FINAL DESIGN SPECIFICATION
FOR
LARSYS/ISOCLS MODIFICATION FOR LACIE PHASE III

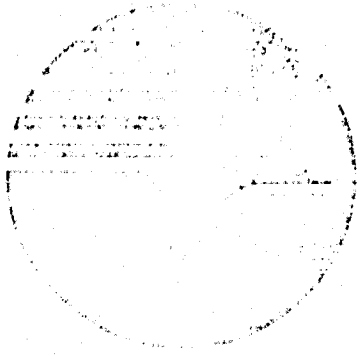
Job Order 71-695

(TI" 76-0086)

(E80-10212) FINAL DESIGN SPECIFICATION FOR LARSYS/ISOCLS MODIFICATION FOR LACIE PHASE 3 (Lockheed Electronics Co.) 70 p HC A04/MF A01 CSCL 05B G3/43

N80-27769
Unclas
00212

Prepared By
Lockheed Electronics Company, Inc.
Aerospace Systems Division
Houston, Texas
Contract NAS 9-12200
For
EARTH OBSERVATIONS DIVISION



National Aeronautics and Space Administration
LYNDON B. JOHNSON SPACE CENTER
Houston, Texas
February 1977

LEC-10161

FINAL DESIGN SPECIFICATION
FOR
LARSYS/ISOCLS MODIFICATION FOR LACIE PHASE III

Job Order 71-695

(TIRF 76-0086)

PREPARED

B. E. Wills

APPROVED BY

Philip L. Krumm, Supervisor
Applications Software Section

Prepared By
Lockheed Electronics Company, Inc.
For
Earth Observations Division

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION
LYNDON B. JOHNSON SPACE CENTER
HOUSTON, TEXAS
February 1977

LEC-10161

CONTENTS

1. SCOPE

1.1 GENERAL

This specification establishes the final design of modifications to the ISOCLS program within the EOD-LARSYS system, as specified in the Requirements Document REF: 642-2104.

2. APPLICABLE DOCUMENTS

The following documents, of exact issue shown, form a part of this specification to the extent specified herein.

- Action Document: 63-1327-1695-06
- Requirements Specification: REF: 642-2104
- TIRF 76-0086

3. SYSTEM DESCRIPTION

3.1 HARDWARE DESCRIPTION

N/A

3.2 SOFTWARE DESCRIPTION

The purpose of this modification is to:

a. Develop a ranking capability for cluster splitting and cluster merging. In cluster splitting, the standard deviations will be ranked in descending order to insure that the clusters having the largest of standard deviations greater than STDMAX will be given the highest priority for splitting. In cluster merging, the inter-cluster distances will be ranked in ascending order to insure that the clusters having the smallest of inter-cluster distances less than DLMIN will be given the highest priority for combining.

b. Compute and print out an Error Criterion E after each iteration.

c. Develop a cluster deletion procedure after the final iteration. The procedure is as follows: After the final or statistics pass P_1, there will be a test on the population of each cluster. If the population of any cluster falls below a user specified level, PMIN, plus the number of channels, the cluster(s) will be eliminated. There will be a second pass P_2 to reassign the pixels of the eliminated cluster(s) to the cluster centers of P_1. The statistics pass will be P_2. If there are no clusters deleted after P_1, the statistics pass will be P_1.

d. Simplify the coding logical in ISODAT. To accomplish this requirement, only one procedure for clustering will be developed and maintained. The simulated ERIPS procedure will be maintained, and the coding for the standard LARSYS procedure will be extracted from ISODAT and SETUP7. The control card OPTION SIMERP will no longer be a valid control card.

e. Expand the input capabilities of the clustering processor to optionally access mean values from a saved statistic file (cards, drum, or tape file). These mean values will serve as starting vectors for the cluster processor.

The following control card inputs are required for this modification:

- Cluster Ranking During Splitting and Combining
  No new control card input.

- Error Computation
  Control card - OPTION ERCOMP
  ERCOMP is a key that triggers the computation of

$$E = \left[\frac{1}{M} \sum_{K=1}^{LNCAT} DN(K) \sum_{J=1}^{NOFEAT} \sigma_j^2 (K) \right]^{\frac{1}{2}}$$

where DN(K) is the number of pixels in cluster K,
LNCAT is the number of clusters,
NOFEAT is the number of channels used in clustering,
$\sigma_j^2(K)$ is the variance of the *jth* channel of the *kth* cluster,
M is the total number of pixels being clustered,

and prints the value of E after each iteration.

- Cluster Deletion Procedure

  Control card - PMIN N

  PMIN is a user threshold value for cluster deletion. If the population of a cluster is less than PMIN plus the number of channels, the cluster is deleted after the final pass, P_1.

- Access Mean Values from a Saved Statistics File

  Control Cards - STATFILE INPUT/UNIT=N,FILE=M
  STATFILE OUTPUT/UNIT=N,FILE=M
  CHANNELS STAT=$N_1,\ldots,N_n$,DATA=$M_1,\ldots,M_n$
  MODULE
  SUBCLASSES $N_1,\ldots,N_n$

  with M and N being integer numbers.

  STATFILE INPUT/UNIT=N,FILE=M specifies the unit the input STAT file (starting vectors) is mounted on, and the number of the file to process. Defaults to self-starting vectors. If FILE is omitted, the default is File=1.

  STATFILE OUTPUT/UNIT=N,FILE=M specifies the unit the output STAT file is mounted on, and the number of the file to create. Default values are unit=1 and file=1.

  CHANNELS STAT=$N_1,\ldots,N_n$,DATA=$M_1,\ldots M_n$. $N_1,N_2,\ldots N_n$ are channel numbers referring to the input STAT file. $M_1,M_2,\ldots M_m$ are channel numbers referring to the MSS data tape. If STATFILE INPUT/UNIT=N,FILE=M control card is input, STAT set of channels must be defined.

MODULE triggers the input of the input Module STAT card deck. Default-No statistics card deck.

SUBCLASSES $N_1,\ldots,N_n$ are the subclass numbers referring to the input STAT file.

3.2.1 ISOCLS

3.2.1.1 Linkages

ISOCLS is called from MONTOR. ISOCLS calls LARSAA routines SETUP7, RDDATA, RDFILE, ISODAT, CHAIN, PRINT, DSTAPE, COVAR1, TWRITE, and PCHST1. UNIVAC system routines CMERR and RWRITE are also utilized.

3.2.1.2 Interfaces

N/A

3.2.1.3 Inputs

Input to the clustering processor ISOCLS consist of control cards, field/class definitions and imagery data tape. See SETUP7 description for control card input and RDDATA for imagery data input.

3.2.1.4 Outputs

The outputs of the ISOCLS processor are line printer clustering results, cluster map tape and a statistics file.

3.2.1.5 Storage Requirements

Storage used: Code-1017(8), Data-26475(8).

3.2.1.6 Description

ISOCLS performs a modified version of the clustering algorithm (ISODAT). The program expects multispectral scanner data in either the LARSYS 2 or Universal format.

3.2.1.7 Detailed Flowchart

N/A

3.2.1.8 Listing

See Appendix A for program listing.

3.2.2 SETUP7

3.2.2.1 Linkages

SETUP7 is called from ISOCLS. SETUP7 calls NXTCHR, NUMBER, FLTNUM, FIND and RDMEAN.

3.2.2.2 Interfaces

N/A

3.2.2.3 Inputs

The inputs to the SETUP7 subroutine which affect this specification are the control cards which include the keywords PMIN, STATFILE, CHANNELS, MODULE, SUBCLASSES and OPTION ERCOMP.

3.2.2.4 Outputs

The output of the SETUP7 subroutine is the input summary which includes cluster summary, cluster map, DAS output tape with cluster means (defaults) or cluster numbers.

3.2.2.5 Storage Requirements

Storage used: Code-1222(8), Data-547(8).

3.2.2.6 Description

SETUP7 reads and analyzes all parameter values and sets defaults for all control card input for the ISOCLS processor.

3.2.2.7 Detailed Flowcharts

N/A

3.2.2.8 Listing

See Appendix B for program listing.

3.2.3 ISODAT

3.2.3.1 Linkages

ISODAT is called from ISOCLS. ISODAT calls UNIVAC system routines RREAD, RWRITE and CLOCK. LARSAA routines CLDIST and PRINT are also called.

3.2.3.2 Interfaces

N/A

3.2.3.3 Inputs

The inputs to the ISODAT subroutine which affect this modification are parameter values of keywords PMIN, STATFILE, CHANNELS, SUBCLASSES, MODULE and OPTION ERCOMP.

3.2.3.4 Outputs

The outputs of the ISODAT subroutine are the line printer cluster summaries at each iteration.

3.2.3.5 Storage Requirements

Storage used: Code-2010(8), Data-614(8).

3.2.3.6 Description

All calculations of the clustering algorithms are done by ISODAT.

3.2.3.7 Detailed Flowcharts

N/A

3.2.4 GETSTA

3.2.4.1 Linkages

GETSTA is called from ISOCLS. GETSTA calls LARSAA routine EXIT and Univac routines FSBSFL and CMERR.

3.2.4.2 Interfaces

The module STAT file is the only interface between ISOCLS and other LARSAA processors. The file may be obtained from STAT or a previous execution of ISOCLS.

3.2.4.3 Inputs

The new parameters that were added to the subroutine GETSTA are SUBVEC - Subclasses from input statistics file for initial means, and NOSUB2 - Number of initial means.

3.2.4.4 Outputs

N/A

3.2.4.5 Storage Requirements

Storage used: Code-433(8), Data-1130(8).

3.2.4.6 Description

GETSTA extracts means from input MODULE STAT deck/file. The user may specify a subset of the channels and/or subclasses.

3.2.4.7 Detailed Flowcharts

N/A

3.2.4.8 Listing

See Appendix C for program listing.

3.2.5 DESCEN

3.2.5.1 Linkages

DESCEN is called from ISODAT.

3.2.5.2 Interfaces

N/A

3.2.5.3 Inputs

The arguments for the subroutine DESCEN are:

SCN - An output array dimensioned by LNCAT that contains values in descending order.

LNCAT - The current number of clusters.

IPTT - An array dimensioned by LNCAT. IPTT is used as a pointer to the reordered SCN array.

3.2.5.4 Outputs

N/A

3.2.5.5 Storage Requirements

N/A

3.2.5.6 Description

DESCEN ranks SCN values in descending order and IPTT is used as a pointer to these values.

3.2.5.7 Detailed Flowcharts

N/A

3.2.5.8 Listing

See Appendix D for program listing.

3.2.6 PSPLIT

3.2.6.1 Linkages

PSPLIT is called from ISODAT.

3.2.6.2 Interfaces

N/A

3.2.6.3 Inputs

The arguments for the subroutine PSPLIT are:

MEANS - An input array containing the means of the last NMIN iteration.

