE

C***** DETERMINE WHICH PATCHES CONTAIN DATA

      II=LL3-1
      LP=0
      KK=0
      ICP=1
      DO 87 J=KK1,NB
      II=II+KK
      IBJ=IBND(J)
```

```
      DO 87 K=1,IBJ
      IF(LP .EQ. NOBS) GO TO 88
      KK=K
      KANG=FLOAT(K)*360./FLOAT(IBND(J))
      IF(K .EQ. IBND(J)) KANG=360.5
      DO 86 L=IOP,NOBS
      IF(A(L) .GT. KANG .OR. D(L) .LT. DANG(J+1)) GO TO 79
      LP=LP+1
      IF(LP .NE. NOBS) GO TO 86
   79 IPATCH(II+K)=LP
      IOP=LP+1
      L=II+K+1
      GO TO 87
86    CONTINUE
87    CONTINUE
   88 IF(L .EQ. 801) RETURN
      DO 89 J=L,800
   89 IPATCH(J)=NOBS
      RETURN $ END
      SUBROUTINE OUTPUT
C*****
C***** PRINTS CONTENTS OF CLUSTERS AND EVALUATES OBJECTIVE FUNCTION
C*****
      COMMON/B2/ NDENSE,KPCUT,IFLAG(500)
      COMMON/B3/ RDIS,ISPHER,LANS
      COMMON/B13/ KOU
      COMMON/B14/ CU1(500),CU2(500)
      COMMON/B15/ SLL(500),SMM(500),SNN(500)
      COMMON/B19/ SX,SY,SZ
      COMMON/B21/ R1,R2
      COMMON/B23/ PCTSURF,RDEG
      COMMON/B24/ MC,VARSUM4
      COMMON/B26/ ICUT
      DIMENSION CGX(50),CGY(50),CGZ(50)
      DATA IPLUS/1H+/
      MC=0
      VAR=0.0
      VARA=0.0
      WRITE(6,907)
  907 FORMAT(1H1,*OUTPUT OF CLUSTERS, CLUSTER MEANS*)
      DO 30 MJ=1,NDENSE
      IF(IFLAG(MJ) .NE. MJ) GO TO 30
      VAR1=0.0
      MC=MC+1
      IF(MC-1) 1001,1000,1001
 1000 WRITE(6,908) MC
  908 FORMAT(//* CLUSTER NUMBER*,I5///)
      GO TO 1002
 1001 WRITE(6,910) MC
  910 FORMAT(1H1*CLUSTER NUMBER*,I5///)
 1002 CONTINUE
C***** PUT POINTS BELONGING TO DENSE POINT NUMBER MJ INTO ARRAYS
C***** CU1 AND CU2(AZIMUTH OF DIP AND DIP RESPECTIVELY). KOU IS
C***** RETURNED TO THE CALLING PROGRAM AND IS THE NUMBER OF SUCH
C***** POINTS.
      CALL BPLTPT(MJ,KOU)
      XKOU=FLOAT(KOU)
      WRITE(6,900) (KK,CU1(KK),CU2(KK),KK=1,KOU)
  900 FORMAT(5X,*NUM*,5X,*AZIMUTH  *,*INCLINATION*,/(I8,2F12.2))
```

```
C***** COMPUTE DIRECTION COSINES                                         OUTPUT
   61 CALL DIRCOS(CU1,CU2,KOU,SLL,SMM,SNN)                               OUTPUT
C***** CALCULATE RESULTANT BY ADDING THE DIRECTION COSINES               OUTPUT
      CALL SUMCOS                                                        OUTPUT
C***** DETERMINE SPHERICAL COORDINATES FOR RESULTANT AND MEAN            OUTPUT
      CALL SPHRPT(SX,SY,SZ,DP,AT,R)                                      OUTPUT
      WRITE(6,48) AT,DP                                                  OUTPUT
   48 FORMAT(///1X,*MEAN*/1X,*AZIMUTH*,F7.2,1X,*DEG*/1X,*INCLINATION*,   OUTPUT
     1 F7.2,1X,*DEG*)                                                    OUTPUT
      WRITE(6,906) R,SX,SY,SZ                                            OUTPUT
  906 FORMAT(/1X,*RESULTANT AND ITS COMPONENTS - R,SX,SY,SZ.*/ 4F12.5)   OUTPUT
      SUM2=0.0                                                           OUTPUT
      AL1=CGX(MC)=SX/XKOU                                                OUTPUT
      AM1=CGY(MC)=SY/XKOU                                                OUTPUT
      AN1=CGZ(MC)=SZ/XKOU                                                OUTPUT
      WRITE(6,911) AL1,AM1,AN1                                           OUTPUT
  911 FORMAT(//1X,*CENTER OF GRAVITY - (X,Y,Z) COORDINATES*/             OUTPUT
     1 3F10.4)                                                           OUTPUT
      DO 10 LJ=1,KOU                                                     OUTPUT
      DIS=(SQRT((AL1-SLL(LJ))**2 + (AM1-SMM(LJ))**2 + (AN1-SNN(LJ))**2))OUTPUT
     1**2                                                                OUTPUT
      SUM2=SUM2+DIS                                                      OUTPUT
   10 CONTINUE                                                           OUTPUT
      VARA=VARA+SUM2                                                     OUTPUT
      IF(R1) 31,31,2001                                                  OUTPUT
 2001 CALL PLTPT(CU1,CU2,KOU)                                            OUTPUT
      IF(ICUT .GE. KOU) GO TO 31                                         OUTPUT
      IF(ICUT) 2002,2002,2003                                            OUTPUT
 2003 WRITE(6,912) MC,KOU,RDEG                                           OUTPUT
 2002 CALL PLTPT1(CU1,CU2,KOU)                                           OUTPUT
   31 WRITE(6,912) MC,KOU,RDEG                                           OUTPUT
   30 CONTINUE                                                           OUTPUT
      VARSUM2=0.0                                                        OUTPUT
  912 FORMAT(//1X,*CLUSTER NO.*,I5,1X,                                   OUTPUT
     1 *CONTAINS*,I5,1X,*POINTS.*/1X,                                    OUTPUT
     2 *FOR THIS ANALYSIS, RDIS =*,F7.2)                                 OUTPUT
      VARSUM2=0.0                                                        OUTPUT
      IF(MC .EQ. 1) GO TO 50                                             OUTPUT
      MC1=MC-1                                                           OUTPUT
      DO 40 LJ=1,MC1                                                     OUTPUT
      LJ1=LJ+1                                                           OUTPUT
      DO 40 MJ=LJ1,MC                                                    OUTPUT
      DIS=(CGX(LJ)-CGX(MJ))**2 + (CGY(LJ)-CGY(MJ))**2 +                  OUTPUT
     1 (CGZ(LJ)-CGZ(MJ))**2                                              OUTPUT
      VARSUM2=VARSUM2+DIS                                                OUTPUT
   40 CONTINUE                                                           OUTPUT
   50 VARSUM4=VARSUM2+VARA                                               OUTPUT
      WRITE(6,905) VARSUM4                                               OUTPUT
  905 FORMAT(//1X,*OBJECTIVE FUNCTION =*,F12.3)                          OUTPUT
      RETURN $ END                                                       OUTPUT
      SUBROUTINE INPUT                                                   INPUT
C***** READ INPUT DATA                                                   INPUT
      COMMON/B1/ AZ(500),DIP(500),NOBS                                   INPUT
      COMMON/B3/ RDIS,ISPHER,LANS                                        INPUT
      COMMON/B21/ R1,R2                                                  INPUT
      COMMON/B22/ IOP                                                    INPUT
      DIMENSION HED(8)                                                   INPUT
      READ(5,900) HED                                                    INPUT
  900 FORMAT(8A10)                                                       INPUT
```

```
      IF(EOF,5) 1000,1001                                              INPUT
 1001 LL=1                                                             INPUT
      IF(HED(1) .EQ. 10HCONTINUE    ) LL=2                             INPUT
      GO TO (7002,7000),LL                                             INPUT
 7002 READ(5,906) NOBS,LANS,ISPHER,RDIS,R1,R2                          INPUT
    1 ,(AZ(I),DIP(I),I=1,NOBS)                                         INPUT
  906 FORMAT(3I5,3F12.5/(12F5.0))                                      INPUT
      WRITE(6,907) HED,NOBS,LANS,ISPHER,RDIS,R1,R2                     INPUT
    1 ,(AZ(I),DIP(I),I=1,NOBS)                                         INPUT
  907 FORMAT(1H1/8A10//                                                INPUT
    1 1X,*NOBS =*,I5//                                                 INPUT
    2 1X,*LANS =*,I5//                                                 INPUT
    3 1X,*ISPHER =*,I5//                                               INPUT
    4 1X,*RDIS =*,F6.1//                                               INPUT
    5 1X,*RADIUS OF POINT PLOT =*,F6.1//                               INPUT
    6 1X,*RADIUS OF DENSITY PLOT =*,F6.1//                             INPUT
    7 1X,*AZIMUTH OF THE DIP AND DIP OF THE DATA POINTS*//             INPUT
    8 (12F5.0))                                                        INPUT
      IOP=0                                                            INPUT
      RETURN                                                           INPUT
 7000 READ(5,908) LANS,RDIS                                            INPUT
  908 FORMAT(I5,F12.5)                                                 INPUT
      WRITE(6,909) LANS,RDIS                                           INPUT
  909 FORMAT(1H1,*FOR NEXT ANALYSIS OF SAMPLE*/                        INPUT
    1 1X,*LANS =*,I5//                                                 INPUT
    2 1X,*RDIS =*,F6.1//)                                              INPUT
      CALL INITI(4.0)                                                  INPUT
      IOP=1                                                            INPUT
      GO TO 1002                                                       INPUT
 1000 STOP                                                             INPUT
 1002 CONTINUE                                                         INPUT
      RETURN $ END                                                     INPUT
      SUBROUTINE POISSON                                               POISSON
C***** CONSTRUCT TEST FOR RANDOMNESS - SPHERICAL COUNTING CELL MUST     POISSON
C***** CONTAIN AT LEAST KPCUT POINTS FOR A SIGNIFICANT(95)             POISSON
C***** CONCENTRATION TO BE PRESENT                                     POISSON
      COMMON/B1/ AZ(500),DIP(500),NOBS                                 POISSON
      COMMON/B2/ NDENSE,KPCUT,IFLAG(500)                               POISSON
      COMMON/B3/ RDIS,ISPHER,LANS                                      POISSON
      COMMON/B12/ DEGRAD,RADEG                                         POISSON
      COMMON/B23/ P,RDEG                                               POISSON
      RDEG=RDIS*DEGRAD                                                 POISSON
      FN=FLOAT(NOBS)                                                   POISSON
      PCTSURF=1.0-COS(RDEG)                                            POISSON
      IF(ISPHER .EQ. 1) PCTSURF=0.5*PCTSURF                            POISSON
      A=FN*PCTSURF                                                     POISSON
      P=100.0*PCTSURF                                                  POISSON
      WRITE(6,910) RDIS,P                                              POISSON
  910 FORMAT(1X,*FOR RDIS =*,F6.2,1X,*PERCENTAGE OF SURFACE AREA IS*,  POISSON
    1 F6.2)                                                            POISSON
      RDIS=4.0*(SIN(RDEG/2.0))**2                                      POISSON
      RDEG=RDEG*RADEG                                                  POISSON
      EFAC=EXP(-A)                                                     POISSON
      SUM=EFAC                                                         POISSON
      FAC=1.0                                                          POISSON
      DO 876 N=1,1000                                                  POISSON
      FIN=N                                                            POISSON
      FAC=FAC*A/FIN                                                    POISSON
      PROB=EFAC*FAC                                                    POISSON
```

```
      SUM=SUM+PROB                                                      POISSON
      TEST=1.0-SUM                                                      POISSON
      IF(TEST - 0.05) 875,876,876                                       POISSON
  876 CONTINUE                                                          POISSON
  875 KPCUT=FIN                                                         POISSON
      WRITE(6,901) KPCUT,RDIS                                           POISSON
  901 FORMAT(1X,*A POINT X IS DENSE IF THERE ARE*,I5,1X,*OBSERVATIONS*/ POISSON
     1 * WHOSE EUCLIDEAN DISTANCE FROM X IS LESS THAN*,E10.2,*.*/       POISSON
     2)                                                                 POISSON
      RETURN                                                            POISSON
      END                                                               POISSON
      SUBROUTINE OPTION(KK1)                                            OPTION
C***** BGINNING WITH PATCH NUMBER KK1, EXAMINE EACH POINT TO SEE        OPTION
C***** IF IT IS  DENSE                                                  OPTION
      COMMON/B1/ AZ(500),DIP(500),NOBS                                  OPTION
      COMMON/B2/ NDENSE,KPCUT,IFLAG(500)                                OPTION
      COMMON/B3/ RDIS,ISPHER,LANS                                       OPTION
      COMMON/B6/ SLD(500),SMD(500),SND(500)                             OPTION
      COMMON/B7/ B(500)                                                 OPTION
      COMMON/B8/ SL(500),SM(500),SN(500)                                OPTION
      COMMON/B9/ LPATCH(800),KBAND(800)                                 OPTION
      COMMON/B11/ IBND(36)                                              OPTION
      COMMON/B12/ DEGRAD,RADEG                                          OPTION
      COMMON/B14/ CU1(500),CU2(500)                                     OPTION
      COMMON/B21/ R1,R2                                                 OPTION
      COMMON/B26/ ICUT                                                  OPTION
      INTEGER B                                                         OPTION
      DATA IPLUS/1H+/                                                   OPTION
      KOUNT=1                                                           OPTION
      KS=1                                                              OPTION
      LANS1=LANS-1                                                      OPTION
      IPRE=0                                                            OPTION
      DO 10 KK=KK1,800                                                  OPTION
      N1=LPATCH(KK)                                                     OPTION
      IF(N1 - IPRE) 104,104,100                                         OPTION
  100 DO 11 JJ=KS,N1                                                    OPTION
      ISUM=0                                                            OPTION
      DO 12 MM=1,NOBS                                                   OPTION
      FD=DIST(JJ,MM)                                                    OPTION
      IF(LANS1) 7,9,7                                                   OPTION
    9 FD1=DIST3(JJ,MM)                                                  OPTION
      IF(FD1-RDIS) 6,6,7                                                OPTION
    7 IF(FD-RDIS) 6,6,12                                                OPTION
    6 ISUM=ISUM+1                                                       OPTION
      IF(ISUM-KPCUT)12,8,8                                              OPTION
    8 CONTINUE                                                          OPTION
C***** IF PATCH NUMBER KK CONTAINS A DENSE POINT, STORE THE             OPTION
C***** DIRECTION COSINES OF THE POINT IN (SLD,SMD,SND)                  OPTION
      SLD(KOUNT)=SL(JJ)                                                 OPTION
      SMD(KOUNT)=SM(JJ)                                                 OPTION
      SND(KOUNT)=SN(JJ)                                                 OPTION
      CU1(KOUNT)=AZ(JJ)                                                 OPTION
      CU2(KOUNT)=DIP(JJ)                                                OPTION
C***** ALLOCATE ALL OTHER POINTS IN THE PATCH TO THE DENSE POINT        OPTION
      DO 1001 JM=KS,N1                                                  OPTION
 1001 B(JM)=KOUNT                                                       OPTION
      KOUNT=KOUNT+1                                                     OPTION
      GO TO 104                                                         OPTION
   12 CONTINUE                                                          OPTION
```

```
   11 CONTINUE                                                          OPTION
  104 CONTINUE                                                          OPTION
      KS=LPATCH(KK)+1                                                   OPTION
      IPRE=N1                                                           OPTION
   10 CONTINUE                                                          OPTION
      NDENSE=KOUNT-1                                                    OPTION
      IF(NDENSE) 90,90,91                                               OPTION
C***** NDENSE IS THE NUMBER OF DENSE POINTS                             OPTION
   91 IF(R1) 90,90,1999                                                 OPTION
 1999 CALL PLTPT(CU1,CU2,NDENSE)                                        OPTION
      IF(ICUT .GE. NDENSE) GO TO 90                                     OPTION
      IF(ICUT) 92,92,93                                                 OPTION
   93 WRITE(6,900) NDENSE                                               OPTION
   92 CALL PLTPT1(CU1,CU2,NDENSE)                                       OPTION
   90 WRITE(6,900) NDENSE                                               OPTION
  900 FORMAT(//1X,*THERE ARE*,I5,1X,*DENSE POINTS.*)                    OPTION
      RETURN $ END                                                      OPTION
      SUBROUTINE BPLTPT(NN,K)                                           BPLTPT
C***** POINTS BELONGING TO DENSE POINT NN ARE PUT INTO ARRAYS CU1   CU2 BPLTPT
      COMMON/B1/ AZ(500),DIP(500),NOBS                                  BPLTPT
      COMMON/B7/ B(500)                                                 BPLTPT
      COMMON/B14/ CU1(500),CU2(500)                                     BPLTPT
      INTEGER B                                                         BPLTPT
      MM=NN                                                             BPLTPT
      KOUNT=1                                                           BPLTPT
      DO 3 M=1,NOBS                                                     BPLTPT
      IF(B(M)-MM) 3,4,3                                                 BPLTPT
    4 CU1(KOUNT)=AZ(M)                                                  BPLTPT
      CU2(KOUNT)=DIP(M)                                                 BPLTPT
      KOUNT=KOUNT+1                                                     BPLTPT
    3 CONTINUE                                                          BPLTPT
      K=KOUNT-1                                                         BPLTPT
      RETURN                                                            BPLTPT
      END                                                               BPLTPT
      SUBROUTINE PCTPLT(KK1)                                            PCTPLT
C***** PLOTS RLEVEL(J)(DENSITY OF PATCH J, I.E., PER CENT OF TOTAL SAMPLPCTPLT
C***** CONTAINED IN PATCH J) AT THE PROJECTED MIDPOINT OF PATCH         PCTPLT
C***** NUMBER J(POLAR EQUAL AREA PROJECTION).                           PCTPLT
      COMMON/B5/ LPAGE(130,78)                                          PCTPLT
      COMMON/B9/ LPATCH(800),KBAND(800)                                 PCTPLT
      COMMON/B10/ POS(800)                                              PCTPLT
      COMMON/B11/ IBND(36)                                              PCTPLT
      COMMON/B16/ DANG(37)                                              PCTPLT
      COMMON/B20/ RLEVEL(800)                                           PCTPLT
      COMMON/B21/ R1,R2                                                 PCTPLT
      DIMENSION NUM1(10),NUM(10)                                        PCTPLT
      INTEGER O                                                         PCTPLT
      WRITE(6,905)                                                      PCTPLT
  905 FORMAT(1H1)                                                       PCTPLT
      DATA NUM1/1H ,1H1,1H2,1H3,1H4,1H5,1H6,1H7,1H8,1H9/,               PCTPLT
     1       NUM/1H0,1H1,1H2,1H3,1H4,1H5,1H6,1H7,1H8,1H9/,LDOT/1H./     PCTPLT
      WRITE(6,900)                                                      PCTPLT
  900 FORMAT(1HQ,*DENSITY PLOT - LOWER AND UPPER HEMISPHERES*//)        PCTPLT
      DO 1 J=KK1,800                                                    PCTPLT
      IBJ=KBAND(J)                                                      PCTPLT
      AZN=(POS(J)-0.5)*360./FLOAT(IBND(IBJ))                            PCTPLT
      IBJ1=IBJ+1                                                        PCTPLT
      DIN=(DANG(IBJ)+DANG(IBJ1))/2.                                     PCTPLT
      IF(DIN .GT. 90.0) DIN=180.0-DIN                                   PCTPLT
```

```
      CALL SPRPAG(IX,IY,DIN,AZN)                                        PCTPLT
      IF(DIN .LT. 81.0) GO TO 3033                                      PCTPLT
      IF(AZN .LT. 180.) IX=IX-2                                         PCTPLT
      IF(AZN .GT. 180.) IX=IX+2                                         PCTPLT
 3033 CONTINUE                                                          PCTPLT
      IF(J .EQ. 1 .OR. J .EQ. 800) IY=IY-1                              PCTPLT
      K=RLEVEL(J)*10000.+5.                                             PCTPLT
      M=K/1000                                                          PCTPLT
      N=(K-(M*1000))/100                                                PCTPLT
      O=(K-M*1000-N*100)/10                                             PCTPLT
      LPAGE(IX-2,IY)=NUM1(M+1)                                          PCTPLT
      LPAGE(IX-1,IY)=NUM(N+1)                                           PCTPLT
      IF(M .EQ. 0) LPAGE(IX-1,IY)=NUM1(N+1)                             PCTPLT
      LPAGE(IX,IY)=LDOT                                                 PCTPLT
      LPAGE(IX+1,IY)=NUM(O+1)                                           PCTPLT
      IF(M .EQ. 0 .AND. N .EQ. O .AND. O .EQ. 0) GO TO 2                PCTPLT
      GO TO 5                                                           PCTPLT
    2 LPAGE(IX-2,IY)=NUM1(1)                                            PCTPLT
      LPAGE(IX-1,IY)=NUM1(1)                                            PCTPLT
      LPAGE(IX+1,IY)=NUM1(1)                                            PCTPLT
    5 IF(J .NE. 400 .AND. J .NE. 800) GO TO 1                           PCTPLT
      IF(J .EQ. 800) GO TO 6                                            PCTPLT
      WRITE(6,901)                                                      PCTPLT
  901 FORMAT(1X,*LOWER HEMISPHERE*//)                                   PCTPLT
      CALL POLARC                                                       PCTPLT
      GO TO 7                                                           PCTPLT
    6 WRITE(6,902)                                                      PCTPLT
  902 FORMAT(1X,*UPPER HEMISPHERE*//)                                   PCTPLT
      CALL POLARC                                                       PCTPLT
    7 WRITE(6,903)                                                      PCTPLT
  903 FORMAT(1HR)                                                       PCTPLT
      WRITE(6,905)                                                      PCTPLT
      WRITE(6,904)                                                      PCTPLT
  904 FORMAT(1HQ)                                                       PCTPLT
    1 CONTINUE                                                          PCTPLT
      WRITE(6,903)                                                      PCTPLT
      RETURN                                                            PCTPLT
      END                                                               PCTPLT
      SUBROUTINE DIRCOS(A,D,N,SL,SM,SN)                                 DIRCOS
C***** GIVEN THE SPHERICAL COORDINATES (1,A(I),D(I),I=1,N),             DIRCOS
C***** RETURN THE CORRESPONDING (X,Y,Z) COORDINATES (SL(I),SM(I),       DIRCOS
C***** SN(I),I=1,N)                                                     DIRCOS
      COMMON/B12/ DEGRAD,RADEG                                          DIRCOS
      DIMENSION A(N),D(N),SL(N),SM(N),SN(N)                             DIRCOS
      DO 1 JJ=1,N                                                       DIRCOS
      DD=D(JJ)*DEGRAD                                                   DIRCOS
      AA=A(JJ)*DEGRAD                                                   DIRCOS
      SL(JJ)=SIN(DD)*COS(AA)                                            DIRCOS
      SM(JJ)=SIN(DD)*SIN(AA)                                            DIRCOS
      SN(JJ)=COS(DD)                                                    DIRCOS
    1 CONTINUE                                                          DIRCOS
      RETURN                                                            DIRCOS
      END                                                               DIRCOS
      SUBROUTINE SPHRPT(X,Y,Z,D,A,R)                                    SPHRPT
C***** GIVEN THE (X,Y,Z) COORDINATES OF A POINT, RETURN ITS             SPHRPT
C***** SPHERICAL COORDINATES (R,D,A)                                    SPHRPT
      COMMON/B12/ DEGRAD,RADEG                                          SPHRPT
      DATA PI2/6.2831853071795864769/                                   SPHRPT
      R=SQRT(X*X + Y*Y + Z*Z)                                           SPHRPT
```

```
      IF(X .EQ. 0.0 .AND. Y .EQ. 0.0) GO TO 9395                        SPHRPT
      IF(Y) 9397,9396,9397                                              SPHRPT
 9397 IF(X) 9398,9393,9398                                              SPHRPT
 9398 A=ATAN2(Y,X)                                                      SPHRPT
      IF(A .LT. 0.0) A=A+PI2                                            SPHRPT
      A=A*RADEG                                                         SPHRPT
      GO TO 9394                                                        SPHRPT
 9395 D=A=0.0                                                           SPHRPT
      RETURN                                                            SPHRPT
 9396 A=ABS((X/ABS(X)-1.0)*90.0)                                        SPHRPT
      GO TO 9394                                                        SPHRPT
 9393 A=ABS((Y/ABS(Y)-2.0)*90.0)                                        SPHRPT
 9394 D=ACOS(Z/R)*RADEG                                                 SPHRPT
      RETURN                                                            SPHRPT
      END                                                               SPHRPT
      SUBROUTINE DENCLUS                                                DENCLUS
C***** CLUSTER THE DENSE POINTS AS FOLLOWS - TWO DENSE POINTS           DENCLUS
C***** BELONG TO THE SAME CLUSTER IF THE DISTANCE BETWEEN THEM          DENCLUS
C***** IS LESS THAN RDIS DEGREES                                        DENCLUS
      COMMON/B2/ NDENSE,KPCUT,IFLAG(500)                                DENCLUS
      COMMON/B3/ RDIS,ISPHER,LANS                                       DENCLUS
      NDENSE1=NDENSE-1                                                  DENCLUS
      DO 6 K=1,NDENSE1                                                  DENCLUS
      K1=K+1                                                            DENCLUS
      DO 6 J=K1,NDENSE                                                  DENCLUS
      FD=DIST2(K,J)                                                     DENCLUS
  301 IF(FD .GE. RDIS .OR. IFLAG(K) .EQ. IFLAG(J)) GO TO 6              DENCLUS
      CALL CHANGE(IFLAG(J),IFLAG(K))                                    DENCLUS
      DO 7 NN=1,NDENSE                                                  DENCLUS
      IF(IFLAG(NN) .EQ. IFLAG(J)) IFLAG(NN)=IFLAG(K)                    DENCLUS
    7 CONTINUE                                                          DENCLUS
    6 CONTINUE                                                          DENCLUS
      RETURN                                                            DENCLUS
      END                                                               DENCLUS
      SUBROUTINE PLTPT(A,D,N)                                           PLTPT
C***** PLOT THE ARRAY (A(I),D(I),I=1,N) (POLAR EQUAL AREA               PLTPT
C***** PROJECTION                                                       PLTPT
      COMMON/B5/ LPAGE(130,78)                                          PLTPT
      COMMON/B26/ ICUT                                                  PLTPT
      DIMENSION A(N),D(N)                                               PLTPT
      DATA IPLUS/1H+/                                                   PLTPT
      ICUT=0                                                            PLTPT
      DO 777 KK=1,N                                                     PLTPT
      IF(D(KK) - 90.0) 778,778,776                                      PLTPT
  776 D180=180.0-D(KK)                                                  PLTPT
      CALL SPRPAG(IX,IY,D180,A(KK))                                     PLTPT
      LPAGE(IX,IY)=1H+                                                  PLTPT
      ICUT=KK                                                           PLTPT
  777 CONTINUE                                                          PLTPT
  778 IF(ICUT .EQ. 0) GO TO 783                                         PLTPT
      WRITE(6,781)                                                      PLTPT
  781 FORMAT(1H1,*LOWER HEMISPHERE*//)                                  PLTPT
      CALL POLARC                                                       PLTPT
  783 RETURN                                                            PLTPT
      ENTRY PLTPT1                                                      PLTPT
  780 ICUT1=ICUT+1                                                      PLTPT
      DO 779 KK=ICUT1,N                                                 PLTPT
      CALL SPRPAG(IX,IY,D(KK),A(KK))                                    PLTPT
  779 LPAGE(IX,IY)=1H+                                                  PLTPT
```

```
      WRITE(6,782)                                                      PLTPT
  782 FORMAT(1H1,*UPPER HEMISPHERE*//)                                  PLTPT
      CALL POLARC                                                       PLTPT
      RETURN                                                            PLTPT
      END                                                               PLTPT
      SUBROUTINE INIT(RADIUS)                                           INIT
C***** INITIALIZE CONSTANTS AND VARIABLES                               INIT
      COMMON/B1/ AZ(500),DIP(500),NOBS                                  INIT
      COMMON/B2/ NDENSE,KPCUT,IFLAG(500)                                INIT
      COMMON/B4/ NXMID,NYMID,NX,NY,XCONV,YCONV                          INIT
      COMMON/B7/ B(500)                                                 INIT
      COMMON/B9/ LPATCH(800),KBAND(800)                                 INIT
      COMMON/B10/ POS(800)                                              INIT
      COMMON/B11/ IBND(36)                                              INIT
      COMMON/B12/ DEGRAD,RADEG                                          INIT
      COMMON/B16/ DANG(37)                                              INIT
      DATA G90/90.0/                                                    INIT
      DATA DEGRAD/0.0174532925/,RADEG/57.2957795131/                    INIT
      DATA DANG/180.0,175.9,171.9,164.8,160.1,154.2,149.3,              INIT
     1 144.1,139.5,134.0,129.1,124.1,119.3,114.5,109.9,105.1,           INIT
     2 100.4,95.2,90.0,84.8,79.6,74.9,70.1,65.5,60.7,55.9,              INIT
     3 50.9,46.0,40.5,35.9,30.7,25.8,19.9,15.2,8.1,4.1,0.0/,            INIT
     4 IBND/1,3,10,10,16,16,20,20,26,26,28,28,30,30,32,32,              INIT
     5 36,36,36,36,32,32,30,30,28,28,26,26,20,20,16,16,                 INIT
     6 10,10,3,1/,                                                      INIT
     7 KBAND/1,3*2,10*3,10*4,16*5,16*6,20*7,20*8,26*9,26*10,            INIT
     8 28*11,28*12,30*13,30*14,32*15,32*16,36*17,36*18,36*19,           INIT
     9 36*20,32*21,32*22,30*23,30*24,28*25,28*26,26*27,26*28,           INIT
     1 20*29,20*30,16*31,16*32,10*33,10*34,3*35,1*36/                   INIT
      INTEGER B                                                         INIT
      M=0                                                               INIT
      DO 3 J=1,36                                                       INIT
      IBJ=IBND(J)                                                       INIT
      DO 4 K=1,IBJ                                                      INIT
      M=M+1                                                             INIT
      POS(M)=K                                                          INIT
    4 CONTINUE                                                          INIT
    3 CONTINUE                                                          INIT
      ENTRY INITI                                                       INIT
      DO 1 J=1,NOBS                                                     INIT
    1 B(J)=0                                                            INIT
      DO 2 J=1,500                                                      INIT
    2 IFLAG(J)=J                                                        INIT
      RETURN                                                            INIT
      ENTRY INITR                                                       INIT
      IF(RADIUS .LE. 0.0) RETURN                                        INIT
      XPINCH=10.0                                                       INIT
      YPINCH=6.0                                                        INIT
      NT=RADIUS*XPINCH+0.5                                              INIT
      NX=2*NT+1                                                         INIT
      NT=RADIUS*YPINCH+0.5                                              INIT
      NY=2*NT+1                                                         INIT
      NX1=NX-1                                                          INIT
      NY1=NY-1                                                          INIT
      NYMID=NY/2+1                                                      INIT
      NXMID=NX/2+1                                                      INIT
      NXMID1=NXMID-1                                                    INIT
      NYMID1=NYMID-1                                                    INIT
      XCON=G90/NXMID1                                                   INIT
```