STDEV - An input/output array containing the standard deviations.

N - An input/output array containing the number of points in each cluster.

CLD - An input/output array containing the intercluster distances.

C - An input array for storing the imagery data.

IPLACE - An input array for storing the cluster number each corresponding pixels belongs to.

AVP - A temporary array for computing the means.

MEN - An output array containing the means of the PMIN iteration.

3.2.6.4 Outputs

N/A

3.2.6.5 Storage Requirements

Requires 577_8 words of core.

3.2.6.6 Description

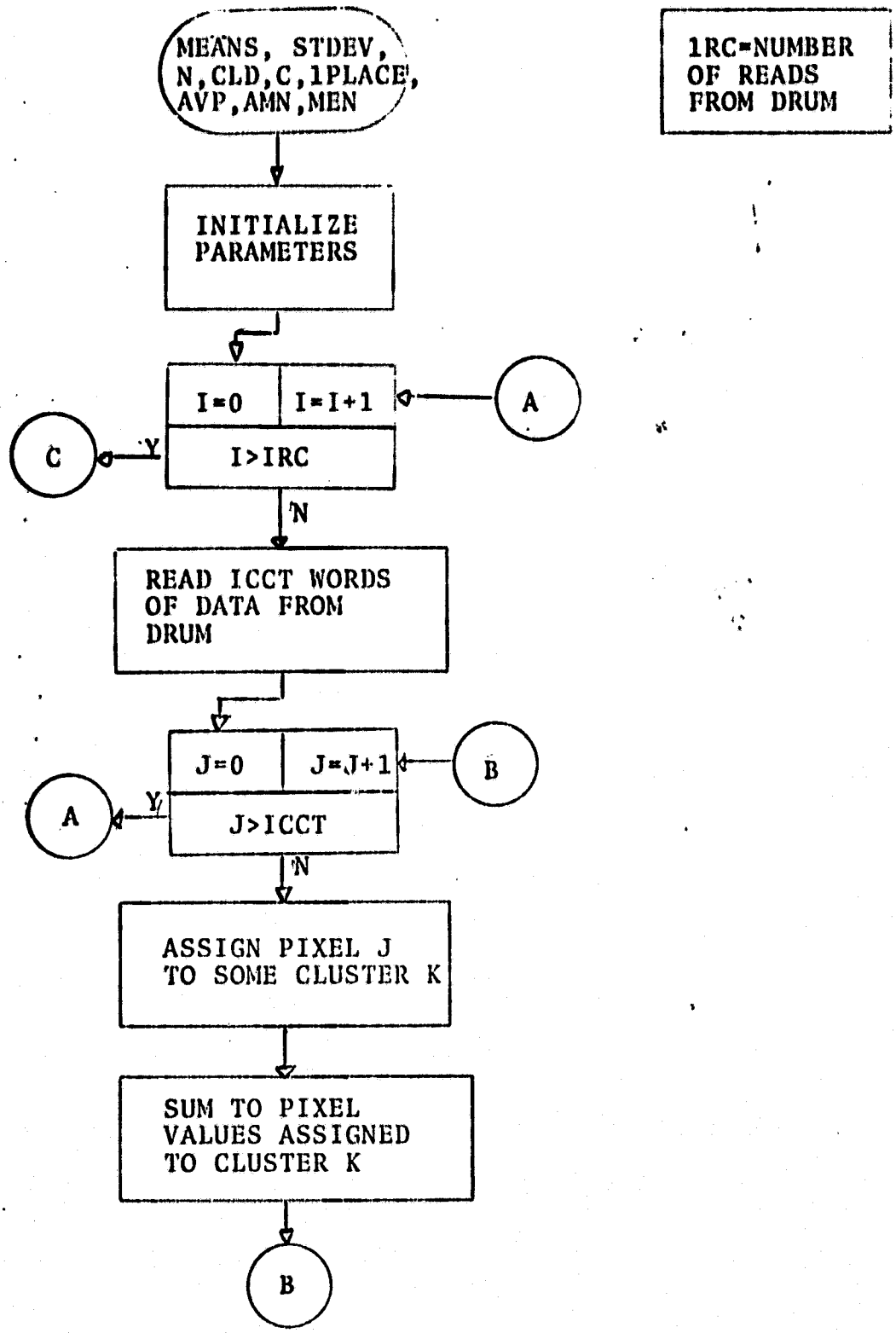
If all clusters have a population greater than the parameter PMIN, PSPLIT returns to the calling routine without any reassignment of the data; otherwise, all clusters with less than PMIN points are deleted, and the data reassigned to the remaining clusters.

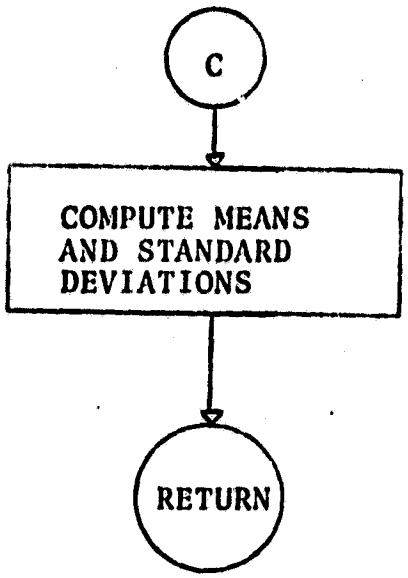
3.2.6.7 Detailed Flowchart

See following page.

3.2.6.8 Listing

See Appendix E for program listing.




C
COMPUTE MEANS AND STANDARD DEVIATIONS
RETURN

3.2.7 ISODAT

3.2.7.1 Linkages

ISODAT is called by ISOCLS and calls the routines CLDIST, PRINT, PSPLIT, and DESCEN.

3.2.7.2 Interfaces

N/A

3.2.7.3 Inputs

N/A

3.2.7.4 Outputs

N/A

3.2.7.5 Storage Requirements

Requires 30476_8 words of core.

3.2.7.6 Description

ISODAT performs the clustering. The routine was modified to rank the standard deviations in descending order to insure that the clusters having the largest of standard deviations would be given the highest priority for splitting. The inter-cluster distances are ranked in ascending order to insure that the clusters having the smallest of intercluster distances would be given highest priority for combining.

3.2.7.7 Detailed Flowchart

N/A

3.2.7.8 Listing

See Appendix F for program listing.

4. OPERATION

The EOD-LARSYS system is operational on the Univac 1108 EXEC2. To exercise the capabilities implemented as described in this document, see the User's Documentation EOD-LARSYS and list of added control cards in section 3.2.

5. TEST PROCEDURE

5.1 GENERAL

Using a 16 channel tape for checkout, vertification that the modifications to EOD-LARSYS system were correct was substantiated in the following manner:

a. The output of the cluster summary showed that the clusters having the largest of standard deviations greater than STDMAX were split and that the clusters having the smallest of intercluster distances less than DLMIN were combined.

b. The error criterion E was printed after each iteration.

c. The output of the final cluster summary showed that after the statistics pass, there were no clusters with a population less than the value of PMIN plus the number of channels.

d. The means from the statistics tape created by the ISOCLS processor were used as starting vectors for clustering. This vertified the expansion of the input capabilities of the clustering processor to optionally access mean values from a saved statistics file.

5.2 COMPUTER RUN

To further vertify that the modifications were implemented as outlined by the Requirements Specification, Ref: 642-2104, the same set of data was clustered on both ERIPS and LARSYS. The results are shown on the following pages.

LARSYS RUN

SISOCLS

INPUT SUMMARY

```
SEP       1.0
OPTION    ERCOMP
CHANNE    DATA=13,14,15,16
CLASSE    5
NMIN      20
PERCEN    90
DLMIN     3.6
STDMAX    3.6
SEQUEN    SC
ISTOP     12
•END•
```


YOU HAVE SELECTED THE FOLLOWING PARAMETER VALUES AND OPTIONS

```
STOP AFTER   12 ITERATION(S)
ALLOW A MINIMUM OF   20 PIXELS PER CLUSTER
PRINT A CLUSTER SUMMARY EVERY   20 ITERATION(S)
PRINT A CLUSTER MAP EVERY   20 ITERATION(S)
ALLOW A MAXIMUM OF   60 CLUSTERS PER CLASS
THE STATISTICS FILE WILL BE WRITTEN AFTER   5 CLASS(ES) HAVE BEEN CLUSTERED
CHANNELS ARE--- 13 14 15 16
DLMIN =  3.600
STDMAX=  3.600
SEP   =  1.000
```

INPUT IMAGE DATA TAPE INFORMATION

```
FORMAT                  UNIVERSAL
NO. OF CHANNELS           20
NO. OF PIXELS/LINE       196
FIRST SCAN LINE NO.        1
FIRST PIXEL REFERENCE PT   1
```

LYNDON B. JOHNSON SPACE CENTER
HOUSTON, TEXAS

22 JAN 77

FIELDS TO BE CLUSTERED FOR CLASS WHEAT1

	FIELD NAME	SAMPLE INC.	LINE INC.	VERTICES (SAMPLE,LINE)
1	ALL-F	1	1	(1, 1) (196, 1) (196, 117) (1, 117)

LYNDON B. JOHNSON SPACE CENTER
HOUSTON, TEXAS

22 JAN 77

FINAL CLUSTER SUMMARY FOR CLASS WHEAT1

TOTAL NUMBER OF CLUSTERS = 13

TOTAL NUMBER OF POINTS = 22932

CLUSTER	SYMBOL	POINTS IN CLUSTER
1	1	479
2	2	277
3	3	4231
4	4	6501
5	5	448
6	6	2424
7	7	1295
8	8	1449
9	9	2668
10	A	891
11	B	567
12	C	628
13	D	1079

MEANS

CLUSTER	CH(13)	CH(14)	CH(15)	CH(16)
1	45.30	54.60	55.72	26.14
2	29.26	20.19	61.90	34.53
3	32.90	32.78	36.00	17.12
4	28.72	26.07	29.94	14.03
5	36.12	39.62	50.73	25.11
6	29.36	25.87	37.14	18.44
7	39.04	44.38	46.04	21.69
8	31.23	27.73	44.37	22.84
9	36.04	32.11	41.95	19.99
10	41.95	48.83	51.07	24.28
11	26.59	21.93	50.53	27.87
12	34.68	32.39	50.05	25.54
13	27.23	20.75	43.02	23.44

Standard Deviations

CLUSTER	CH(13)	CH(14)	CH(15)	CH(16)
1	1.77	2.49	2.57	1.29
2	2.36	3.42	4.51	3.00
3	1.52	2.16	2.56	1.50
4	1.70	2.60	2.70	1.83
5	1.49	2.33	3.35	1.99
6	1.48	2.37	2.17	1.48
7	1.35	2.04	1.69	1.19
8	1.47	2.44	2.10	1.69
9	1.38	2.48	2.20	1.43
10	1.39	1.98	2.22	1.33
11	2.22	3.21	2.66	1.74
12	1.49	2.36	3.35	2.11
13	1.43	2.20	2.35	1.84