```
      XCONV=1.0/XCON                                                    INIT
      YCON=Q90/NYMID1                                                   INIT
      YCONV=1.0/YCON                                                    INIT
      RETURN                                                            INIT
      END                                                               INIT
      SUBROUTINE SUMCOS                                                 SUMCOS
C***** COMPUTE (X,Y,Z) COORDINATES OF RESULTANT OF CLUSTER CU1,CU2       SUMCOS
      COMMON/B13/ KOU                                                   SUMCOS
      COMMON/B15/ SLL(500),SMM(500),SNN(500)                            SUMCOS
      COMMON/B19/ SX,SY,SZ                                              SUMCOS
      SX=SY=SZ=0.0                                                      SUMCOS
      DO 1 K=1,KOU                                                      SUMCOS
      SX=SX+SLL(K)                                                      SUMCOS
      SY=SY+SMM(K)                                                      SUMCOS
      SZ=SZ+SNN(K)                                                      SUMCOS
    1 CONTINUE                                                          SUMCOS
      RETURN                                                            SUMCOS
      END                                                               SUMCOS
      SUBROUTINE SORT(X,Y,N)                                            SORT
C SORT THE N ELEMENTS OF X AND Y IN DESCENDING ORDER OF X               SORT
      DIMENSION X(N),Y(N),LT(20),UT(20)                                 SORT
      J=N                                                               SORT
      I=1                                                               SORT
      M=1                                                               SORT
    5 IF(J-I-1)85,85,10                                                 SORT
   10 IP=(I+J)/2                                                        SORT
      T=X(IP)                                                           SORT
      X(IP)=X(I)                                                        SORT
      YT=Y(IP)                                                          SORT
      Y(IP)=Y(I)                                                        SORT
      IQ=J                                                              SORT
      K=I+1                                                             SORT
   15 IF(K-IQ) 25,25,65                                                 SORT
   25 IF(X(K)-T)40,55,55                                                SORT
   40 IF(X(IQ)-T)60,60,50                                               SORT
   50 P=X(K)                                                            SORT
      X(K)=X(IQ)                                                        SORT
      X(IQ)=P                                                           SORT
      YP=Y(K)                                                           SORT
      Y(K)=Y(IQ)                                                        SORT
      Y(IQ)=YP                                                          SORT
      IQ=IQ-1                                                           SORT
   55 K=K+1                                                             SORT
      GO TO 15                                                          SORT
   60 IQ=IQ-1                                                           SORT
      IF(IQ-K)65,40,40                                                  SORT
   65 X(I)=X(IQ)                                                        SORT
      X(IQ)=T                                                           SORT
      Y(I)=Y(IQ)                                                        SORT
      Y(IQ)=YT                                                          SORT
      IF(2*IQ-I-J)75,75,70                                              SORT
   70 LT(M)=I                                                           SORT
      UT(M)=IQ-1                                                        SORT
      I=IQ+1                                                            SORT
      GO TO 80                                                          SORT
   75 LT(M)=IQ+1                                                        SORT
      UT(M)=J                                                           SORT
      J=IQ-1                                                            SORT
   80 M=M+1                                                             SORT
```

```
      GO TO 5                                                            SORT
   85 IF(I-J)95,110,110                                                  SORT
   95 IF(X(I)-X(J))105,110,110                                           SORT
  105 P=X(I)                                                             SORT
      X(I)=X(J)                                                          SORT
      X(J)=P                                                             SORT
      YP=Y(I)                                                            SORT
      Y(I)=Y(J)                                                          SORT
      Y(J)=YP                                                            SORT
  110 IF(M-1)120,120,115                                                 SORT
  115 M=M-1                                                              SORT
      I=LT(M)                                                            SORT
      J=UT(M)                                                            SORT
      GO TO 5                                                            SORT
  120 RETURN                                                             SORT
      END                                                                SORT
      FUNCTION DIST3(I,J)                                                DIST3
C***** DIST3 = DISTANCE FROM POINT NUMBER I TO THE NEGATIVE OF POINT    DIST3
C***** NUMBER J.                                                         DIST3
      COMMON/B8/ SL(500),SM(500),SN(500)                                 DIST3
      DIST3=(SL(I)+SL(J))**2+(SM(I)+SM(J))**2+(SN(I)+SN(J))**2           DIST3
      RETURN                                                             DIST3
      END                                                                DIST3
      SUBROUTINE XLEVEL(KK1)                                             XLEVEL
      COMMON/B1/ A(500),D(500),N                                         XLEVEL
      COMMON/B9/ LPATCH(800),KBAND(800)                                  XLEVEL
      COMMON/B20/ RLEVEL(800)                                            XLEVEL
      FN=FLOAT(N)                                                        XLEVEL
      RLEVEL(KK1)=FLOAT(LPATCH(KK1))/FN                                  XLEVEL
      KK2=KK1+1                                                          XLEVEL
      DO 1 J=KK2,800                                                     XLEVEL
      JM1=J-1                                                            XLEVEL
      RLEVEL(J)=FLOAT(LPATCH(J)-LPATCH(JM1))/FN                          XLEVEL
    1 CONTINUE                                                           XLEVEL
      RETURN                                                             XLEVEL
      END                                                                XLEVEL
```

EXAMPLE OUTPUT

AXIAL DATA CONSISTING OF 6 HEMISPHERICALLY NORMAL CLUSTERS

NOBS = 144

LANS = 0

ISPHER = 0

RDIS = 12.5

RADIUS OF POINT PLOT = 4.0