DISTANCES BETWEEN CLUSTERS

CLUSTER	1	2	3	4	5	6	7	8	9	10	11	12	13
1	.00	14.90	15.71	19.36	7.78	18.16	8.24	14.95	11.54	4.14	14.48	11.49	19.19
2	14.90	.00	12.27	12.82	9.69	11.18	13.71	8.21	11.69	14.44	4.43	7.01	7.56
3	15.71	12.27	.00	4.86	8.28	3.99	9.08	5.65	4.48	12.84	9.88	6.64	8.25
4	19.36	12.82	4.86	.00	12.15	4.03	13.73	8.04	9.12	17.11	11.00	10.00	7.69
5	7.78	9.69	8.28	12.15	.00	10.52	3.56	7.39	4.72	5.00	8.51	3.96	11.63
6	18.16	11.18	3.99	4.03	10.52	.00	11.98	4.62	7.30	15.73	8.22	7.74	4.81
7	8.24	13.71	9.08	13.73	3.56	11.98	.00	9.34	4.33	4.37	11.59	8.89	14.13
8	14.94	8.21	5.65	8.04	7.39	4.62	9.34	.00	5.81	12.68	4.67	4.00	4.15
9	11.54	11.69	4.48	9.12	4.72	7.30	4.33	5.81	.00	8.28	9.40	5.09	9.97
10	4.14	14.44	12.84	17.11	5.00	15.73	4.37	12.68	8.28	.00	13.32	9.22	17.42
11	14.48	4.43	9.88	11.00	8.51	8.22	11.59	4.67	9.40	13.32	.00	5.24	3.99
12	11.49	7.01	6.64	10.00	3.96	7.74	8.89	4.00	5.09	9.22	5.24	.00	7.78
13	19.19	7.56	8.25	7.69	11.63	4.81	14.13	4.15	9.97	17.42	3.99	7.78	.00

50 - 43

SISOCLS

INPUT SUMMARY

```
SEP         1.0
OPTION      ERCOMP
CHANNE      DATA=13,14,15,16
CLASSE      5
NMIN        20
PERCEN      90
DLMIN       3.6
STDMAX      3.6
SEQUEN      SC
ISTOP       12
•END•
```


26

YOU HAVE SELECTED THE FOLLOWING PARAMETER VALUES AND OPTIONS

```
STOP AFTER   12 ITERATION(S)
ALLOW A MINIMUM OF    20 PIXELS PER CLUSTER
PRINT A CLUSTER SUMMARY EVERY   20 ITERATION(S)
PRINT A CLUSTER MAP EVERY   20 ITERATION(S)
ALLOW A MAXIMUM OF   60 CLUSTERS PER CLASS
THE STATISTICS FILE WILL BE WRITTEN AFTER   5 CLASS(ES) HAVE BEEN CLUSTERED
CHANNELS ARE--- 13 14 15 16
DLMIN =  3.600
STDMAX=  3.600
SEP   =  1.000
```

INPUT IMAGE DATA TAPE INFORMATION

```
FORMAT                    UNIVERSAL
NO. OF CHANNELS              20
NO. OF PIXELS/LINE          196
FIRST SCAN LINE NO.           1
FIRST PIXEL REFERENCE PT      1
```

LYNDON B. JOHNSON SPACE CENTER
HOUSTON, TEXAS

22 JAN 77

FIELDS TO BE CLUSTERED FOR CLASS WHEAT:

	FIELD NAME	SAMPLE INC.	LINE INC.	VERTICES (SAMPLE, LINE)			
1	ALL-F	1	1	(1, 1)	(196, 1)	(196, 117)	(1, 117)

21

LYNDON B. JOHNSON SPACE CENTER
HOUSTON, TEXAS

22 JAN 77

FINAL CLUSTER SUMMARY FOR CLASS WHEAT1

TOTAL NUMBER OF CLUSTERS = 13

TOTAL NUMBER OF POINTS = 22932

CLUSTER	SYMBOL	POINTS IN CLUSTER
1	1	479
2	2	277
3	3	9231
4	4	6501
5	5	448
6	6	2929
7	7	1295
8	8	1449
9	9	2668
10	A	891
11	B	567
12	C	628
13	D	1079

MEANS

CLUSTER	CH(13)	CH(14)	CH(15)	CH(16)
1	45.30	54.60	55.72	26.19
2	29.26	20.19	61.90	34.53
3	32.90	32.78	36.00	17.12
4	28.72	26.07	29.94	14.03
5	36.12	39.62	50.73	25.11
6	29.36	25.87	37.14	18.44
7	39.04	44.38	46.04	21.69
8	31.23	27.73	44.37	22.84
9	36.04	38.11	41.95	19.99
10	41.95	48.83	51.07	24.28
11	26.59	21.93	50.53	27.87
12	39.68	32.39	50.05	25.54
13	27.23	20.75	43.02	23.44

STANDARD DEVIATIONS

CLUSTER	CH(13)	CH(14)	CH(15)	CH(16)
1	1.77	2.49	2.57	1.29
2	2.36	3.42	4.51	3.00
3	1.52	2.16	2.56	1.50
4	1.70	2.60	2.70	1.83
5	1.49	2.33	3.35	1.99
6	1.48	2.37	2.17	1.48
7	1.35	2.04	1.89	1.19
8	1.47	2.44	2.10	1.69
9	1.38	2.48	2.20	1.43
10	1.39	1.98	2.22	1.33
11	2.22	3.21	2.66	1.74
12	1.44	2.36	3.35	2.11
13	1.43	2.20	2.35	1.84

DISTANCES BETWEEN CLUSTERS

CLUSTER	1	2	3	4	5	6	7	8	9	10	11	12	13
1	.00	14.90	15.71	19.36	7.78	18.16	8.24	14.94	11.54	4.14	14.48	11.45	19.19
2	14.90	.00	12.27	12.82	9.69	11.18	13.71	8.21	11.89	14.44	4.43	7.01	7.56
3	15.71	12.27	.00	4.86	8.28	3.99	9.08	5.65	4.98	12.84	9.88	6.84	8.25
4	19.36	12.82	4.86	.00	12.15	4.03	13.73	8.04	9.12	17.11	11.00	10.00	7.69
5	7.78	9.69	8.28	12.15	.00	10.52	3.56	7.39	4.72	5.00	8.51	3.66	11.63
6	18.16	11.18	3.99	4.03	10.52	.00	11.98	4.62	7.30	15.73	8.22	7.74	4.81
7	8.24	13.71	9.08	13.73	3.56	11.98	.00	9.34	4.33	4.37	11.59	8.89	14.13
8	14.94	8.21	5.65	8.04	7.39	4.62	9.34	.00	5.81	12.60	4.67	4.00	4.15
9	11.54	11.89	4.98	9.12	4.72	7.30	4.33	5.81	.00	8.28	9.40	5.09	9.97
10	4.14	14.44	12.84	17.11	5.00	15.73	4.37	12.60	8.28	.00	13.32	9.22	17.42
11	14.48	4.43	9.88	11.00	8.51	8.22	11.59	4.67	9.40	13.32	.00	5.24	3.99
12	11.49	7.01	6.84	10.00	3.66	7.74	8.89	4.00	5.09	9.22	5.24	.00	7.76
13	19.19	7.56	8.25	7.69	11.63	4.81	14.13	4.15	9.97	17.42	3.99	7.76	.00

56 43

EPIPS RUN

1633

* 04/16/33 1

ITERATIVE INITIALIZATION MENU

INPUT YOUR ITERATIVE PARAMETERS:

DEFAULT	N	R2	8191
CO	20	NMIN2	20
PERCENT	.9	ITMAX	12
SEP	1	P OF N	20
STDMAX	3.6	MATCH THRS	4
DLMIN	3.6	SPLIT/COMBINE SEQUENCE	SC

[ ] L1 MATCH (IF SELECTED THE MATCH THRS VALUE MUST BE AN L1 VALUE)

STARTING VECTORS:

[ ] SELF GENERATING VECTORS

[ ] SUBCLASS STATISTICS VECTORS

[ ] VECTORS FROM RUN NUMBER 4

[ ] EXTERNAL VECTORS

[ ] SPP SELF PRIMING VECTORS THRESHOLD 4

S 02
S 04
S 06
S 09
S 13
****
S 14
C 13

IMD
SCR
RET
EOF
EOT

5-8-25

ORIGINAL PAGE IS OF POOR QUALITY

1840

* 04/16/49 1

CLUSTERING REPORT MENU

CLUSTER	POPULATION	SUBCLASS MATCH	L2 DISTANCE	CLUSTER MATCH	L2 DISTANCE	DEFAULT NAME	THRESHOLD FLAG
1	474.0		0.00		0.00		
2	277.0	0	0.00		0.00		
3	6501.0		0.00		0.00		
4	4231.0		0.00		0.00		
5	440.0		0.00		0.00		
6	2424.0		0.00		0.00		
7	1449.0		0.00		0.00		
8	1295.0		0.00		0.00		
9	2668.0		0.00		0.00		
10	567.0		0.00		0.00		
11	891.0		0.00		0.00		
12	628.0		0.00		0.00		
13	1079.0		0.00		0.00		

****** 0 UNASSIGNED PIXELS

Sag 26

* 04/16/49 1

1340

CLUSTERING REPORT MENU

CLUSTER	POPULATION	SUBCLASS MATCH	L2 DISTANCE	CLUSTER MATCH	L2 DISTANCE	DEFAULT NAME	THRESHOLD FLAG
1	474.0		0.00		0.00		
2	277.0	0	0.00		0.00		
3	6581.0		0.00		0.00		
4	4231.0		0.00		0.00		
5	448.0		0.00		0.00		
6	2424.0		0.00		0.00		
7	1449.0		0.00		0.00		
8	1295.0		0.00		0.00		
9	2668.0		0.00		0.00		
10	567.0		0.00		0.00		
11	891.0		0.00		0.00		
12	628.0		0.00		0.00		
13	1079.0		0.00		0.00		
******	0 UNASSIGNED PIXELS						