RADIUS OF DENSITY PLOT = 6.0

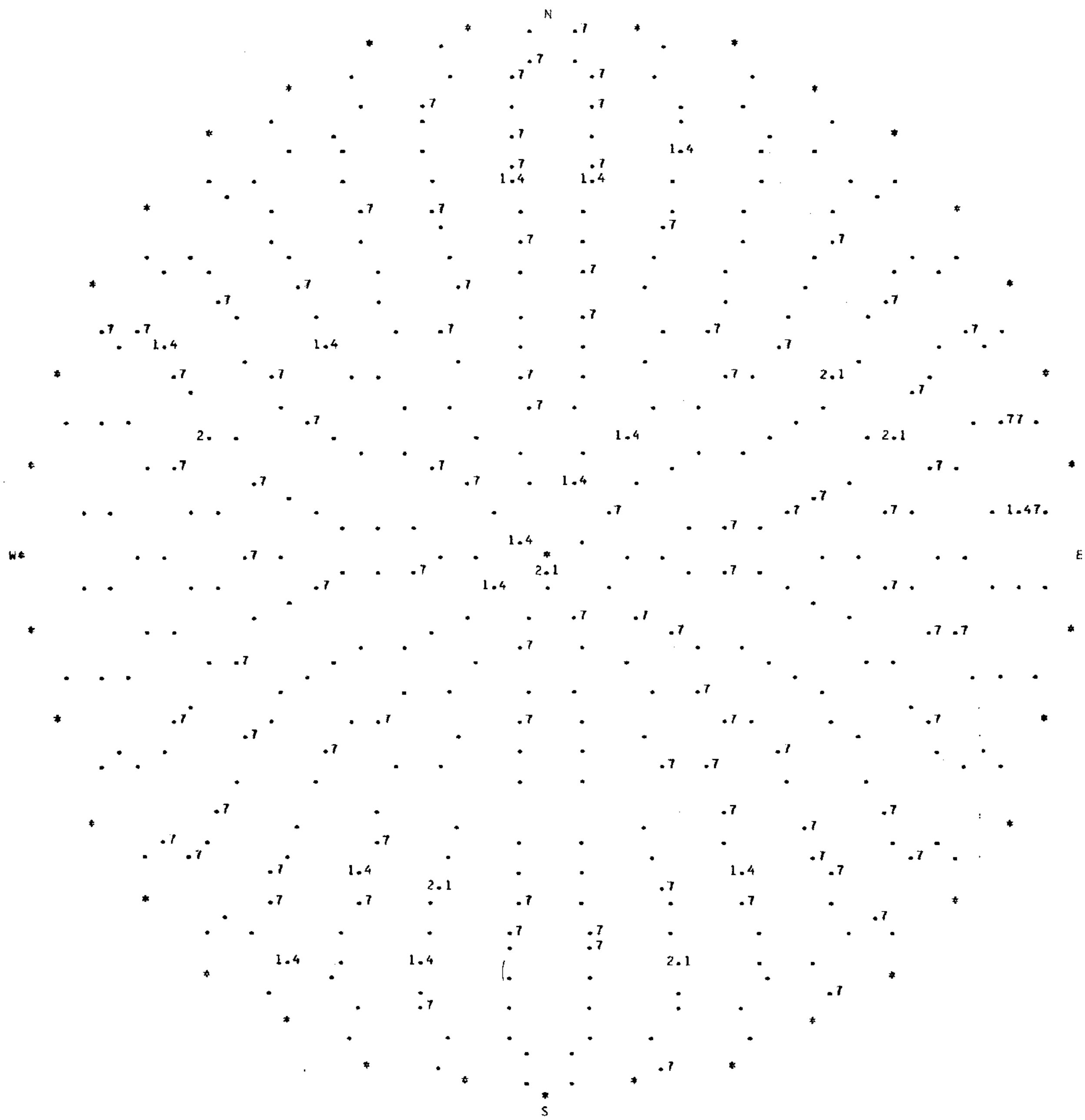
AZIMUTH OF THE DIP AND DIP OF THE DATA POINTS

360	60	0	61	1	57	4	57	351	58	356	69
349	65	9	71	347	49	0	76	20	55	348	75
357	79	337	60	8	41	22	66	24	67	339	42
6	36	352	84	329	64	2	89	335	37	32	44
72	60	72	61	74	57	77	63	66	59	64	55
80	52	69	75	56	53	53	51	80	77	67	79
77	41	74	40	87	76	71	81	50	69	49	47
98	51	85	80	102	63	44	70	101	68	47	37
144	57	146	63	145	56	140	65	136	56	134	68
157	67	143	46	158	69	128	68	129	48	161	68
165	53	143	41	148	40	129	75	142	39	131	80
172	58	149	85	137	34	113	68	178	61	161	85
217	59	218	59	210	64	215	54	222	68	205	53
223	71	204	68	199	52	229	70	232	50	197	55
226	76	239	58	215	80	193	66	220	36	241	69
212	84	187	56	247	54	186	51	193	78	236	83
288	59	288	61	288	59	287	58	291	69	278	66
301	54	297	71	299	71	286	76	272	50	277	46
298	44	306	68	269	69	310	49	310	48	301	78
262	58	269	78	269	38	278	85	321	59	300	87
168	3	261	3	168	4	336	5	302	8	42	9
230	10	14	14	19	15	163	15	248	15	201	17
324	19	131	19	26	20	356	21	23	21	297	21
257	22	113	24	187	28	86	28	94	29	341	30

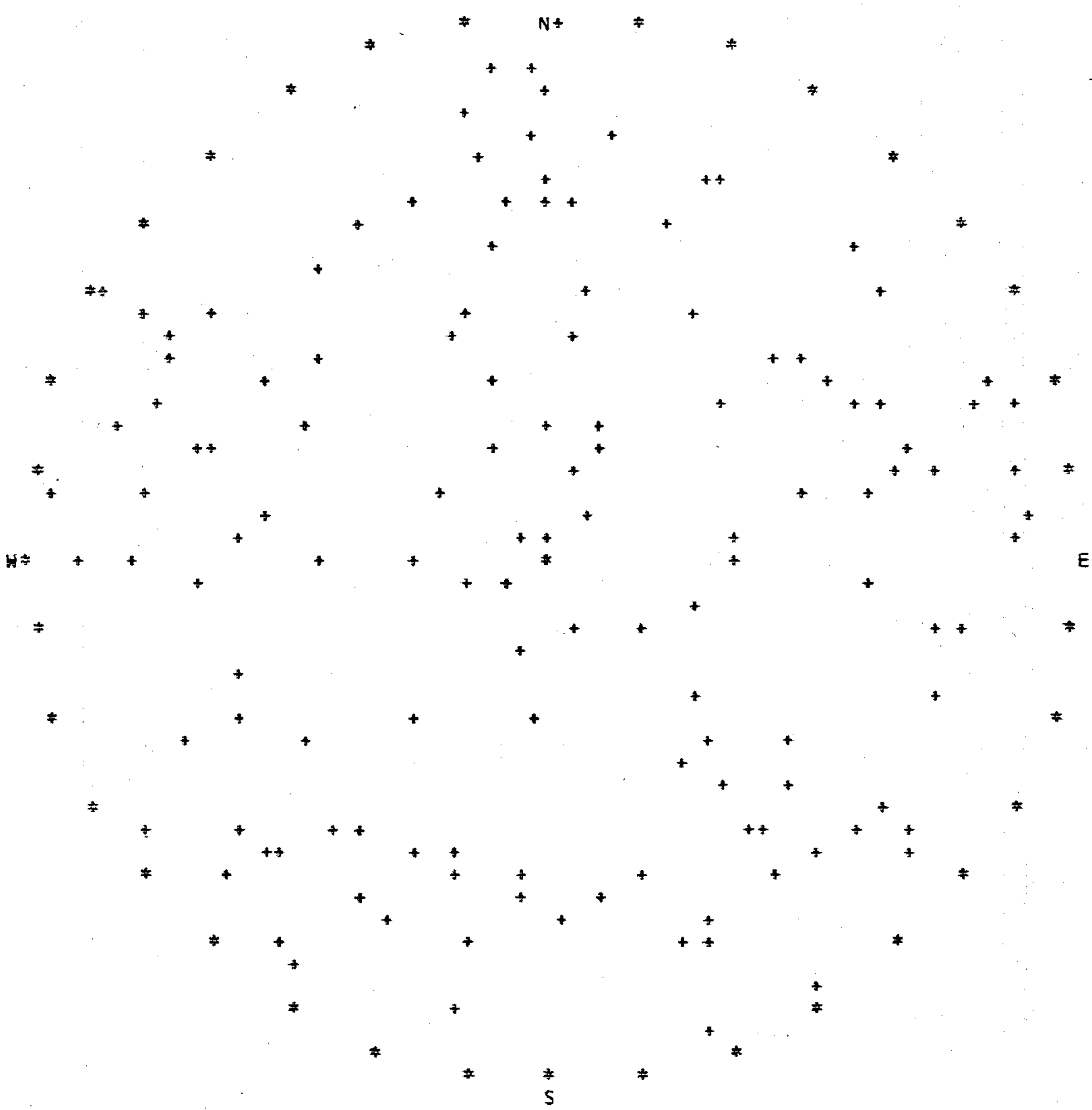
FOR RDIS = 12.50 PERCENTAGE OF SURFACE AREA IS 2.37
A POINT X IS DENSE IF THERE ARE 7 OBSERVATIONS
WHOSE EUCLIDEAN DISTANCE FROM X IS LESS THAN 4.74E-02.

DENSITY PLOT – LOWER AND UPPER HEMISPHERES

UPPER HEMISPHERE

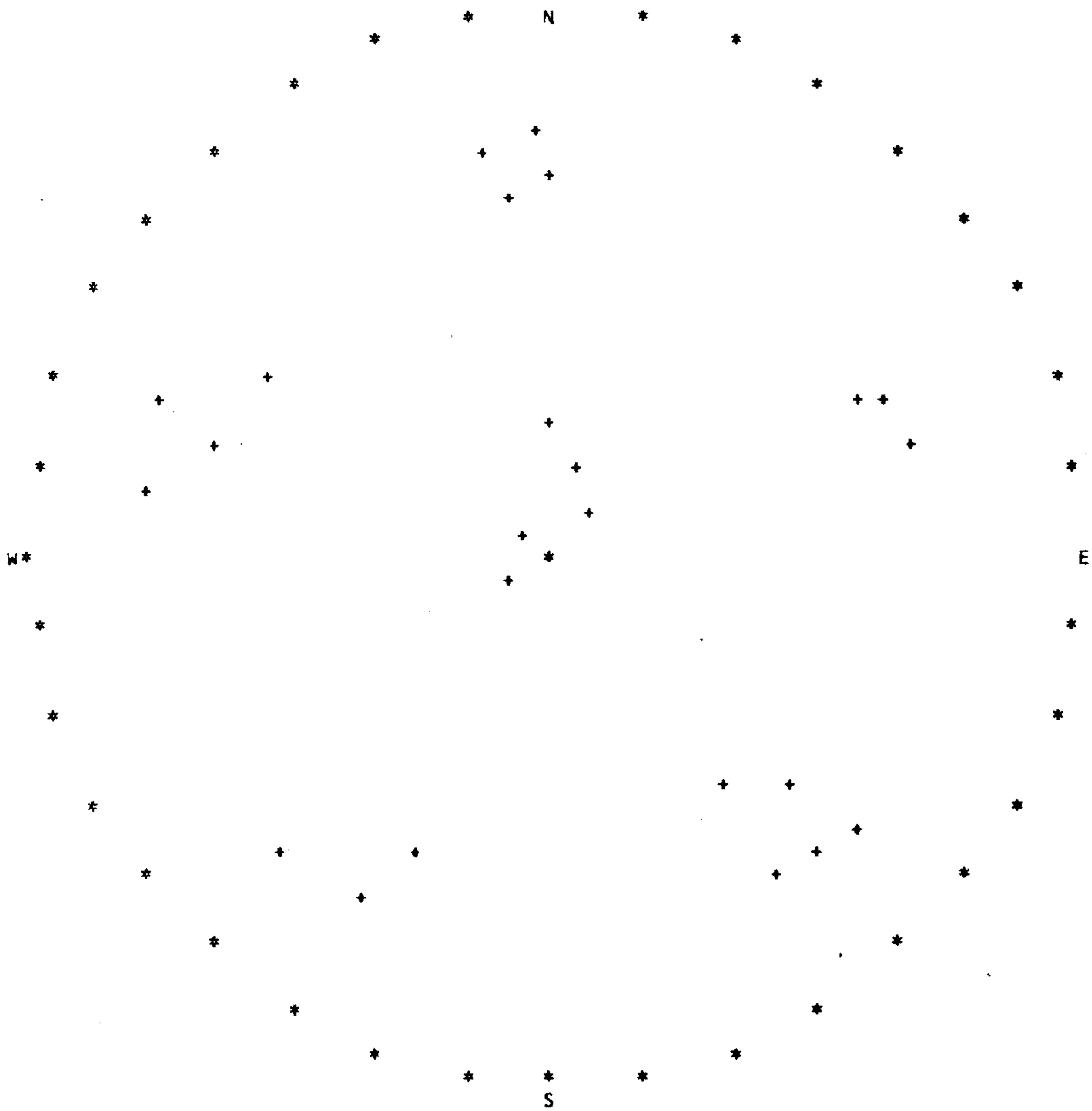


UPPER HEMISPHERE



DATA PLOT(POLAR EQUAL AREA PROJECTION).
THERE ARE 144 POINTS IN THE SAMPLE.

UPPER HEMISPHERE



THERE ARE 25 DENSE POINTS.

OUTPUT OF CLUSTERS, CLUSTER MEANS

CLUSTER NUMBER 1

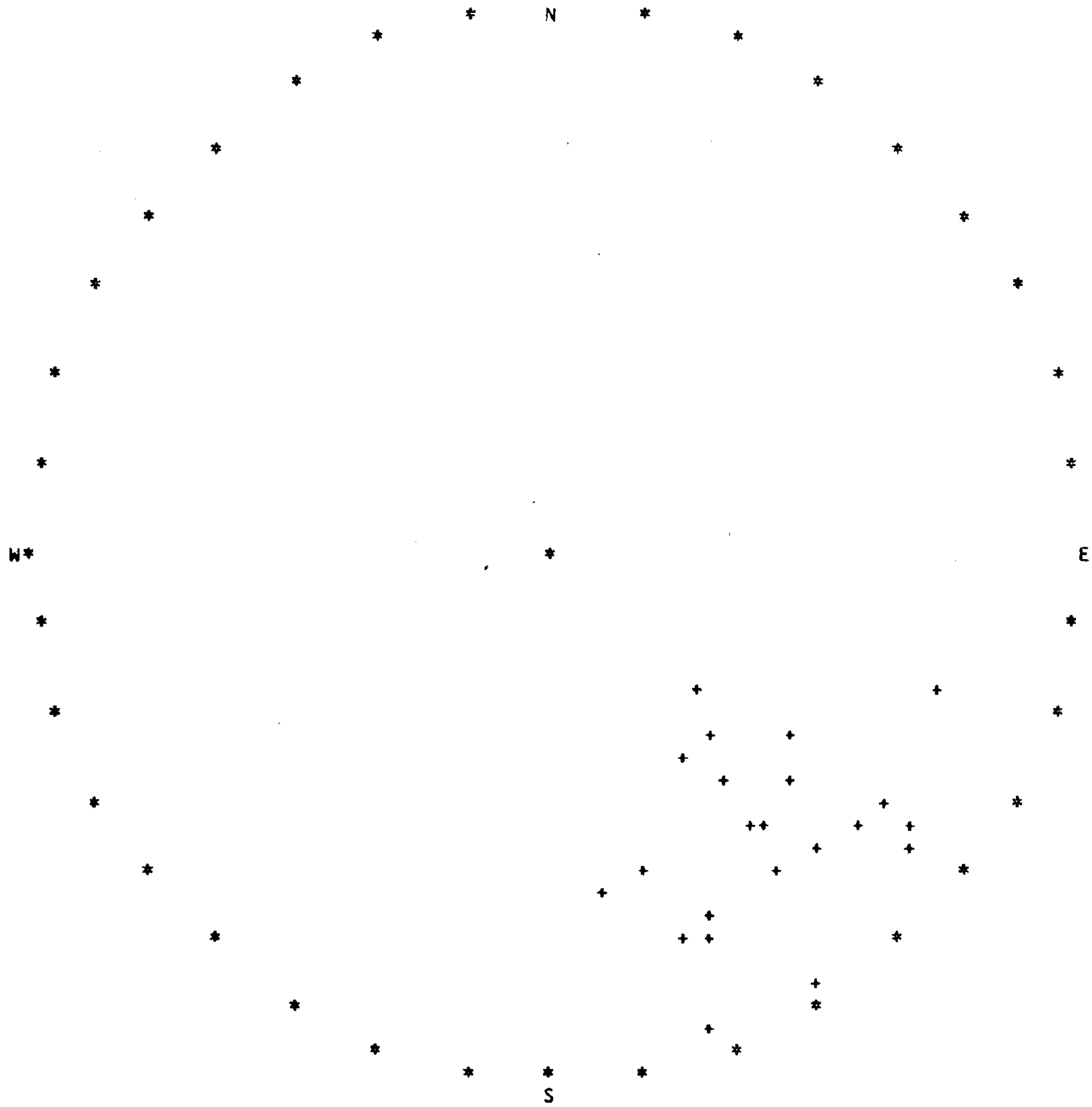
NUM	AZIMUTH	INCLINATION
1	149.00	85.00
2	161.00	85.00
3	131.00	80.00
4	129.00	75.00
5	113.00	68.00
6	128.00	68.00
7	134.00	68.00
8	157.00	67.00
9	158.00	69.00
10	161.00	68.00
11	140.00	65.00
12	146.00	63.00
13	136.00	56.00
14	144.00	57.00
15	145.00	56.00
16	172.00	58.00
17	165.00	53.00
18	129.00	48.00
19	143.00	46.00
20	143.00	41.00
21	142.00	39.00
22	148.00	40.00
23	137.00	34.00

MEAN
AZIMUTH 144.02 DEG
INCLINATION 59.66 DEG

RESULTANT AND ITS COMPONENTS - R,SX,SY,SZ.
21.80881 -15.23083 11.05826 11.01640

CENTER OF GRAVITY - (X,Y,Z) COORDINATES
-.6622 .4808 .4790

UPPER HEMISPHERE



CLUSTER NO. 1 CONTAINS 23 POINTS.
FOR THIS ANALYSIS, RDIS = 12.50

CLUSTER NUMBER 2

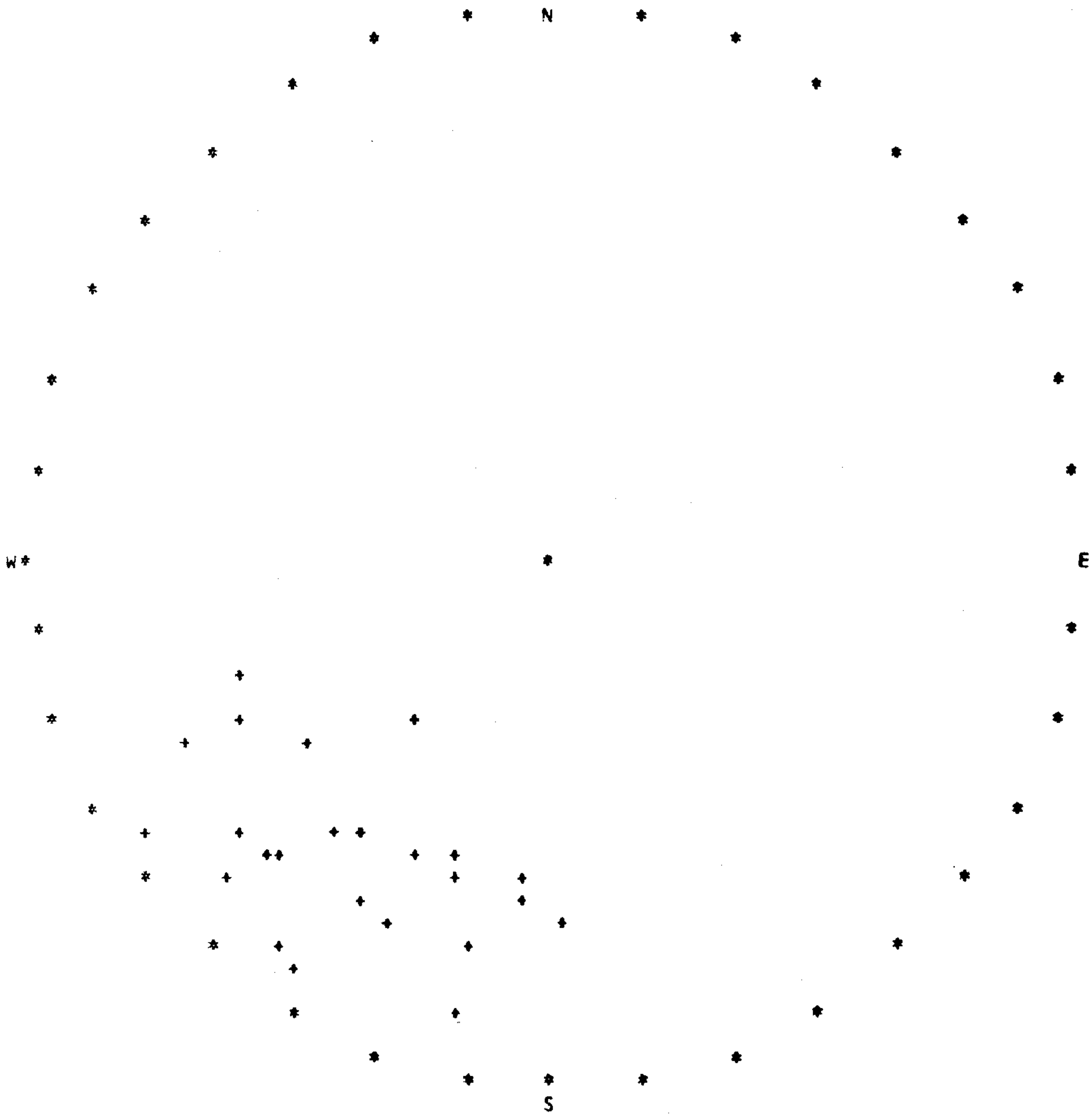
NUM	AZIMUTH	INCLINATION
1	212.00	84.00
2	215.00	80.00
3	236.00	83.00
4	193.00	78.00
5	226.00	76.00
6	223.00	71.00
7	193.00	66.00
8	204.00	68.00
9	222.00	68.00
10	229.00	70.00
11	241.00	69.00
12	178.00	61.00
13	210.00	64.00
14	187.00	56.00
15	217.00	59.00
16	218.00	59.00
17	239.00	58.00
18	186.00	51.00
19	197.00	55.00
20	199.00	52.00
21	205.00	53.00
22	215.00	54.00
23	247.00	54.00
24	232.00	50.00
25	220.00	36.00

MEAN
AZIMUTH 213.97 DEG
INCLINATION 61.82 DEG

RESULTANT AND ITS COMPONENTS - R,SX,SY,SZ.
23.55255 -17.21700 -11.60176 11.12190

CENTER OF GRAVITY - (X,Y,Z) COORDINATES
-.6887 -.4641 .4449

UPPER HEMISPHERE



CLUSTER NO. 2 CONTAINS 25 POINTS.
FOR THIS ANALYSIS, RLIS = 12.50