Sept 26

1305

DETAILED CLUSTERING REPORT

RUN NO.	CLUSTER 1 C OF 0/ 0 S FROM 0		CLUSTER 2 C OF 0/ 0 S FROM 0		CLUSTER 3 C OF 0/ 0 S FROM 0		CLUSTER 4 C OF 0/ 0 S FROM 0		CLUSTER 5 C OF 0/ 0 S FROM 0	
CHANNELS	MEAN	STD	MEAN	STD	MEAN	STD	MEAN	STD	MEAN	STD
13	45.30	1.77	29.26	2.36	28.72	1.70	32.60	1.52	38.10	1.40
14	54.60	2.49	20.19	3.43	26.07	2.60	32.78	2.16	39.80	2.32
15	55.72	2.57	61.90	4.52	29.94	2.70	36.00	2.56	50.70	2.35
16	26.14	1.29	34.53	3.08	14.03	1.83	17.12	1.50	26.10	1.90
DIFF	0.95		1.41		0.50		0.90		0.35	
DISTANCE MEAN	6.46		18.15		6.60		6.30		7.10	
STDEV	0.00		0.00		0.00		0.00		0.00	

Scan 27

1305

DETAILED CLUSTERING REPORT

RUN NO. 1	CLUSTER 1 C OF 0/ 0 S FROM 0		CLUSTER 2 C OF 0/ 0 S FROM 0		CLUSTER 3 C OF 0/ 0 S FROM 0		CLUSTER 4 C OF 0/ 0 S FROM 0		CLUSTER 5 C OF 0/ 0 S FROM 0	
CHANNELS	MEAN	STD	MEAN	STD	MEAN	STD	MEAN	STD	MEAN	STD
13	45.30	1.77	29.26	2.36	28.72	1.78	32.50	1.52	38.12	1.59
14	54.60	2.49	20.19	3.43	26.07	2.60	32.78	2.16	39.01	2.82
15	55.72	2.57	61.90	4.52	29.94	2.78	36.00	2.56	48.78	3.38
16	26.14	1.29	34.53	3.00	14.03	1.83	17.12	1.50	25.11	1.90
DIFF	0.95		1.41		0.50		0.98		0.35	
DISTANCE MEAN	6.46		10.15		6.60		6.30		7.10	
STDEV	0.00		0.00		0.00		0.00		0.00	

S-10-27

1805

04/18/15 1

DETAILED CLUSTERING REPORT

RUN NO. 1	CLUSTER 6 C OF 0/ 0 S FROM 0		CLUSTER 7 C OF 0/ 0 S FROM 0		CLUSTER 8 C OF 0/ 0 S FROM 0		CLUSTER 9 C OF 0/ 0 S FROM 0		CLUSTER 10 C OF 0/ 0 S FROM 0	
CHANNELS	MEAN	STD	MEAN	STD	MEAN	STD	MEAN	STD	MEAN	STD
13	29.36	1.48	31.23	1.47	39.04	1.35	36.04	1.38	28.59	2.02
14	25.87	2.37	27.73	2.44	44.38	2.85	38.11	2.48	21.93	3.02
15	37.14	2.17	44.37	2.10	46.04	1.89	41.95	2.20	30.53	2.16
16	18.44	1.48	22.84	1.69	21.89	1.19	19.99	1.43	27.87	1.70
DIFF	1.08		1.08		1.08		0.96		0.28	
DISTANCE MEAN	6.32		6.22		5.16		6.02		7.01	
STDEV	0.00		0.00		0.04		0.00		0.00	

S-XX/28

* 04/18/15 1

1805

DETAILED CLUSTERING REPORT

RUN NO. 1	CLUSTER 6 C OF 0/ 0 S FROM 0		CLUSTER 7 C OF 0/ 0 S FROM 0		CLUSTER 8 C OF 0/ 0 S FROM 0		CLUSTER 9 C OF 0/ 0 S FROM 0		CLUSTER 10 C OF 0/ 0 S FROM 0	
CHANNELS	MEAN	STD	MEAN	STD	MEAN	STD	MEAN	STD	MEAN	STD
13	29.36	1.48	31.23	1.47	39.04	1.35	36.04	1.38	28.59	2.02
14	25.87	2.37	27.73	2.44	44.38	2.05	38.11	2.48	21.93	3.02
15	37.14	2.17	44.37	2.10	46.04	1.89	41.95	2.20	56.53	2.[illegible]
16	18.44	1.48	22.84	1.69	21.89	1.19	19.99	1.43	27.87	1.[illegible]
DIFF	1.08		1.00		1.08		0.96		0.28	
DISTANCE MEAN	6.32		6.22		5.16		6.02		7.21	
STDEV	0.00		0.00		0.00		0.00		0.[illegible]	

5-II 28

1805 * 84/18/21 1

DETAILED CLUSTERING REPORT

RUN NO.	CLUSTER 11 C OF 0/ 0 S FROM 0		CLUSTER 12 C OF 0/ 0 S FROM 0		CLUSTER 13 C OF 0/ 0 S FROM 0	
1						
CHANNELS	MEAN	STD	MEAN	STD	MEAN	STD
13	41.95	1.39	34.68	1.44	27.23	1.43
14	48.83	1.98	32.39	2.36	20.75	2.20
15	61.07	2.22	60.06	3.36	43.02	2.36
16	24.28	1.33	26.64	2.11	23.44	1.64
DIFF	0.67		0.44		1.23	
DISTANCE						
MEAN	5.30		7.41		6.46	
STDEV	0.00		0.00		0.00	

Sept 29

1805

DETAILED CLUSTERING REPORT

RUN NO.	CLUSTER 11 C OF 0/ 0 S FROM 0		CLUSTER 12 C OF 0/ 0 S FROM 0		CLUSTER 13 C OF 0/ 0 S FROM 0	
1						
CHANNELS	MEAN	STD	MEAN	STD	MEAN	STD
13	41.95	1.39	34.68	1.44	27.23	1.43
14	48.83	1.98	32.39	2.36	20.75	2.20
15	51.07	2.22	60.06	3.36	43.02	2.35
16	24.28	1.33	25.64	2.11	23.44	1.84
DIFF	0.67		0.44		1.23	
DISTANCE						
MEAN	5.30		7.41		6.46	
STDEV	0.00		0.00		0.00	

Sept 29

6. TEST APPROVAL SHEET

UNIVAC 1108 LARSYS/ISOCLS MODIFICATIONS
FOR
LACIE PHASE III

Job Order 71-695

(TIRF 76-0086)

Verification Test Specification

APPROVAL SHEET

TEST CONDUCTOR: _____ DATE: _____

COGNIZANT ENGINEER: B. Hills for J.C. Mantia DATE: 2/10/77

REQUIREMENTS: J.C. Mantia DATE: 2/10/77

USER: J.C. Mantia DATE: 2/10/77

NASA TECHNICAL MONITOR: M C Trichel DATE: 10 Feb 77

MAINTENANCE & OPERATIONS: _____ DATE: _____

LACIE QUALITY ASSURANCE: _____ DATE: _____

QUALITY ASSURANCE: W. Rowland DATE: 10 Feb 77

REMARKS: _____

6. TEST APPROVAL SHEET

UNIVAC 1108 LARSYS/ISOCLS MODIFICATIONS
FOR
LACIE PHASE III

Job Order 71-695

(TIRF 76-0086)

Verification Test Specification

APPROVAL SHEET

TEST CONDUCTOR: B. Hill DATE:

COGNIZANT ENGINEER: J. C. Hamilton DATE: 2/10/77

REQUIREMENTS: J. C. Mantia DATE: 2/10/77

USER: J. C. Mantia DATE: 2/10/77

NASA TECHNICAL MONITOR: M C Trichel DATE: 10 Feb 77

MAINTENANCE & OPERATIONS: DATE:

LACIE QUALITY ASSURANCE: DATE:

QUALITY ASSURANCE: W. Lenhart DATE: 10 Feb 77

REMARKS:

APPENDIX A

PROGRAM LISTING

APPENDIX A

PROGRAM LISTING

```
10 I 027313 KPLCE    0003 I 000012 KPTS     0003 I 000003 KRN      0000 I 027316 KV       0000 I 000000 KVAR
10 I 027301 LAST     0003 I 000001 LNCAT    0000 I 027323 LSTAT    0000 I 027315 LI       0003 I 000007 MAP
4 I 000100 MAPFIL    0003 I 000279 MAPFMT   0003 I 000275 MAPKEY   0004 I 000052 MAPTAP   0004 I 000064 MAPU
3 I 000136 MAXCLS    0000 I 027277 MAXDIM   0000 I 027303 MEANSI   0003 I 000125 NCLASS   0004 I 000076 NMSI
4 I 000075 NMSTUN    0003 I 000002 NMIN     0003 I 000064 NOCHAN   0003 I 000132 NOCL     0003 I 000126 NOCL
3 I 000135 NOFEAT    0004 I 000065 NOFILE   0003 I 000121 NOFLD    0003 I 000013 NOPTS    0003 I 000060 NOSE
3 I 000125 NOSUB2    0003 I 000134 NVRT     0003 I 000113 NXDS     0003 I 000134 NXTCLS   0000 I 027302 N1
4 I 000070 PAGSIZ    0003 I 000322 PERCEN   0003 I 000330 PMIN     0003 I 000014 PUNCH    0004 I 000059 SAVI
4 I 000077 SCTRUN    0003 I 000006 SEP      0003 I 000276 SEQUEN   0003 I 000323 SIMERP   0000 I 027314 SMI
3 I 000010 SPTRIG    0004 I 000074 STAFIL   0003 I 000272 STATKY   0000 I 027304 STDEVI   0003 I 000004 STOP
3 I 000000 STOP      0003 I 000331 SUBVEC   0003 I 000175 SYMMTX   0003 I 000130 TOTFLD   0003 I 000124 TOTF
3 I 000127 TOTSUB    0003 I 000131 TOTVRT   0003 I 000123 TOTWRD   0004 I 000061 TRFORM   0000 I 027305 TTOP
3 I 000271 VARSIZ
```

```
      1*          SUBROUTINE ISOCLS(ARRAY,TOP)                                          ISOC0000
      2*     C*                                                                         ISOC0010
      3*     C*.....ISOC0010
      4*     C*                                                                         ISOC0020
      5*     C*   THIS PROGRAM PERFORMS A MODIFIED VERSION OF THE CLUSTERING            ISOC0030
      6*     C*   ALGORITHM (ISODATA) ORIGINALLY DEVELOPED BY BALL AND HALL OF          ISOC0040
      7*     C*   STANFORD RESEARCH INSTITUTE.  THE ALGORITHM HAS BEEN MODIFIED         ISOC0050
      8*     C*   ON THE RECOMMENDATIONS OF ED KAN (LEC).                               ISOC0060
      9*     C*                                                                         ISOC0070
     10*     C*   THE PROGRAM EXPECTS MULTISPECTRAL SCANNER DATA                        ISOC0080
     11*     C*   IN EITHER THE LARSYS 22 OR THE UNIVERSAL                              ISOC0090
     12*     C*   FORMAT.  THE DATA TAPE SHOULD BE ASSIGNED TO FORTRAN UNIT 3.          ISOC0100
     13*     C*                                                                         ISOC0130
     14*     C*.....ISOC0140
     15*     C*                                                                         ISOC0150
     16*          INCLUDE COMBKS,LIST
     16*          PARAMETER MAXPOP=60,MXFET1=30
     16*          COMMON/PASS/STOP,LNCAT,NMIN,KRN,STDMAX,DLMIN,SEP,
     16*         *                  MAP,SPTRIG,   IRD,  KPTS, NOPTS, PUNCH,
     16*         *           ICHN,CHNTHS,ICHAIN(MAXPOP),NXDS,IBEGIN,BEGIN1,
     16*         *          BEGIN2,BEGIN3,CLSNAM,NOFLD,IPT,TOTWRD,TOTPTS,
     16*         *          NCLASS,NOCLS,TOTSUB,TOTFLD,TOTVRT,NOCL,NVRT
     16*         *          ,MATCLS,NOFEAT,MAXCLS,FETVEC(30),SYMMTX(60)
     16*         *,VARSIZ,STATKY,ISOKEY,MAPFMT,MAPKEY,SEQUEN(20),PERCEN,SIMERP
     16*         *  ,IORDER,INUNIT,INFILE,INITN,PMIN,SUBVEC(60),NOSUB2,CHNVC(30)      *..EN
     16*         *   ,NOCHAN,ERCOMP,NOSEQ                                             *..NE*
     16*          IMPLICIT INTEGER (A-X)                                              *..-I
     16*          END
     17*          INCLUDE COMNTS,LIST
     17*     C*
     17*     C    COMMON BLOCK *PASS* IS USED ONLY BY THE ISOCLS PROCESSOR.
     17*     C*
     17*     C*   ISOCLS USES THE RANDOM ACCESS DRUM FILE AS FOUR DISTINCT FILES.
     17*     C*   SEE DEFINITIONS OF IBEGIN,BEGIN1,BEGIN2,BEGIN3 BELOW
     17*     C*
     17*     C*   DEFINITIONS
     17*     C*
     17*     C*      ISTOP  - MAX. NO. OF ITERATIONS FOR THE CLUSTERING PROCEDURE
     17*     C*               SET IN SETUP7 ROUTINE. (USER INPUT)
     17*     C*      LNCAT  - CURRENT NO. OF CLUSTERS. SET INITIALLY IN RDFILE OR ISOCLS
     17*     C*               ISOCLS, THEN ONLY IN ISODAT.
```