CLUSTER NUMBER 3

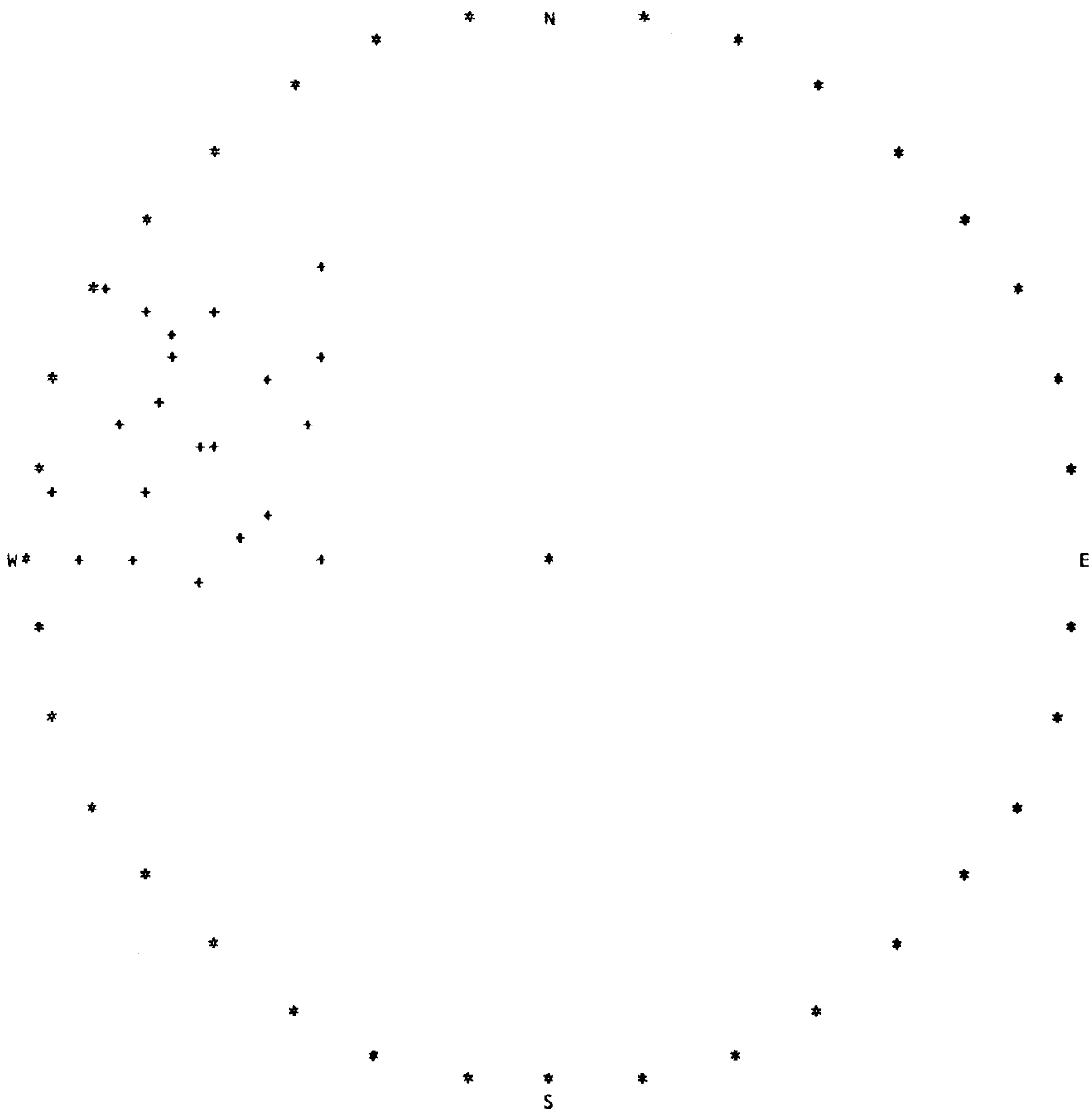
NUM	AZIMUTH	INCLINATION
1	278.00	85.00
2	300.00	87.00
3	269.00	78.00
4	286.00	76.00
5	301.00	78.00
6	297.00	71.00
7	299.00	71.00
8	269.00	69.00
9	278.00	66.00
10	291.00	69.00
11	306.00	68.00
12	288.00	61.00
13	262.00	58.00
14	287.00	58.00
15	288.00	59.00
16	288.00	59.00
17	321.00	59.00
18	301.00	54.00
19	272.00	50.00
20	277.00	46.00
21	310.00	49.00
22	310.00	48.00
23	298.00	44.00
24	269.00	38.00

MEAN
AZIMUTH 289.42 DEG
INCLINATION 61.73 DEG

RESULTANT AND ITS COMPONENTS - R,SX,SY,SZ.
22.79199 6.67552 -18.93151 10.79399

CENTER OF GRAVITY - (X,Y,Z) COORDINATES
.2781 -.7888 .4497

UPPER HEMISPHERE



CLUSTER NO.   3 CONTAINS   24 POINTS.
FOR THIS ANALYSIS, RDIS = 12.50

CLUSTER NUMBER 4

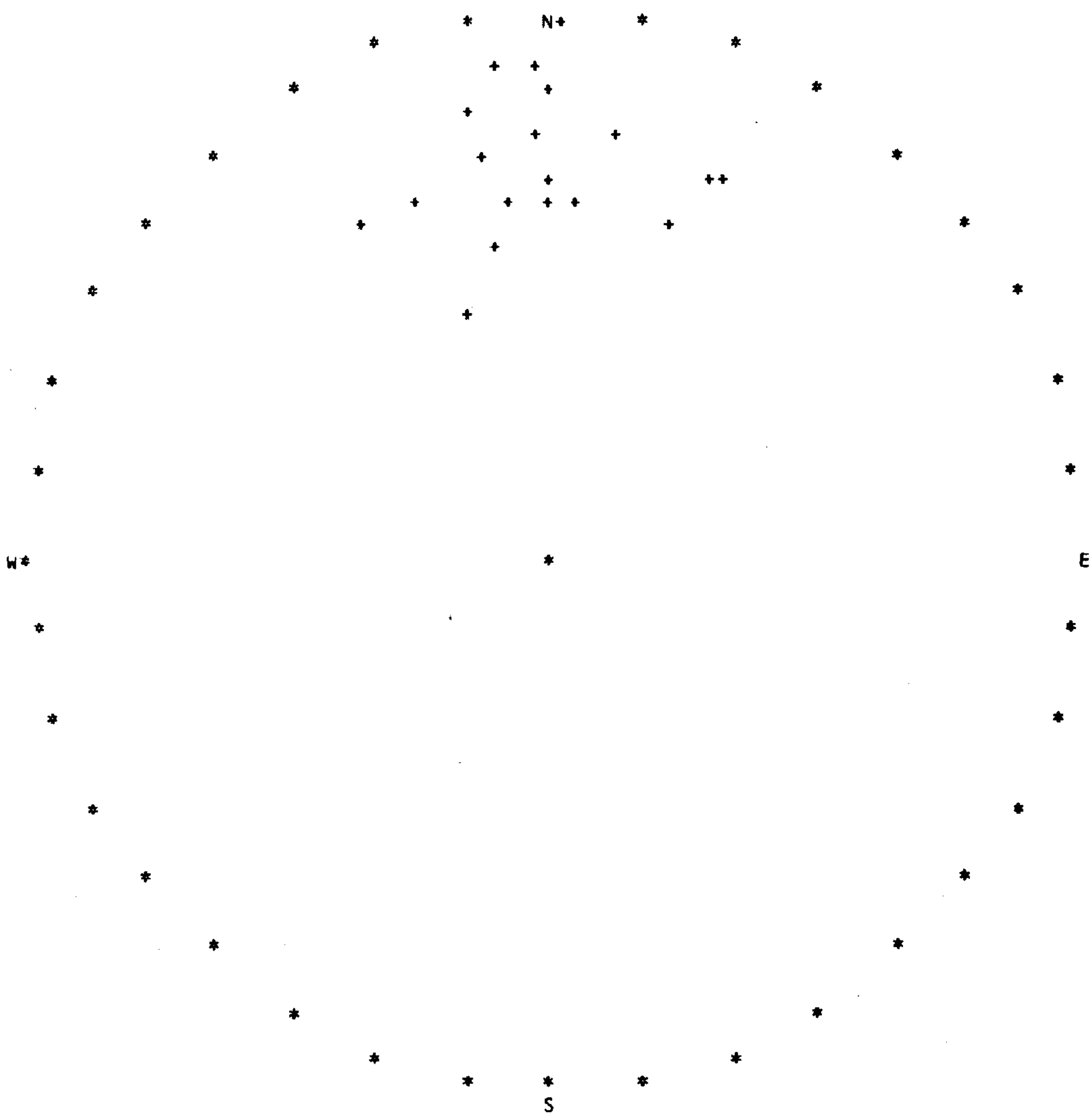
NUM	AZIMUTH	INCLINATION
1	2.00	89.00
2	352.00	84.00
3	0.00	76.00
4	348.00	75.00
5	357.00	79.00
6	9.00	71.00
7	22.00	66.00
8	24.00	67.00
9	356.00	69.00
10	0.00	61.00
11	329.00	64.00
12	349.00	65.00
13	1.00	57.00
14	4.00	57.00
15	337.00	60.00
16	351.00	58.00
17	360.00	60.00
18	20.00	55.00
19	347.00	49.00
20	339.00	42.00

MEAN
AZIMUTH 357.61 DEG
INCLINATION 64.54 DEG

RESULTANT AND ITS COMPONENTS - R,SX,SY,SZ.
19.14823 17.27309 -.71946 8.23271

CENTER OF GRAVITY - (X,Y,Z) COORDINATES
.8637 -.0360 .4116

UPPER HEMISPHERE



CLUSTER NO. 4 CONTAINS 2C PCINTS.
FOR THIS ANALYSIS, RDIS = 12.50

CLUSTER NUMBER 5

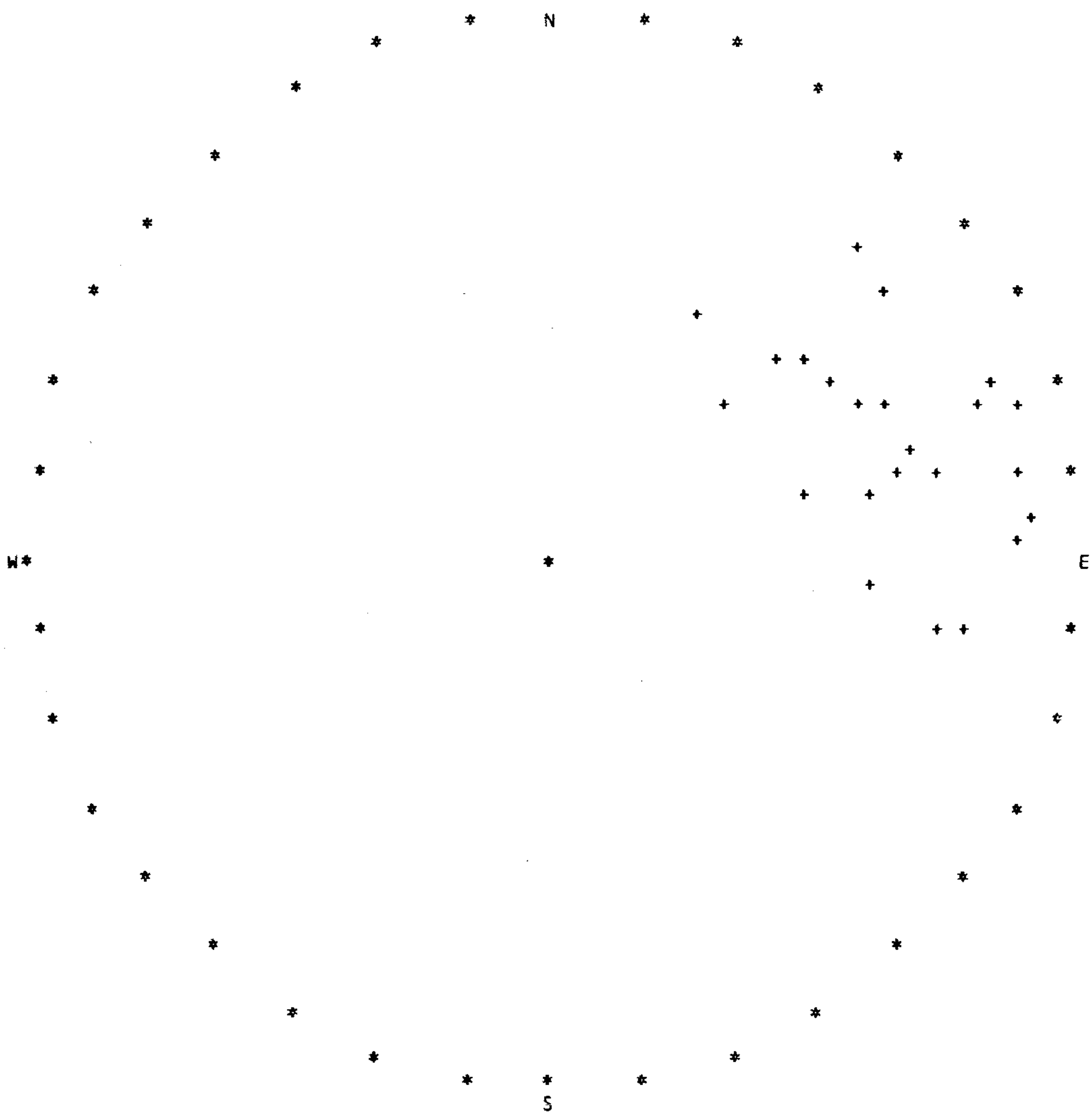
NUM	AZIMUTH	INCLINATION
1	71.00	81.00
2	85.00	80.00
3	67.00	79.00
4	69.00	75.00
5	80.00	77.00
6	87.00	76.00
7	44.00	70.00
8	50.00	69.00
9	101.00	68.00
10	72.00	61.00
11	77.00	63.00
12	102.00	63.00
13	66.00	59.00
14	72.00	60.00
15	74.00	57.00
16	53.00	51.00
17	56.00	53.00
18	64.00	55.00
19	80.00	52.00
20	98.00	51.00
21	49.00	47.00
22	32.00	44.00
23	77.00	41.00
24	47.00	37.00
25	74.00	40.00

MEAN
AZIMUTH 70.68 DEG
INCLINATION 59.24 DEG

RESULTANT AND ITS COMPONENTS - R,SX,SY,SZ.
23.54675 6.69527 19.09328 12.04447

CENTER OF GRAVITY - (X,Y,Z) COORDINATES
.2678 .7637 .4818

UPPER HEMISPHERE



CLUSTER NO. 5 CONTAINS 25 POINTS.
FOR THIS ANALYSIS, RDIS = 12.50

CLUSTER NUMBER 6

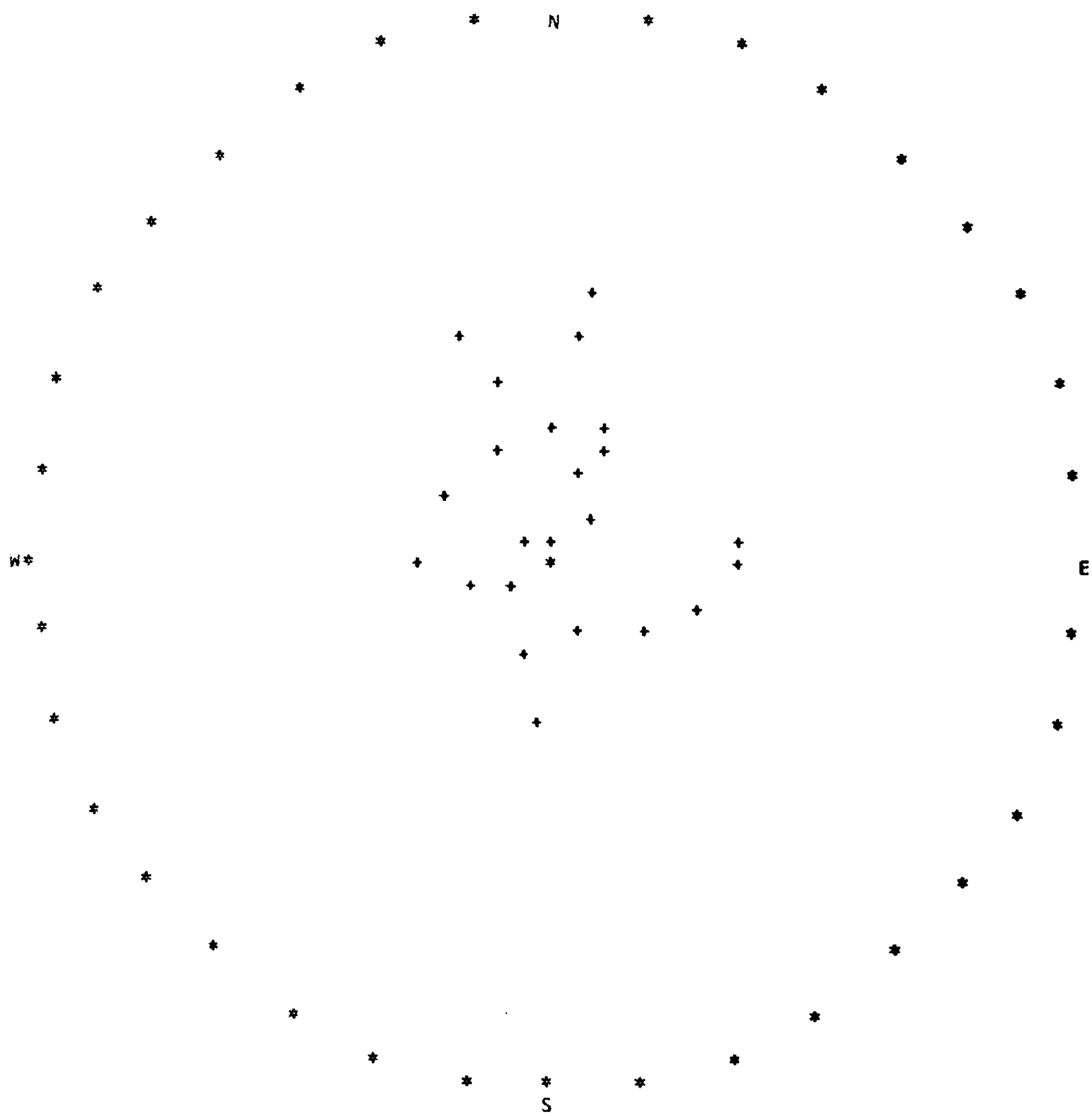
NUM	AZIMUTH	INCLINATION
1	8.00	41.00
2	6.00	36.00
3	335.00	37.00
4	86.00	28.00
5	94.00	29.00
6	187.00	28.00
7	341.00	30.00
8	23.00	21.00
9	26.00	20.00
10	113.00	24.00
11	257.00	22.00
12	297.00	21.00
13	356.00	21.00
14	131.00	19.00
15	201.00	17.00
16	324.00	19.00
17	14.00	14.00
18	19.00	15.00
19	42.00	9.00
20	163.00	15.00
21	230.00	10.00
22	248.00	15.00
23	302.00	8.00
24	336.00	5.00
25	168.00	4.00
26	168.00	3.00
27	261.00	3.00

MEAN
AZIMUTH 6.59 DEG
INCLINATION 6.04 DEG

RESULTANT AND ITS COMPONENTS - R,SX,SY,SZ.
25.25966 2.64190 .30534 25.11926

CENTER OF GRAVITY - (X,Y,Z) COORDINATES
.0978 .0113 .9303

UPPER HEMISPHERE



CLUSTER NO. 6 CONTAINS 27 POINTS.
FOR THIS ANALYSIS, RDIS = 12.50

OBJECTIVE FUNCTION = 37.301

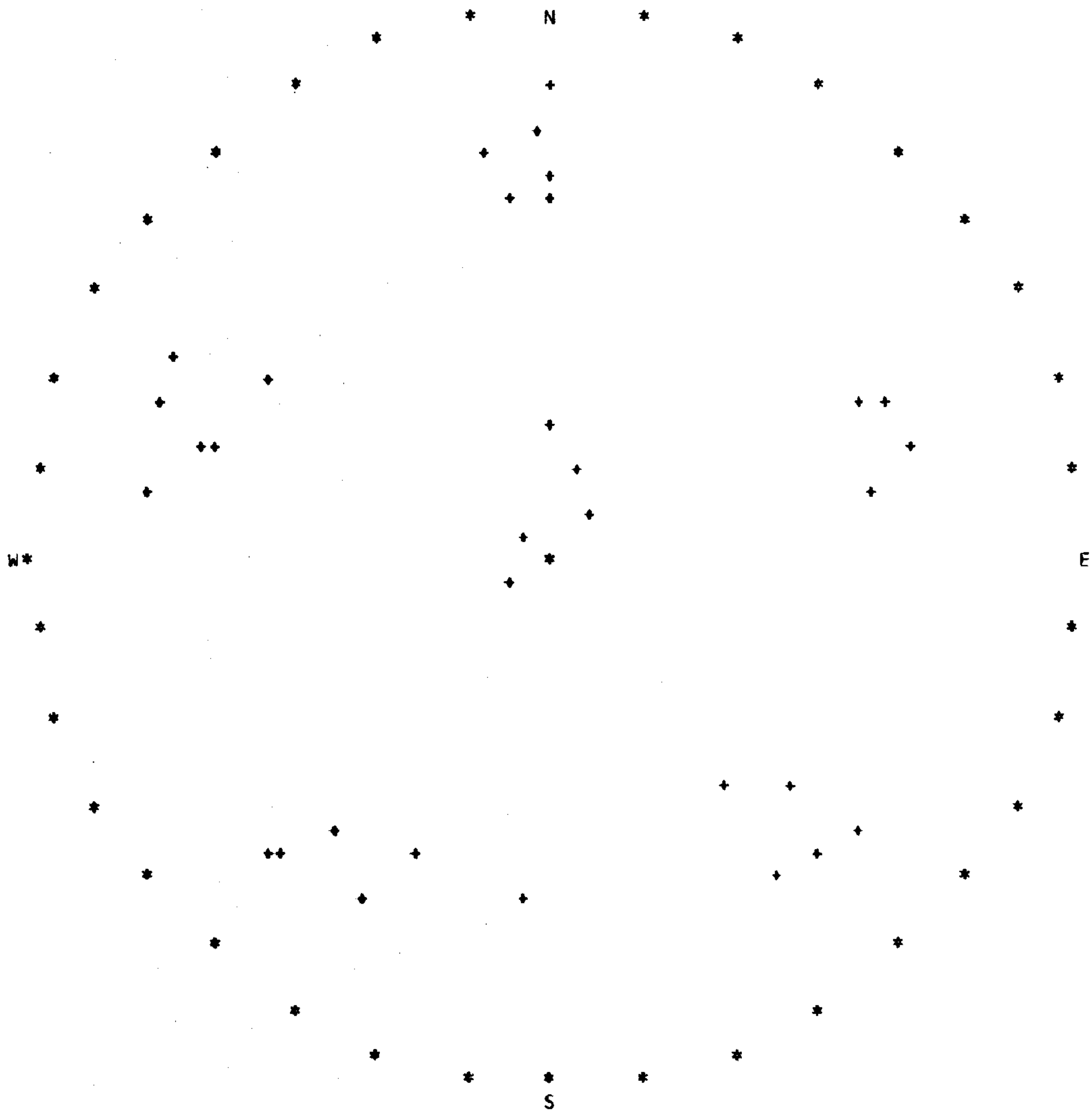
TIME REQUIRED TO ANALYZE 144 DATA POINTS 8.584 SECONDS