A-I 22

```
10 I 027313 KPLCE    0003 I 000012 KPTS     0003 I 000003 KRN      0000 I 027316 KV       0000 I 000000 KVAR
10 I 027301 LAST     0003 I 000001 LNCAT    0000 I 027323 LSTAT    0000 I 027315 LI       0003 I 000007 MAP
 4 I 000100 MAPFIL   0003 I 000274 MAPFMT   0003 I 000275 MAPKEY   0004 I 000052 MAPTAP   0004 I 000064 MAPU
 3 I 000136 MAXCLS   0000 I 027277 MAXDIM   0000 I 027303 MEANSI   0003 I 000125 NCLASS   0004 I 000076 NMSI
 4 I 000075 NMSTUN   0003 I 000002 NMIN     0003 I 000464 NOCHAN   0003 I 000132 NOCL     0003 I 000126 NOCU
 3 I 000135 NOFEAT   0004 I 000065 NOFILE   0003 I 000121 NOFLD    0003 I 000013 NOPTS    0003 I 000466 NOSE
 3 I 000425 NOSUB2   0003 I 000133 NVRT     0003 I 000113 NRDS     0003 I 000134 NXTCLS   0000 I 027302 NI
 4 I 000070 PAGSIZ   0003 I 000322 PERCEN   0003 I 000330 PMIN     0003 I 000014 PUNCH    0004 I 000054 SAVT
 4 I 000077 SCTRUN   0003 I 000006 SEP      0003 I 000276 SEQUEN   0003 I 000323 SIMERP   0000 I 027314 SMI
 3 I 000010 SPTRIG   0004 I 000074 STAFIL   0003 I 000272 STATKY   0000 I 027304 STDEVI   0003 I 000004 STOP
 3 I 000000 STOP     0003 I 000331 SUBVEC   0003 I 000175 SYMMTX   0003 I 000130 TOTFLD   0003 I 000124 TOTF
 3 I 000127 TOTSUB   0003 I 000131 TOTVRT   0003 I 000123 TOTWRD   0004 I 000061 TRFORM   0000 I 027305 TTOF
 3 I 000271 VARSIZ
```

```
      SUBROUTINE ISOCLS(ARRAY,TOP)                                          ISOC0000
C     ******************************************************************    ISOC0010
C                                                                           ISOC0020
C     THIS PROGRAM PERFORMS A MODIFIED VERSION OF THE CLUSTERING            ISOC0030
C     ALGORITM (ISODATA) ORIGINALLY DEVELOPED BY BALL AND HALL OF           ISOC0040
C     STANFORD RESEARCH INSTITUTE.  THE ALGORITHM HAS BEEN MODIFIED         ISOC0050
C     ON THE RECOMMENDATIONS OF ED KAN (LEC).                               ISOC0060
C                                                                           ISOC0070
C     THE PROGRAM EXPECTS MULTISPECTRAL SCANNER DATA                        ISOC0080
C     IN EITHER THE LARSYS 22 OR THE UNIVERSAL                              ISOC0090
C     FORMAT.  THE DATA TAPE SHOULD BE ASSIGNED TO FORTRAN UNIT 3.          ISOC0100
C                                                                           ISOC0130
C     ******************************************************************    ISOC0140
C                                                                           ISOC0150
      INCLUDE COMBK5,LIST
      PARAMETER MAXPOP=60,MXFET1=30
      COMMON/PASS/STOP,LNCAT,NMIN,KRN,STDMAX,DLMIN,SEP,
     *                    MAP,SPTRIG,   IRD,  KPTS, NOPTS, PUNCH,
     *           ICHN,CHNTHS,ICHAIN(MAXPOP),NRDS,IBEGIN,BEGIN1,
     *          BEGIN2,BEGIN3,CLSMAN,NOFLD,IPT,TOTRRD,TOTPTS,
     *          NCLASS,NOCLS,TOTSUB,TOTFLD,TOTVRT,NOCL,NVRT
     *          ,NXTCLS,NOFEAT,MAXCLS,FETVEC(30),SYMMTX(60)
     *,VARSIZ,STATKY,ISOKEY,MAPFMT,MAPKEY,SEQUEN(20),PERCEN,SIMERP
     *  ,IORDER,INUNIT,INFILE,INITM,PMIN,SUBVEC(60),NOSUB2,CHNVC(30)
     *   ,NOCHAN,ERCOMP,NOSEQ
      IMPLICIT INTEGER (A-X)
      END
      INCLUDE COMNTS,LIST
C
C     COMMON BLOCK 'PASS' IS USED ONLY BY THE ISOCLS PROCESSOR.
C
C     ISOCLS USES THE RANDOM ACCESS DRUM FILE AS FOUR DISTINCT FILES.
C     SEE DEFINITIONS OF IBEGIN,BEGIN1,BEGIN2,BEGIN3 BELOW
C
C     DEFINITIONS
C
C        ISTOP  - MAX. NO. OF ITERATIONS FOR THE CLUSTERING PROCEDURE
C                 SET IN SETUP7 ROUTINE. (USER INPUT)
C        LNCAT  - CURRENT NO. OF CLUSTERS. SET INITIALLY IN RDFILE OR ISOCLS
C                 ISOCLS, THEN ONLY IN ISODAT.
```

A-I 32

```
170    C0      NMIN   - MIN. NO. OF POINTS TO ALLOW PER CLUSTER
170    C0             SET IN SETUP7 ROUTINE. (USER INPUT)
170    C0      KMN    - PRINT CLUSTER SUMMARY EVERY 'KMN' ITERATION(S)
170    C0             SET IN SETUP7 ROUTINE. (USER INPUT)
170    C0      STDMAX - STANDARD DEVIATION FOR SPLITTING CLUSTERS
170    C0             SET IN SETUP7 ROUTINE. (USER INPUT)
170    C0      DLMIN  - MIN. DISTANCE BETWEEN CLUSTERS FOR COMBINING.
170    C0      SEP    - DISTANCE TO SEPARATE CLUSTERS.  SET EITHER IN SETUP7.
170    C0             BY USER INPUT, OR IN ID
170    C0             BY USER INPUT, OR IN ISODAT.
170    C0      MAP    - PRINT A CLUSTER MAP EVERY 'MAP' ITERATION(S)- SETUP7
170    C0      SPTRIG - TRIGGER TELLING WHETHER OR NOT 'SEP' WAS INPUT. -SETUP7
170    C0      IRD    - NO. OF RECORDS TO READ FROM DATA FILE. COMPUTED IN
170    C0             ISUCLS
170    C0      NUPTS  - NO. OF POINTS IN EACH RECORD. COMPUTER IN ISUCLS
170    C0      KPTS   - NO. OF POINTS IN LAST RECORD. COMPUTER IN ISUCLS
170    C0      PUNCH  - TRIGGER TELLING WHETHER OR NOT TO PUNCH THE MODULE
170    C0             STAT DECK. - SETUP7
170    C0      ICHN   - TRIGGER TELLING WHETHER OR NOT CHAINING IS TO BE DONE
170    C0      CHNTHS - MIN. DISTANCE BETWEEN CLUSTERS FOR CHAINING - SETUP7
170    C0      ICHAIN - ARRAY CONTAINING CHAINED CLUSTER NUMBERS. SET IN
170    C0             'CHAIN' ROUTINE.
170    C0      NWDS   - TOTAL NO. OF WORDS AVAILABLE FOR DRUM STORAGE OF
170    C0             IMAGE DATA TO BE CLUSTERED - SET IN ISUCLS
170    C0      IBEGIN - BEGINNING DRUM FILE ADDRESS FOR INPUT INITIAL CLUSTER
170    C0             CENTERS - SET IN ISUCLS
170    C0      BEGINS - BEGINNING DRUM FILE ADDRESS FOR TEMPORARY STORAGE OF
170    C0             CLASS STATISTICS - SET IN ISUCLS ROUTINE
170    C0      BEGIN1 - BEGINNING DRUM FILE ADDRESS FOR IMAGE DATA
170    C0      BEGIN2 - BEGINNING DRUM FILE ADDRESS FOR 'IPLACE' .(CLUSTER TO
170    C0             WHICH CORRESPONDING POINT BELONGS.)
170    C0      CLSNAM - NAME OF CLASS CURRENTLY BEING PROCESSED. - RDDATA
170    C0      NUFLD  - NO. OF FIELDS INPUT FOR THIS CLASS - RDDATA
170    C0      IPT    - NO. OF WORDS OF STORAGE USED IN 'ARRAY' FOR FIELD AND
170    C0             CLASS INFORMATION FOR THIS CLASS. - RDDATA
170    C0      TOTWRD - TOTOL WORDS WRITTEN ON DRUM FILE BEGINNING AT ADDRESS
170    C0             BEGIN1 - RDDATA
170    C0      TOTPTS - TOTOL POINTS TO BE CLUSTERED FOR CURRENT CLASS - RDDATA
170    C0      NCLASS - NO. OF CLASSES TO BE CLUSTERED FOR CURRENT CALL TO
170    C0             ISUCLS. USER INPUT - SETUP7.
170    C0      NOCLS  - CURRENT CLASS NO. - ISOCLS
170    C0      TOTSUB - TOTAL CLUSTERS FOR THIS CALL TO ISOCLS
170    C0      TOTFLD - TOTAL FIELDS FOR ALL CLASSES - ISOCLS
170    C0      TOTVRT - TOTAL VERTICES FOR ALL FIELDS - ISOCLS
170    C0      NUCL   - NO. OF CLASSES SINCE LAST CALL TO SETUP - RDDATA
170    C0      NVRT   - NO.OF VERTICES FOR ALL FIELDS IN CURRENT CLASS
170    C0      NXTCLS - NAME OF NEXT CLASS  (RDDATA)
170    C0      MAPFMT - FORMAT OF OUTPUT MAP TAPE
170    C0      NOFEAT - NO. OF CHANNELS USED IN CLUSTERING
170    C0      MAXCLS - MAX. NO. OF CLUSTERS PER CLASS
170    C0      FETVEC - CHANNELS USED IN CLUSTERING
170    C0      SYMMTA - SYMBOLS FOR PRINTING MAP
170    C0      VARSIZ - SIZE OF EACH CLUSTER'S COVARIANCE MATRIX
170    C0      STATKY - FLAG FOR PRINTING COVARIANCES.
170    C      MAPKEY-INDICATES WHETHER TO USE MEAN VECTOR OR CLUSTER NUMBER
170    C             IN MAPTAPE
170    C      SEQUEN-ARRAY CONTAINING SEQUENCE OF 'S' AND 'C' CHARACTERS FOR
170    C             SPLIT-COMBINE ITERATION CONTROL.
```