```
FOR NEXT ANALYSIS OF SAMPLE
LANS =     0

RDIS =  13.5


FOR RDIS = 13.50 PERCENTAGE OF SURFACE AREA IS  2.76
A POINT X IS DENSE IF THERE ARE     7 OBSERVATIONS
WHOSE EUCLIDEAN DISTANCE FROM X IS LESS THAN  5.53E-02.
```

UPPER HEMISPHERE



THERE ARE 33 DENSE POINTS.

OUTPUT OF CLUSTERS, CLUSTER MEANS

CLUSTER NUMBER 1

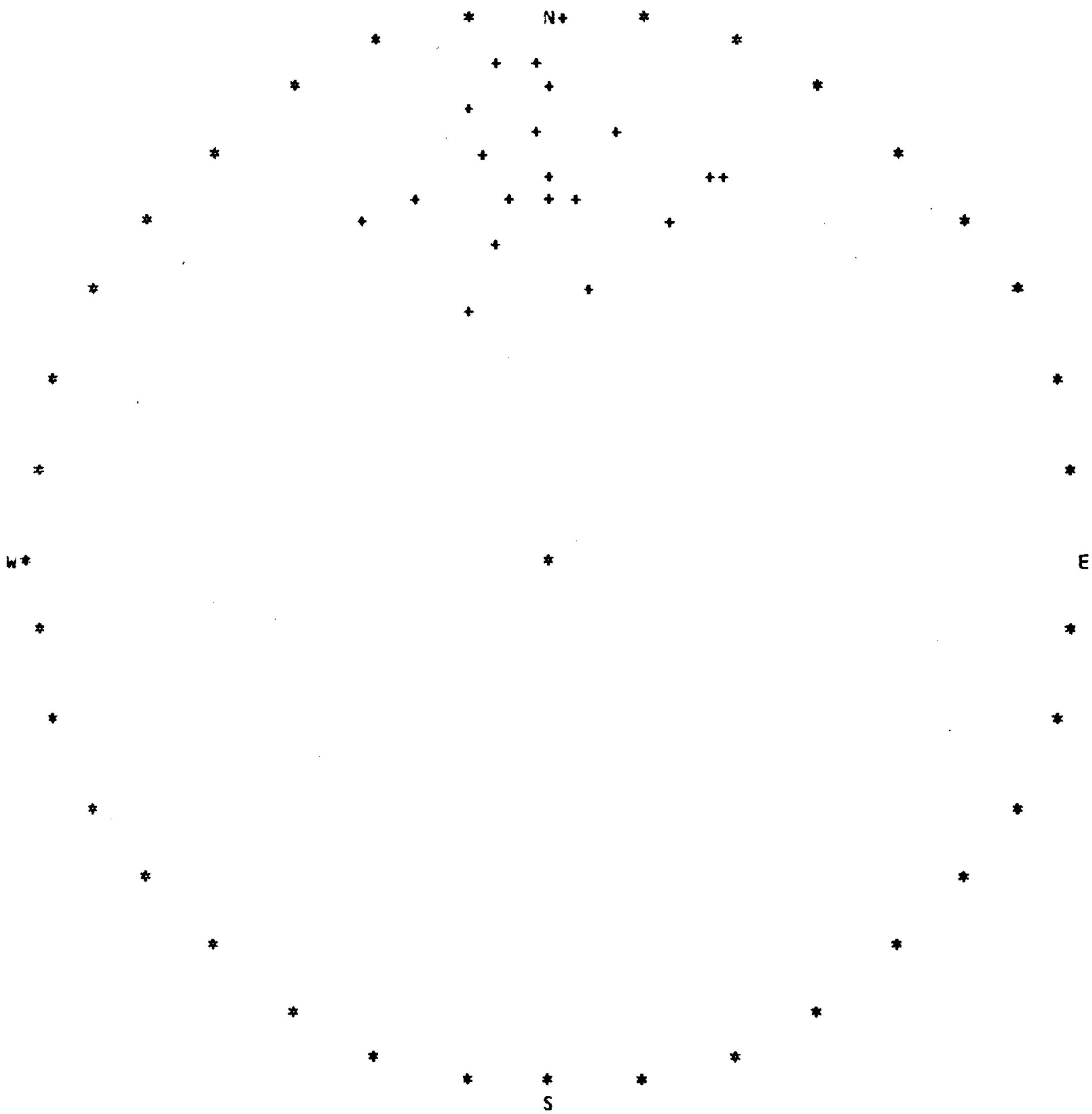
NUM	AZIMUTH	INCLINATION
1	2.00	89.00
2	352.00	84.00
3	0.00	76.00
4	348.00	75.00
5	357.00	79.00
6	9.00	71.00
7	22.00	66.00
8	24.00	67.00
9	356.00	69.00
10	0.00	61.00
11	329.00	64.00
12	349.00	65.00
13	1.00	57.00
14	4.00	57.00
15	337.00	60.00
16	351.00	58.00
17	360.00	60.00
18	20.00	55.00
19	347.00	49.00
20	8.00	41.00
21	339.00	42.00

MEAN
AZIMUTH 357.99 DEG
INCLINATION 63.38 DEG

RESULTANT AND ITS COMPONENTS - R,SX,SY,SZ.
20.05976 17.92276 -.62815 8.98742

CENTER OF GRAVITY - (X,Y,Z) COORDINATES
.8535 -.0299 .4280

UPPER HEMISPHERE



CLUSTER NO. 1 CONTAINS 21 POINTS.
FOR THIS ANALYSIS, RDIS = 13.50

CLUSTER NUMBER 2

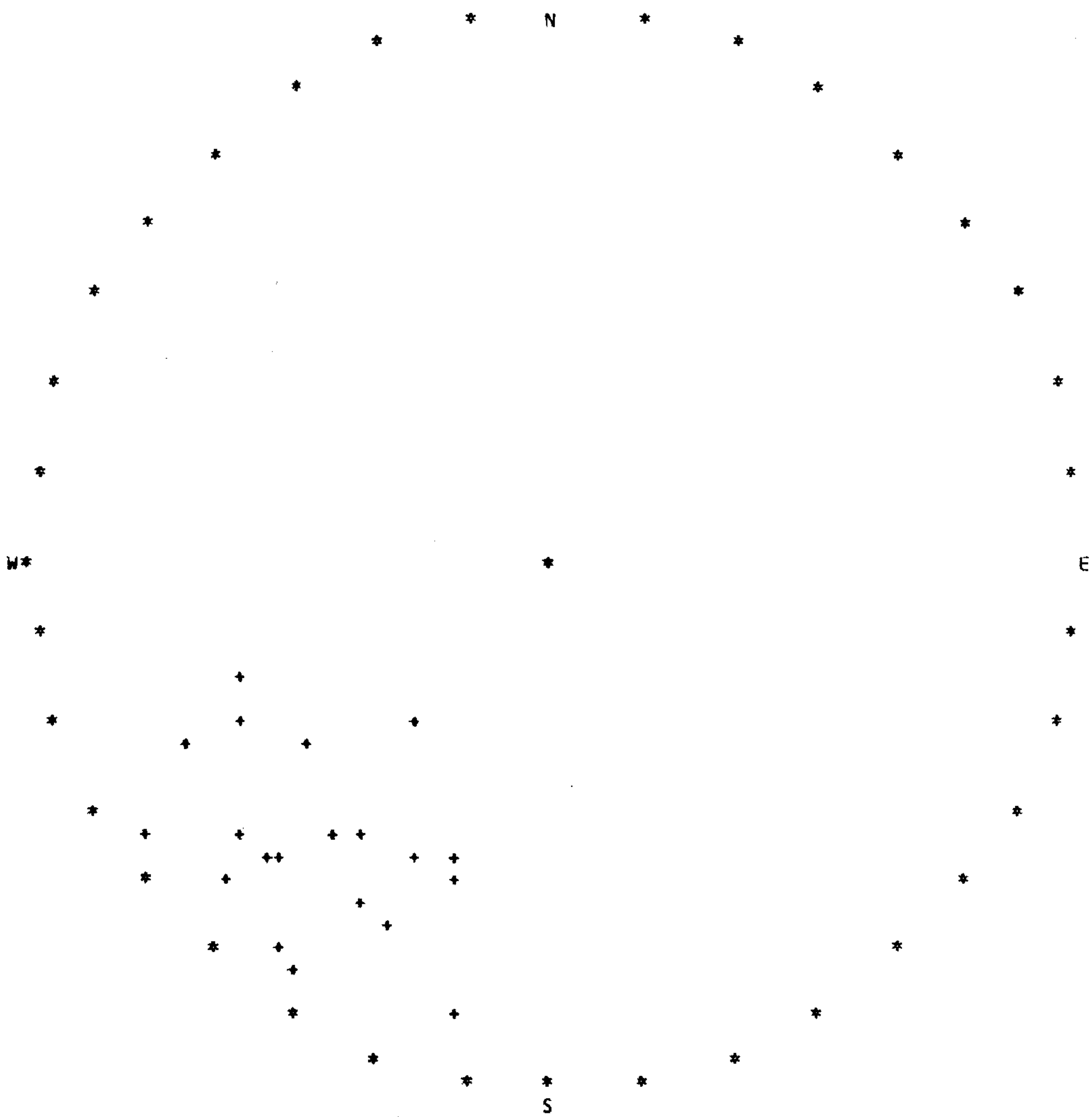
NUM	AZIMUTH	INCLINATION
1	212.00	84.00
2	215.00	80.00
3	236.00	83.00
4	193.00	78.00
5	226.00	76.00
6	223.00	71.00
7	204.00	68.00
8	222.00	68.00
9	229.00	70.00
10	241.00	69.00
11	210.00	64.00
12	217.00	59.00
13	218.00	59.00
14	239.00	58.00
15	197.00	55.00
16	199.00	52.00
17	205.00	53.00
18	215.00	54.00
19	247.00	54.00
20	232.00	50.00
21	220.00	36.00

MEAN
AZIMUTH 219.06 DEG
INCLINATION 63.13 DEG

RESULTANT AND ITS COMPONENTS - R,SX,SY,SZ.
20.00528 -13.85704 -11.24451 9.04184

CENTER OF GRAVITY - (X,Y,Z) COORDINATES
-.6599 -.5355 .4306

UPPER HEMISPHERE



CLUSTER NO. 2 CONTAINS 21 POINTS.
FOR THIS ANALYSIS, RDIS = 13.50

CLUSTER NUMBER 3

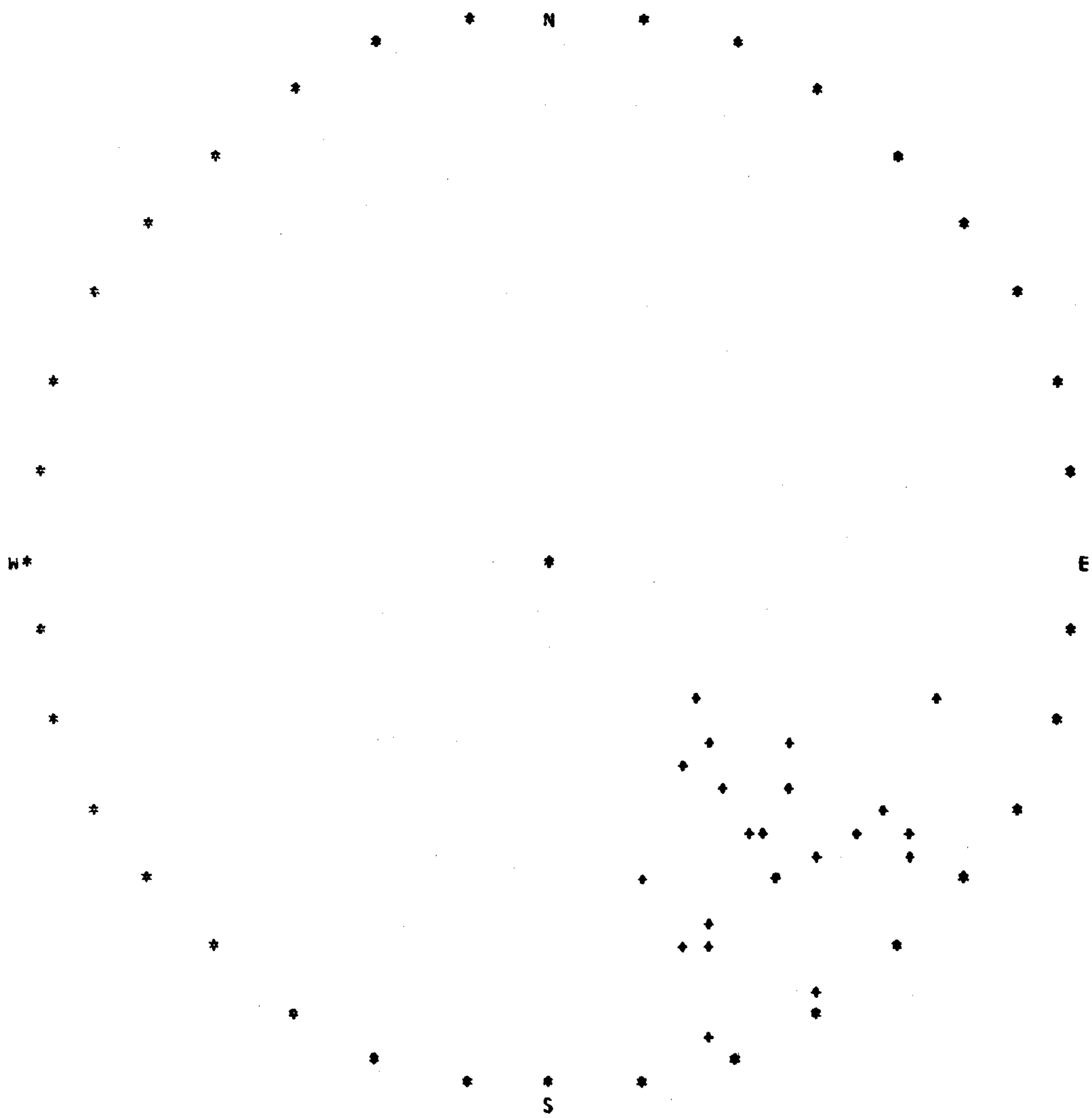
NUM	AZIMUTH	INCLINATION
1	149.00	85.00
2	161.00	85.00
3	131.00	80.00
4	129.00	75.00
5	113.00	68.00
6	128.00	68.00
7	134.00	68.00
8	157.00	67.00
9	158.00	69.00
10	161.00	68.00
11	140.00	65.00
12	146.00	63.00
13	136.00	56.00
14	144.00	57.00
15	145.00	56.00
16	165.00	53.00
17	129.00	48.00
18	143.00	46.00
19	143.00	41.00
20	142.00	39.00
21	148.00	40.00
22	137.00	34.00

MEAN
AZIMUTH 142.76 DEG
INCLINATION 59.88 DEG

RESULTANT AND ITS COMPONENTS - R,SX,SY,SZ.
20.89873 -14.39103 10.94023 10.48648

CENTER OF GRAVITY - (X,Y,Z) COORDINATES
-.6541 .4973 .4767

UPPER HEMISPHERE



CLUSTER NO. 3 CONTAINS 22 POINTS.
FOR THIS ANALYSIS, RDIS = 13.50

CLUSTER NUMBER 4

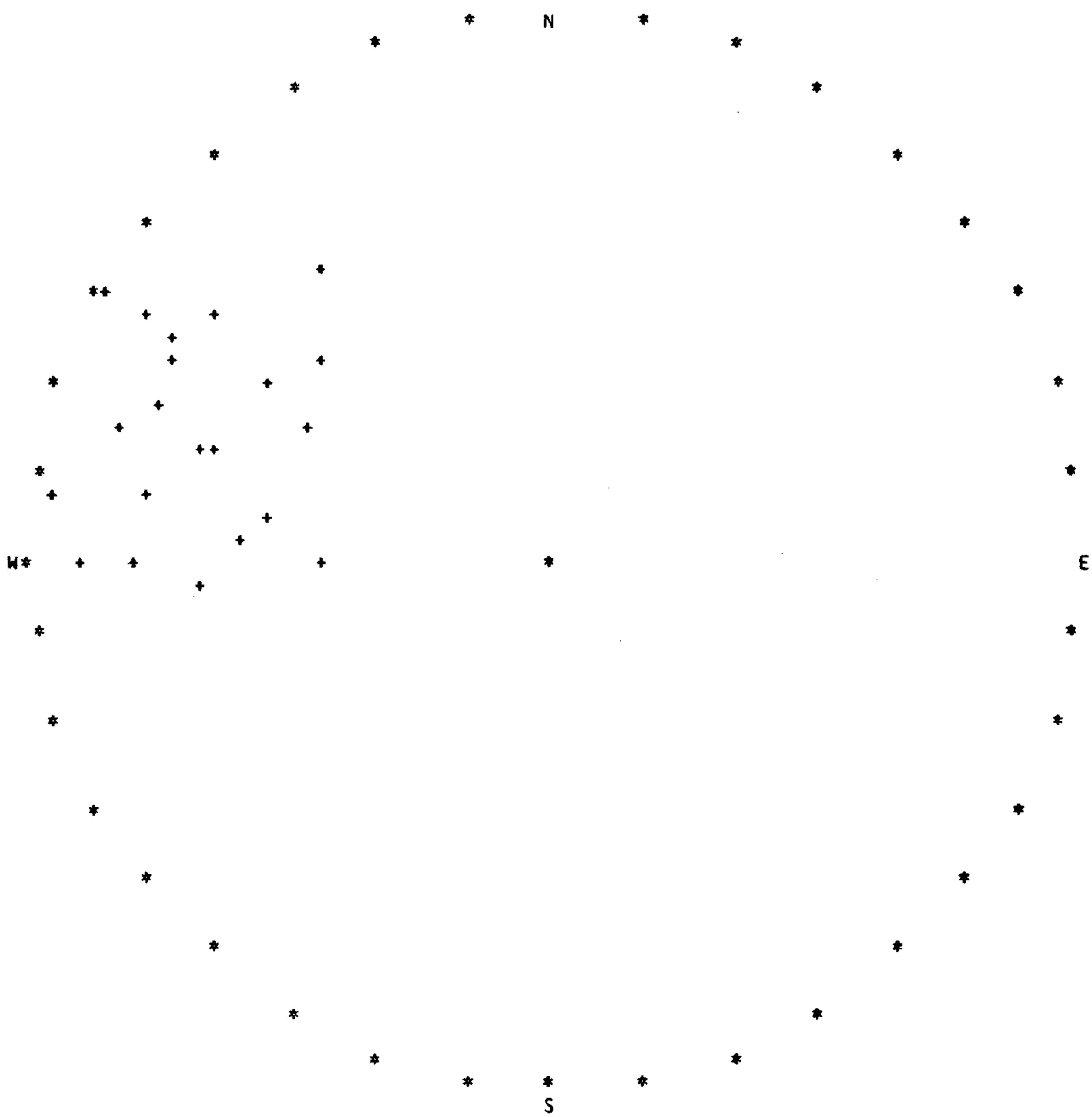
NUM	AZIMUTH	INCLINATION
1	278.00	85.00
2	300.00	87.00
3	269.00	78.00
4	286.00	76.00
5	301.00	78.00
6	297.00	71.00
7	299.00	71.00
8	269.00	69.00
9	278.00	66.00
10	291.00	69.00
11	306.00	68.00
12	288.00	61.00
13	262.00	58.00
14	287.00	58.00
15	288.00	59.00
16	288.00	59.00
17	321.00	59.00
18	301.00	54.00
19	272.00	50.00
20	277.00	46.00
21	310.00	49.00
22	310.00	48.00
23	298.00	44.00
24	269.00	38.00

MEAN
AZIMUTH 289.42 DEG
INCLINATION 61.73 DEG

RESULTANT AND ITS COMPONENTS - R,SX,SY,SZ.
22.79199 6.67552 -18.93151 10.79399

CENTER OF GRAVITY - (X,Y,Z) COORDINATES
.2781 -.7888 .4497

UPPER HEMISPHERE



CLUSTER NO. 4 CONTAINS 24 POINTS.