A-2 33

```
170*      C*      NMIN   - MIN. NO. OF POINTS TO ALLOW PER CLUSTER
170*      C*               SET IN SETUP7 ROUTINE. (USER INPUT)
170*      C*      KRN    - PRINT CLUSTER SUMMARY EVERY 'KRN' ITERATION(S)
170*      C*               SET IN SETUP7 ROUTINE. (USER INPUT)
170*      C*      STDMAX - STANDARD DEVIATION FOR SPLITTING CLUSTERS
170*      C*               SET IN SETUP7 ROUTINE. (USER INPUT)
170*      C*      DLMIN  - MIN. DISTANCE BETWEEN CLUSTERS FOR COMBINING.
170*      C*      SEP    - DISTANCE TO SEPARATE CLUSTERS.  SET EITHER IN SETUP7,
170*      C*               BY USER INPUT, OR IN ID
170*      C*               BY USER INPUT, OR IN ISODAT.
170*      C*      MAP    - PRINT A CLUSTER MAP EVERY 'MAP' ITERATION(S)- SETUP7
170*      C*      SPTRIG - TRIGGER TELLING WHETHER OR NOT 'SEP' WAS INPUT. -SETUP7
170*      C*      IRD    - NO. OF RECORDS TO READ FROM DATA FILE. COMPUTED IN
170*      C*               ISUCLS
170*      C*      NUPTS  - NO. OF POINTS IN EACH RECORD. COMPUTER IN ISUCLS
170*      C*      KPTS   - NO. OF POINTS IN LAST RECORD. COMPUTER IN ISUCLS
170*      C*      PUNCH  - TRIGGER TELLING WHETHER OR NOT TO PUNCH THE MODULE
170*      C*               STAT DECK. - SETUP7
170*      C*      ICHN   - TRIGGER TELLING WHETHER OR NOT CHAINING IS TO BE DONE
170*      C*      CHNTHS - MIN. DISTANCE BETWEEN CLUSTERS FOR CHAINING - SETUP7
170*      C*      ICHAIN - ARRAY CONTAINING CHAINED CLUSTER NUMBERS. SET IN
170*      C*               'CHAIN' ROUTINE.
170*      C*      NWDS   - TOTAL NO. OF WORDS AVAILABLE FOR DRUM STORAGE OF
170*      C*               IMAGE DATA TO BE CLUSTERED - SET IN ISUCLS
170*      C*      IBEGIN - BEGINNING DRUM FILE ADDRESS FOR INPUT INITIAL CLUSTER
170*      C*               CENTERS - SET IN ISUCLS
170*      C*      BEGINS - BEGINNING DRUM FILE ADDRESS FOR TEMPORARY STORAGE OF
170*      C*               CLASS STATISTICS - SET IN ISUCLS ROUTINE
170*      C*      BEGIN1 - BEGINNING DRUM FILE ADDRESS FOR IMAGE DATA
170*      C*      BEGIN2 - BEGINNING DRUM FILE ADDRESS FOR 'IPLACE' (CLUSTER TO
170*      C*               WHICH CORRESPONDING POINT BELONGS.)
170*      C*      CLSNAM - NAME OF CLASS CURRENTLY BEING PROCESSED. - RDDATA
170*      C*      NOFLD  - NO. OF FIELDS INPUT FOR THIS CLASS - RDDATA
170*      C*      IPT    - NO. OF WORDS OF STORAGE USED IN 'ARRAY' FOR FIELD AND
170*      C*               CLASS INFORMATION FOR THIS CLASS. - RDDATA
170*      C*      TOTWRD - TOTOL WORDS WRITTEN ON DRUM FILE BEGINNING AT ADDRESS
170*      C*               BEGIN1 - RDDATA
170*      C*      TOTPTS - TOTOL POINTS TO BE CLUSTERED FOR CURRENT CLASS - RDDATA
170*      C*      NCLASS - NO. OF CLASSES TO BE CLUSTERED FOR CURRENT CALL TO
170*      C*               ISUCLS. USER INPUT - SETUP7.
170*      C*      NOCLS  - CURRENT CLASS NO. - ISOCLS
170*      C*      TOTSUB - TOTAL CLUSTERS FOR THIS CALL TO ISOCLS
170*      C*      TOTFLD - TOTAL FIELDS FOR ALL CLASSES - ISOCLS
170*      C*      TOTVRT - TOTAL VERTICES FOR ALL FIELDS - ISOCLS
170*      C*      NUCL   - NO. OF CLASSES SINCE LAST CALL TO SETUP - RDDATA
170*      C*      NVRT   - NO.OF VERTICES FOR ALL FIELDS IN CURRENT CLASS
170*      C*      NXTCLS - NAME OF NEXT CLASS  (RDDATA)
170*      C*      MAPFMT - FORMAT OF OUTPUT MAP TAPE
170*      C*      NOFEAT - NO. OF CHANNELS USED IN CLUSTERING
170*      C*      MAXCLS - MAX. NO. OF CLUSTERS PER CLASS
170*      C*      FETVEC - CHANNELS USED IN CLUSTERING
170*      C*      SYMMTX - SYMBOLS FOR PRINTING MAP
170*      C*      VARSIZ - SIZE OF EACH CLUSTER'S COVARIANCE MATRIX
170*      C*      STATKY - FLAG FOR PRINTING COVARIANCES.
170*      C     MAPKEY-INDICATES WHETHER TO USE MEAN VECTOR OR CLUSTER NUMBER
170*      C            IN MAPTAPE
170*      C     SEQUEN-ARRAY CONTAINING SEQUENCE OF 'S' AND 'C' CHARACTERS FOR
170*      C            SPLIT-COMBINE ITERATION CONTROL.
```

A-2 33

```
C     PERCEN-CUT OFF PERCENTAGE FOR STABILIZED CLUSTERS IN INITIAL
C           SEQUENCE.
C     SIMERP-TRIGGER FOR SIMULATING ERIPS CLUSTERING ALGORITHM.
C           SIMERP=1   SIMULATE ERIPS-SIMERP=0     STANDARD ISOCLS
C     ORDER--TRIGGER FOR ORDERING THE COLOR KEYS BASE ON THE RANKING
C           OF GREENNESS OF EACH CLUSTER AS COMPUTED BY THE SUBROUTINE
C           RANK.SET IN SETUP7.
C.    INUNIT-UNIT NUMBER FOR INPUT STATISTICS FILE.                    .NEW
C.    INFILE-FILE NUMBER FOR INPUT STATISTICS FILE.                    .NEW
C.    INITM--TRIGGER FOR INITIAL MEANS FROM STATISTICS FILE.           .NEW
C.    PMIN---POPULATION MINIMUM FOR EACH CLUSTER.                      .NEW
C           NOFEAT.
C.
C.
      END
      INCLUDE COMBK6,LIST
      COMMON/GLOBAL/HEAD(42),MAPTAP,DATAPE,SAVTAP,BMFILE,BMKEY,
     .             HISFIL,HISKEY,TRFORM,ERIPTP,ERPKEY,MAPUNT,NOFILE,
     .    DRUMAD,DRMADS,PAGSIZ,DATFIL,STAFIL,ASAV,ASAVFL
     .  ,NMSTUN,NMSTFI,SCTRUN,MAPFIL                                   .NEW
      END                                                              .0-1
      PARAMETER KVARDM = 11500
      DIMENSION KVAR(KVARDM)
      DIMENSION ARRAY(1)
      DIMENSION COVAR(965)                                             .NEW
      IBEGIN=DRUMAD
C.
C.    RESERVE ENOUGH DRUM STORAGE FOR MAXIMUM INITIAL MEANS
C.
      BEGIN3=IBEGIN + MAXPOP*MXFET1 + MXFET1 + 2
C.                                                                     ISOC0330
C.    CALL SETUP TO READ CARD INPUT AND INITIALIZE DEFAULT VALUES      ISOC0340
C.                                                                     ISOC0350
      ITIME=1
      NOCLS = 0
      TOTFLD = 0
      TOTVRT = 0
      TOTSUB = 0
      CORBAS=1
    1 CALL SETUP7(ARRAY(CORBAS),TOP,ITIME)
      IF(ITIME.GT.1)GO TO 2
      VARSIZ=NOFEAT*(NOFEAT+1)/2
      BEGIN1 = BEGIN3 + NCLASS*MAXPOP*(VARSIZ + NOFEAT + 1)
      NADS=DRMADS-(BEGIN1-DRUMAD)
    2 ITIME=ITIME+1
      NOCL=0
C.                                                                     ISOC0370
C     CALL RDDATA  TO COORDINATE READING OF DATA
C.                                                                     ISOC0390
    5 MAXDIM = TOP-CORBAS
      FD1=CORBAS
      CALL RDDATA(ARRAY(FD1),MAXDIM,KVAR,KVARDM,LAST)
      BEGIN2=BEGIN1 + TOTBRD
      N1 = FD1 + IPT
      MEANS1=N1 + MAXCLS
      STDEV1=MEANS1 + MAXCLS*NOFEAT
      TTOP = STDEV1 + MAXCLS*NOFEAT
      MAXDIM=TOP-TTOP
```

A3 30

```
C     PERCEN-CUT OFF PERCENTAGE FOR STABILIZED CLUSTERS IN INITIAL
C           SEQUENCE.
C     SIMERP-TRIGGER FOR SIMULATING ERIPS CLUSTERING ALGORITHM.
C           SIMERP=1    SIMULATE ERIPS-SIMERP=0    STANDARD ISOCLS
C     ORDER--TRIGGER FOR ORDERING THE COLOR KEYS BASE ON THE RANKING
C           OF GREENNESS OF EACH CLUSTER AS COMPUTED BY THE SUBROUTINE
C           RANK.SET IN SETUP7.
C     INUNIT-UNIT NUMBER FOR INPUT STATISTICS FILE.                    *NEW
C     INFILE-FILE NUMBER FOR INPUT STATISTICS FILE.                    *NEW
C     INITM--TRIGGER FOR INITIAL MEANS FROM STATISTICS FILE.           *NEW
C     PMIN---POPULATION MINIMUM FOR EACH CLUSTER.                      *NEW
C           NOFEAT.                                                    *NEW
C
C
      END
      INCLUDE COMBK6,LIST
      COMMON/GLOBAL/HEAD(42),MAPTAP,DATAPE,SAVTAP,BMFILE,BMKEY,
     *      HISFIL,HISKEY,TRFORM,ERIPTP,ERPKEY,MAPUNT,NOFILE,
     *   DRUMAD,DRMWDS,PAGSIZ,DATFIL,STAFIL,ASAV,ASAVFL
     *  ,NHSTUN,NHSTFI,SCTRUN,MAPFIL                                   *NEW
      END                                                              *--1
      PARAMETER KVARDM = 11500
      DIMENSION KVAR(KVARDM)
      DIMENSION ARRAY(1)
      DIMENSION COVAR(465)                                             *NEW
      IBEGIN=DRUMAD
C
C     RESERVE ENOUGH DRUM STORAGE FOR MAXIMUM INITIAL MEANS
C
      BEGIN3=IBEGIN + MAXPOP*MXFET1 + MXFET1 + 2
C                                                                      ISOC0330
C     CALL SETUP TO READ CARD INPUT AND INITIALIZE DEFAULT VALUES      ISOC0340
C                                                                      ISOC0350
      ITIME=1
      NOCLS = 0
      TOTFLD = 0
      TOTVRT = 0
      TOTSUB = 0
      CORBAS=1
    1 CALL SETUP7(ARRAY(CORBAS),TOP,ITIME)
      IF(ITIME.GT.1)GO TO 2
      VARSIZ=NOFEAT*(NOFEAT+1)/2
      BEGIN1 = BEGIN3 + NCLASS*MAXPOP*(VARSIZ + NOFEAT + 1)
      NWDS=DRMWDS-(BEGIN1-DRUMAD)
    2 ITIME=ITIME+1
      NOCL=0
C                                                                      ISOC0370
C     CALL RDDATA  TO COORDINATE READING OF DATA                       ISOC0390
C
    5 MAXDIM = TOP-CORBAS
      FD1=CORBAS
      CALL RDDATA(ARRAY(FD1),MAXDIM,KVAR,KVARDM,LAST)
      BEGIN2=BEGIN1 + TOTWRD
      N1 = FD1 + IPT
      MEANS1=N1 + MAXCLS
      STDEV1=MEANS1 + MAXCLS*NOFEAT
      TTOP = STDEV1 + MAXCLS*NOFEAT
      MAXDIM=TOP-TTOP
```

A3 34

```
      NUPTS = MAXDIM/(NUFEAT+1)
      IDATI = TTOP
       IF(ISUKEY.EQ.1)GO TO 7
C      SUBVEC-SUBCLASSES FROM STATISTICS FILE FOR INTIAL MEANS.
C      NOSUB2-NUMBER OF INITIAL MEANS.
C      CHNVEC-NUMBER OF CHANNELS FROM STATISTICS FILE. NOCHAN MUST EQUAL
       IF(INITM.EQ.1)GO TO 6
       LNCAT=1
       GO TO 8
    6  LNCAT=NOSUB2
       CALL GETSTA(INUNIT,INFILE,ARRAY(MEANSI),DUM,NOSUB2,SUBVEC,NOCHAN
     . ,CHNVC,ARRAY(TTOP),COVAR,0)
       GO TO 8                                                    .NE.
    7  CONTINUE                                                   .EQ.1
       IF(ISOKEY.EQ.1)CALL RDFILE(ARRAY(MEANSI),ARRAY(TTOP))
    8  CONTINUE
       IF(NUPTS.GT.0)GOTO 10
       WRITE(6,100)MAXDIM
  100  FORMAT(' DIMENSION LIMITS EXCEEDED IN ISOCLS BY',I6,
     .   ' REDUCE CHANNELS OR MAX-CLUSTERS')
       CALL CMERR
   10  CONTINUE
       IRD=TOTPTS/NUPTS
       IF(MOD(TOTPTS,NUPTS).EQ.0)GO TO 20
       XPTS=MOD(TOTPTS,NUPTS)
       IRD=IRD+1
       IF(IRD.EQ.1)NOPTS=XPTS
       GO TO 25
   20  XPTS=NUPTS
   25  CONTINUE
C                                                                 ISOC0410
C      CALL ISODAT TO PERFORM CLUSTERING                          ISOC0420
C                                                                 ISOC0430
       A1=1
       A2=A1+ MAXCLS*NUFEAT
       CLDI=A2 + MAXCLS*NOFEAT
       KPLCE = NUPTS*NUFEAT + IDATI
       CALL ISODAT(ARRAY(IDATI),ARRAY(KPLCE),ARRAY(MEANSI),ARRAY(NI),
     .             ARRAY(STDEVI),KVAR (CLDI),ARRAY(FDI),KVAR(A1),
     .             KVAR(A2))
C                                                                 ISOC0550
C      CHAIN CLUSTERS WHOSE DISTANCES ARE LESS THAN DLMIN         ISOC0560
C                                                                 ISOC0570
       IF(ICHN.GT.0)CALL CHAIN(KVAR(CLDI))
C                                                                 ISOC0590
C      PRINT FINAL RESULTS                                        ISOC0600
C                                                                 ISOC0610
       CALL PRINT(-1,ARRAY(KPLCE),ARRAY(MEANSI),ARRAY(STDEVI),
     .            KVAR (CLDI),ARRAY(FDI),ARRAY(NI))
C                                                                 ISOC0630
C      CREATE MAP OUTPUT TAPE FOR PMIS DAS IF DESIRED
C
       IF(MAPFMT.GT.0)CALL DSTAPE(ARRAY(KPLCE),KVAR(1),ARRAY(MEANSI),
     .                            ARRAY(FDI))
C
C      CALCULATE COVARIANCE MATRIX FOR EACH CLUSTER
C
       IF(VARSIZ*LNCAT.GT.KVARDM)GO TO 30
```

Ang 35

```
      NUPTS = MAXDIM/(NUFEAT+1)
      IDAT1 = TTOP
       IF(ISUKEY.EQ.1)GO TO 7
C*     SUBVEC-SUBCLASSES FROM STATISTICS FILE FOR INTIAL MEANS.
C*     NOSUB2-NUMBER OF INITIAL MEANS,
C*     CHNVEC-NUMBER OF CHANNELS FROM STATISTICS FILE. NOCHAN MUST EQUAL
       IF(INITM.EQ.1)GO TO 6
       LNCAT=1
       GO TO 8
    6  LNCAT=NOSUB2
       CALL GETSTA(INUNIT,INFILE,ARRAY(MEANS1),DUM,NOSUB2,SUBVEC,NOCHAN
     *,CHNVC,ARRAY(TTOP),COVAR,0)                                    *NEW
       GO TO 8                                                       **-1
    7  CONTINUE
       IF(ISOKEY.EQ.1)CALL RDFILE(ARRAY(MEANS1),ARRAY(TTOP))
    8  CONTINUE
       IF(NUPTS.GT.0)GOTO 10
       WRITE(6,100)MAXDIM
  100 FORMAT(' DIMENSION LIMITS EXCEEDED IN ISOCLS BY',I6,
     *  ' REDUCE CHANNELS OR MAX.CLUSTERS')
       CALL CMERR
   10 CONTINUE
       IRD=TOTPTS/NUPTS
       IF(MOD(TOTPTS,NUPTS).EQ.0)GO TO 20
       KPTS=MOD(TOTPTS,NUPTS)
       IRD=IRD+1
       IF(IRD.EQ.1)NUPTS=KPTS
       GO TO 25
   20 NPTS=NUPTS
   25 CONTINUE
C*
C*     CALL ISODAT TO PERFORM CLUSTERING                              ISOC0410
C*                                                                    ISOC0420
C*                                                                    ISOC0430
       A1=1
       A2=A1+ MAXCLS*NUFEAT
       CLD1=A2 + MAXCLS*NOFEAT
       KPLCE = NUPTS*NOFEAT + IDAT1
       CALL ISODAT(ARRAY(IDAT1),ARRAY(KPLCE),ARRAY(MEANS1),ARRAY(N1),
     *             ARRAY(STDEV1),KVAR (CLD1),ARRAY(FD1),KVAR(A1),
     *             KVAR(A2))
C*                                                                    ISOC0550
C*     CHAIN CLUSTERS WHOSE DISTANCES ARE LESS THAN DLMIN             ISOC0560
C*                                                                    ISOC0570
       IF(ICHN.GT.0)CALL CHAIN(KVAR(CLD1))
C*                                                                    ISOC0590
C*     PRINT FINAL RESULTS                                            ISOC0600
C*                                                                    ISOC0610
       CALL PRINT(-1,ARRAY(KPLCE),ARRAY(MEANS1),ARRAY(STDEV1),
     *            KVAR (CLD1),ARRAY(FD1),ARRAY(N1))
C*                                                                    ISOC0630
C*     CREATE MAP OUTPUT TAPE FOR PMTS DAS IF DESIRED
C*
       IF(MAPFMT.GT.0)CALL DSTAPE(ARRAY(KPLCE),KVAR(1),ARRAY(MEANS1),
     *                            ARRAY(FD1))
C*
C*     CALCULATE COVARIANCE MATRIX FOR EACH CLUSTER
C*
       IF(VARSIZ*LNCAT.GT.KVARDM)GO TO 30
```