FOR THIS ANALYSIS, RDIS = 13.50

CLUSTER NUMBER 5

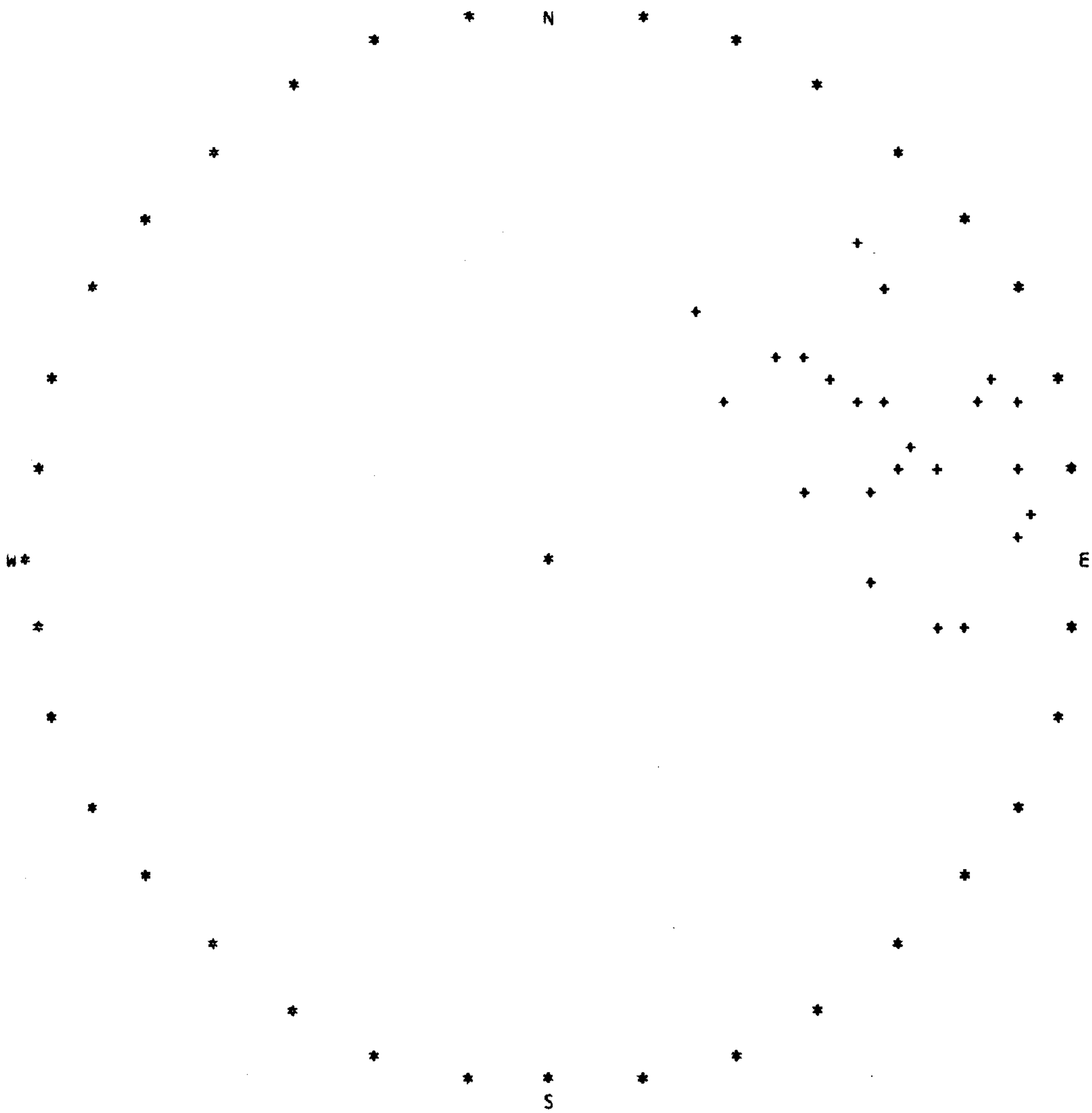
NUM	AZIMUTH	INCLINATION
1	71.00	81.00
2	85.00	80.00
3	67.00	79.00
4	69.00	75.00
5	80.00	77.00
6	87.00	76.00
7	44.00	70.00
8	50.00	69.00
9	101.00	68.00
10	72.00	61.00
11	77.00	63.00
12	102.00	63.00
13	66.00	59.00
14	72.00	60.00
15	74.00	57.00
16	53.00	51.00
17	56.00	53.00
18	64.00	55.00
19	80.00	52.00
20	98.00	51.00
21	49.00	47.00
22	32.00	44.00
23	77.00	41.00
24	47.00	37.00
25	74.00	40.00

MEAN
AZIMUTH 70.68 DEG
INCLINATION 59.24 DEG

RESULTANT AND ITS COMPONENTS - R,SX,SY,SZ.
23.54675 6.69527 19.09328 12.04447

CENTER OF GRAVITY - (X,Y,Z) COORDINATES
.2678 .7637 .4818

UPPER HEMISPHERE



CLUSTER NO. 5 CONTAINS 25 POINTS.
FOR THIS ANALYSIS, RDIS = 13.50

CLUSTER NUMBER 6

NUM	AZIMUTH	INCLINATION
1	193.00	66.00
2	178.00	61.00
3	172.00	58.00
4	187.00	56.00
5	186.00	51.00

MEAN
AZIMUTH 183.26 DEG
INCLINATION 58.18 DEG

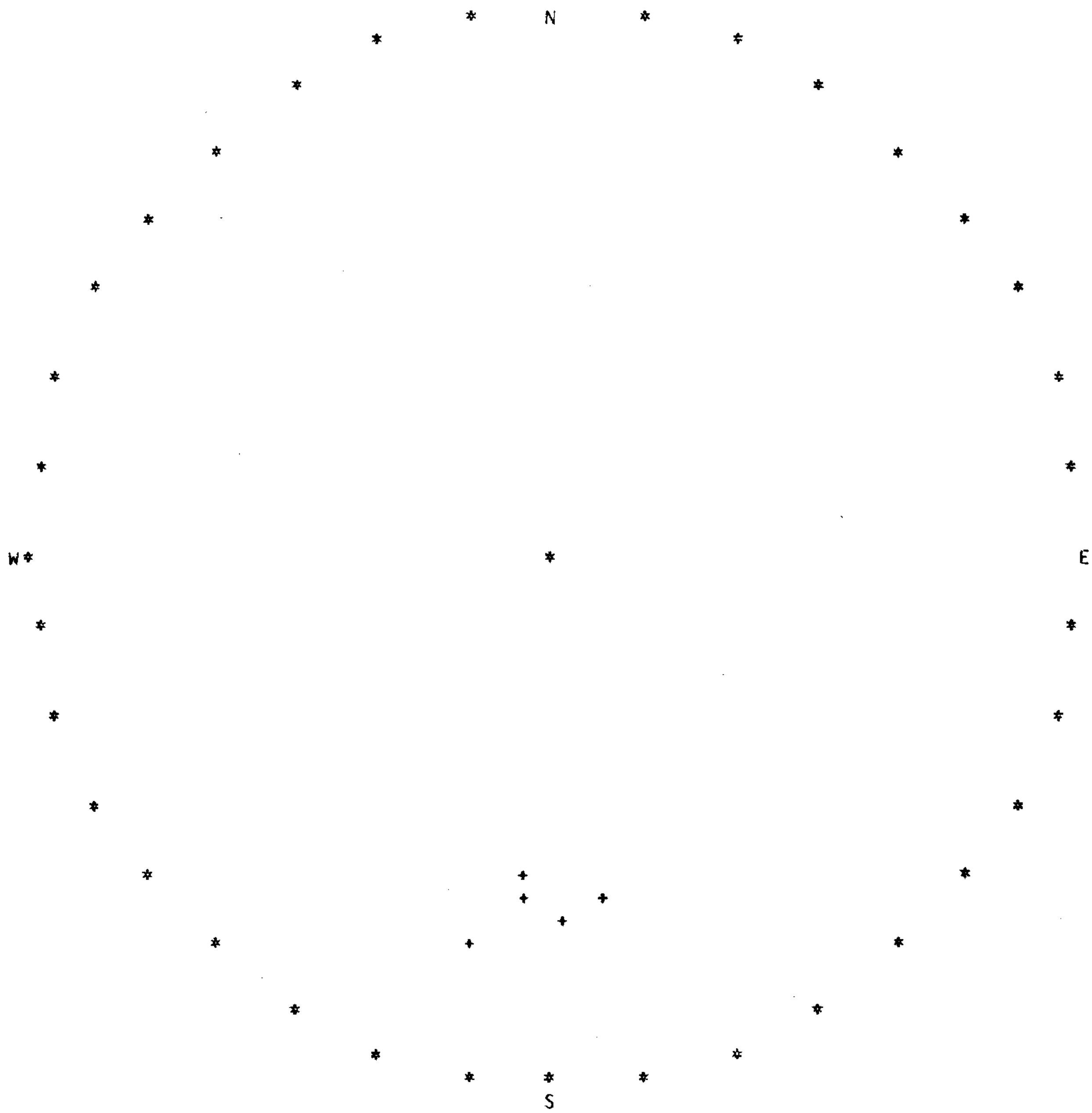
RESULTANT AND ITS COMPONENTS - R,SX,SY,SZ.

4.95047	-4.19976	-.23922	2.60998

CENTER OF GRAVITY - (X,Y,Z) COORDINATES

-.8400	-.0478	.5220

UPPER HEMISPHERE



CLUSTER NO. 6 CONTAINS 5 POINTS.
FOR THIS ANALYSIS, RDIS = 13.50

CLUSTER NUMBER 7

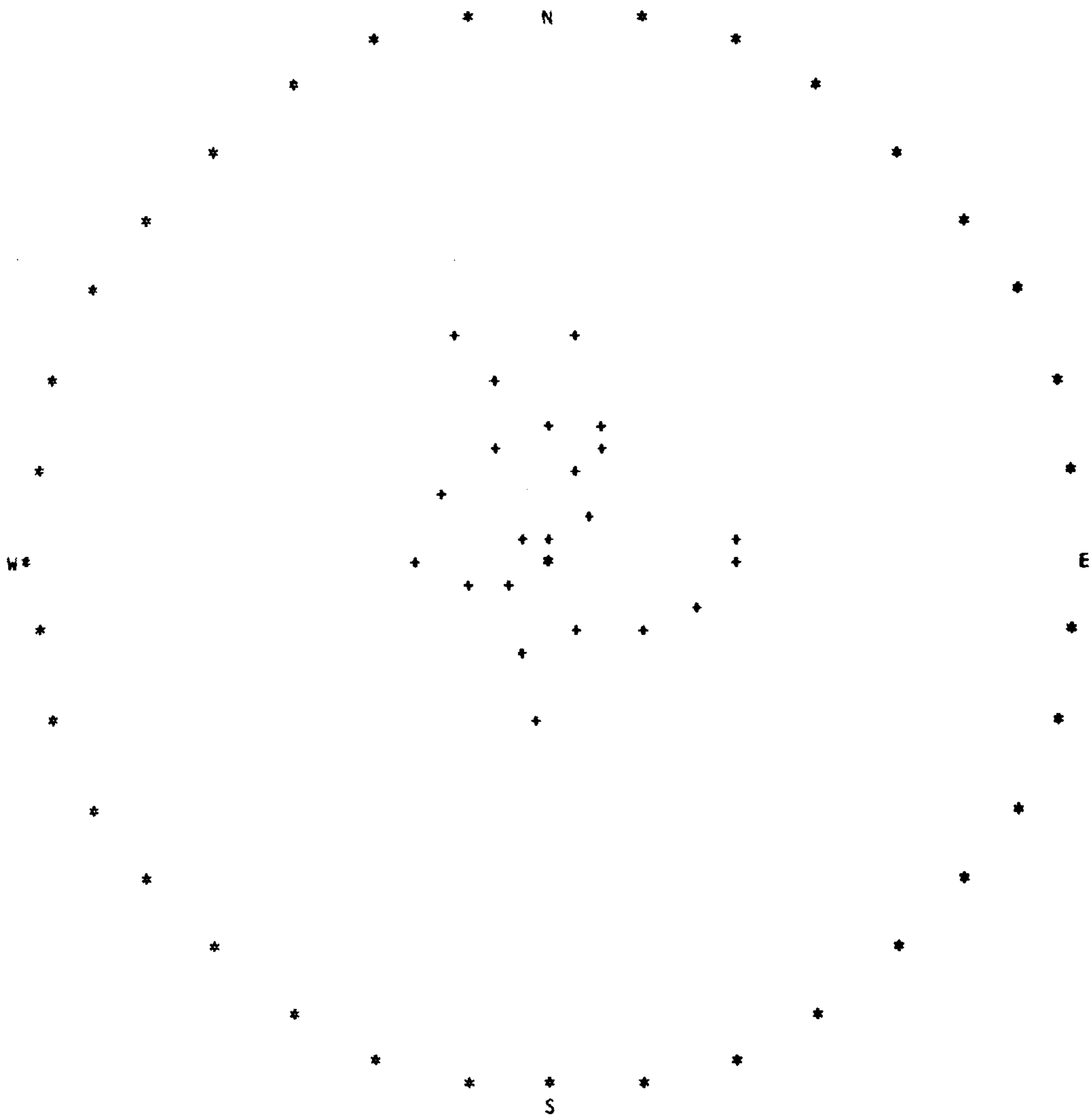
NUM	AZIMUTH	INCLINATION
1	6.00	36.00
2	335.00	37.00
3	86.00	28.00
4	94.00	29.00
5	187.00	28.00
6	341.00	30.00
7	23.00	21.00
8	26.00	20.00
9	113.00	24.00
10	257.00	22.00
11	297.00	21.00
12	356.00	21.00
13	131.00	19.00
14	201.00	17.00
15	324.00	19.00
16	14.00	14.00
17	19.00	15.00
18	42.00	9.00
19	163.00	15.00
20	230.00	10.00
21	248.00	15.00
22	302.00	8.00
23	336.00	5.00
24	168.00	4.00
25	168.00	3.00
26	261.00	3.00

MEAN
AZIMUTH 6.13 DEG
INCLINATION 4.70 DEG

RESULTANT AND ITS COMPONENTS - R,SX,SY,SZ.
24.44680 1.99222 .21404 24.36455

CENTER OF GRAVITY - (X,Y,Z) COORDINATES
.0766 .0082 .9371

UPPER HEMISPHERE



CLUSTER NO. 7 CONTAINS 26 POINTS.
FOR THIS ANALYSIS, RDIS = 13.50

OBJECTIVE FUNCTION = 44.489

TIME REQUIRED TO ANALYZE 144 DATA POINTS 6.729 SECONDS

SUMMARY OF RESULTS

RDIS	ISPHER	LANS	PCTSURF	RANDOMNESS TEST	NO OF CLUSTERS	OBJECTIVE FUNCTION	TIME(SECONDS) REQUIRED
12.5	0	0	2.37	7	6	3.730E+01	8.584
13.5	0	0	2.76	7	7	4.449E+01	6.729

REFERENCES

1. Mahtab, M. A., D. D. Bolstad, J. R. Alldredge, and R. J. Shanley. Analysis of Fracture Orientations for Input to Structural Models of Discontinuous Rock. BuMines RI 7669, 1972, 76 pp.

2. Wishart, D. Mode Analysis: A Generalization of Nearest Neighbour Which Reduces Chaining Effects. Colloquium in Numerical Taxonomy, Computing Laboratory, St. Andrews, United Kingdom, 1968, pp. 282-309.

3. Jeran, P. W., and J. R. Mashey. A Computer Program for the Stereographic Analysis of Coal Fractures and Cleats. BuMines IC 8454, 1970, 34 pp.

4. Wishart, D. Fortran II Programs for 8 Methods of Cluster Analysis (Clustan II). Computer Contribution 38, State Geological Survey, The University of Kansas, Lawrence, Kans., 1969, pp. 85-87.

U.S. GOVERNMENT PRINTING OFFICE: 1974— 705-098:17

INT.-BU.OF MINES,PGH.,PA. 18907