Aug 35

ORIGINAL PAGE IS
OF POOR QUALITY

```
      CALL COVARI(KVAR,ARRAY(IDAT),ARRAY(KPLCE),ARRAY(MEANS),
     *            ARRAY(NI))
C
C     CREATE ENIPS INTERFACE TAPE
C
      IF(ERPKEY.GT.0)CALL TWRITE(KVAR,ARRAY(MEANS),ARRAY(STDEV),
     *                  ARRAY(NI))
      TOTSUB = TOTSUB + LNCAT
      NOCLS = NOCLS + 1
      TOTFLD = TOTFLD + NOFLD
      TOTVRT = TOTVRT + NVRT
      ARRAY(FD1+1)=IPT + FD1
      ARRAY(FD1+2)=LNCAT
      ARRAY(FD1+3)=NOFLD
C
C     WRITE STATS FOR THESE CLUSTERS ON SCRATCH FILE 18
C
      CALL SCRTCH(KVAR,ARRAY(NI),ARRAY(MEANS))
C
C     GO READ IN ANOTHER CLASS
C
      CORBAS=CORBAS+IPT
      IF(LAST.NE.1)GO TO 5
      IF(NOCLS.LT.NCLASS)GO TO 1
C
C     NOW READ SCRATCH FILE AND STORE ON SAVTAP FILE AND PUNCH ON
C     CARDS IF REQUESTED.
C
      MEANS1=N1+MAXPUP
      SM1=MEANS1 + MAXPUP*NOFEAT
      L1=SM1
      CALL PCHSTI(ARRAY(1),ARRAY(N1),ARRAY(MEANS1),KVAR,ARRAY(L1),
     *            ARRAY(SM1))
      RETURN
   30 KV=KVARDM
      WRITE(6,200)KV
      CALL CMERR
  200 FORMAT(' DIMENSION LIMIT OF ',I6,' FOR COVARIANCES EXCEEDED')
C
C     INTERNAL ROUTINE TO SAVE STATS ON SCRATCH FILE
C
      SUBROUTINE SCRTCH(COVAR,N,NMEANS)
      DIMENSION COVAR(1),NMEANS(1),N(1)
      IF(NOCLS.EQ.1)ADRES=BEGIN3
      CALL RWRITE(ADRES,N,LNCAT,ISTAT)
      ADRES=ADRES+LNCAT
      IN=NOFEAT*LNCAT
      CALL RWRITE(ADRES,NMEANS,IN,JSTAT)
      ADRES=ADRES+IN
      IN=VARSIZ*LNCAT
      CALL RWRITE(ADRES,COVAR,IN,LSTAT)
      ADRES=ADRES+IN
C     WAIT FOR I/O COMPLETION
   60 IF(LSTAT.EQ.1 )GO TO 60
      RETURN
      END                                                    ISOCC830
```

AFS 36

```
      CALL COVARI(KVAR,ARRAY(IDAT),ARRAY(KPLCE),ARRAY(MEANS1),
     *            ARRAY(NI))
C
C     CREATE ENIPS INTERFACE TAPE
C
      IF(ERPKEY.GT.0)CALL TWRITE(KVAR,ARRAY(MEANS1),ARRAY(STDEV1),
     *                           ARRAY(NI))
      TOTSUB = TOTSUB + LNCAT
      NOCLS = NOCLS + 1
      TOTFLD = TOTFLD + NOFLD
      TOTVRT = TOTVRT + NVRT
      ARRAY(FD1+1)=IPT + FD1
      ARRAY(FD1+2)=LNCAT
      ARRAY(FD1+3)=NOFLD
C
C     WRITE STATS FOR THESE CLUSTERS ON SCRATCH FILE 18
C
      CALL SCRTCH(KVAR,ARRAY(NI),ARRAY(MEANS1))
C
C     GO READ IN ANOTHER CLASS
C
      CORBAS=CORBAS+IPT
      IF(LAST.NE.1)GO TO 5
      IF(NOCLS.LT.NCLASS)GO TO 1
C
C     NOW READ SCRATCH FILE AND STORE ON SAVTAP FILE AND PUNCH ON
C     CARDS IF REQUESTED.
C
      MEANS1=N1+MAXPOP
      SM1=MEANS1 + MAXPOP*NOFEAT
      L1=SM1
      CALL PCHSTI(ARRAY(1),ARRAY(N1),ARRAY(MEANS1),KVAR,ARRAY(L1),
     *            ARRAY(SM1))
      RETURN
   30 KV=KVARDM
      WRITE(6,200)KV
      CALL CMERR
  200 FORMAT(' DIMENSION LIMIT OF ',I6,' FOR COVARIANCES EXCEEDED')
C
C     INTERNAL ROUTINE TO SAVE STATS ON SCRATCH FILE
C
      SUBROUTINE SCRTCH(COVAR,N,NMEANS)
      DIMENSION COVAR(1),NMEANS(1),N(1)
      IF(NUCLS.EQ.1)ADRES=BEGIN3
      CALL RWRITE(ADRES,N,LNCAT,ISTAT)
      ADRES=ADRES+LNCAT
      IN=NOFEAT*LNCAT
      CALL RWRITE(ADRES,NMEANS,IN,JSTAT)
      ADRES=ADRES+IN
      IN=VARSIZ*LNCAT
      CALL RWRITE(ADRES,COVAR,IN,LSTAT)
      ADRES=ADRES+IN
C     WAIT FOR I/O COMPLETION
   60 IF(LSTAT.EQ.1 )GO TO 60
      RETURN
      END                                                    ISOCC830
```

A-S 36

APPENDIX B

APPENDIX B

```
000540 279L      0001    001650 280L      0001    001694 282L      0001    001630 283L      0001    001691
000665 285L      0001    000210 30L       0001    000232 35L       0001    000461 355G      0001    000294
000479 364G      0001    000250 37L       0001    000567 373G      0001    000522 402G      0001    000305
000309 490F      0000    000306 490F      0001    000055 5L        0001    000330 50L       0000    000316
000320 510F      0000    000322 550F      0000    000325 630F      0000    000331 640F      0000    000264
000346 650F      0000    000364 660F      0000    000401 670F      0000    000475 680F      0000    000503
000510 695F      0001    000353 70L       0000    000520 700F      0000    000526 710F      0000    000533
000544 720F      0000    000566 725F      0000    000570 730F      0000    000576 740F      0001    001661
000271 750F      0000    000277 755F      0001    001710 765G      0001    000376 80L       0001    000421
000073 ASAV      0004    000074 ASAVFL    0003    000115 BEGIN1    0003    000116 BEGIN2    0003    000113
1 000247 BLANK   0004    000055 BMFILE    0004    000056 BNKEY     0000 I  000041 CARD      0000 I  000294
1 000000 CHAR    0003 R  000016 CHNTHS    0003 I  000426 CHNVC     0003    000120 CLSNAM    0000 I  000255
1 000254 COL     0004 I  000037 COMENT    0000 I  000004 COMVEC    0000 I  000257 DASKEY    0004 I  000055
1 000016 DATE    0004 I  000071 DATFIL    0000 I  000244 DBCD      0003 R  000005 DLMIN     0004    000063
000066 DROMAD    0000 I  000243 EBCD      0000 I  000241 EQUAL     0000 I  000004 EQUVEC    0003 I  000465
000062 ERIPTP    0004 I  000063 EMPKEY    0000 I  000245 FBCD      0003 I  000137 FETVEC    0006 I  000001
1 000000 FLTNUM  0004 I  000000 HEAD      0004 I  000002 HED1      0004 I  000023 HED2      0004    000055
000060 HISKEY    0000 I  000251 I         0000 I  000235 IBCD      0003    000119 IBEGIN    0003    000013
1 000015 ICHN    0000 I  000250 ICNT      0003 I  000326 INFILE    0003 I  000327 INITM     0000    000643
1 000325 INUNIT  0000 I  000006 INVEC     0003 I  000324 IORDER    0000 I  000260 IPCT      0003    000123
000011 IRD       0003 I  000273 ISOKEY    0000 I  000252 ISTART    0003 I  000000 ISTOP     0000 I  000251
000012 KPTS      0003 I  000004 KRN       0000 I  000237 LBCD      0003    000001 LBCAT     0000 I  000254
1 000007 MAP     0004    000100 MAPFIL    0003 I  000274 MAPFMT     0003 I  000275 MAPKEY    0004    000055
000064 MAPUNT    0003 I  000136 MAXCLS    0000 I  000240 MBCD      0003 I  000125 NCLASS    0004    000074
000075 NMSTUN    0003 I  000002 NMIN      0003 I  000464 NOCHAN    0003    000132 NOCL      0003    000124
1 000135 NOFEAT  0004    000065 NOFILE    0003    000121 NOFLD     0003    000013 NOPTS     0003 I  000466
1 000425 NOSUB2  0007 I  000000 NUMBER    0003    000133 NVRT      0003    000113 NWDS      0005 I  000001
000134 NXTCLS    0004    000070 PAGSIZ    0000 I  000242 PBCD      0003 R  000322 PERCEN    0003 I  000334
1 000014 PUNCH   0000 I  000261 SAVE1     0000 I  000262 SAVE2     0004 I  000054 SAVTAP    0004    000077
1 000006 SEP     0003 I  000276 SEQUEN    0003 I  000323 SIMERP    0000 I  000137 SLASH     0000 I  000194
1 000010 SPTRIG  0004 I  000074 STAFIL    0003 I  000272 STATKY    0003 R  000004 STDMAX    0003    000001
1 000331 SUBVEC  0003 I  000175 SYMMTX    0003    000130 TOTFLD    0003    000124 TOTPTS    0003    000125
000131 TOTVRT    0003    000123 TOTWRD    0004    000061 TRFORM    0000 I  000236 UBCD      0003    000270
1 000263 ZERO
```

B-138

```
1•          SUBROUTINE SETUP7(ARRAY,TOP,ITIME)
2•     C•••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••SETU0010
3•     C                                                                       SETU0020
4•     C•    SETUP7 READS AND ANALYZES ALL CONTROL CARD INPUT FOR THE
5•     C•    ISOCLS PROCESSOR
6•     C                                                                       SETU0060
7•     C•••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••SETU0070
8•     C                                                                       SETU0080
9•          INCLUDE COMBK5,LIST
9•          PARAMETER MAXPOP=60,MXFET1=30
9•          COMMON/PASS/STOP,LNCAT,BMIN,KRN,STDMAX,DLMIN,SEP,
9•         •                    MAP,SPTRIG,   IRD,  KPTS, NOPTS, PUNCH,
9•         •               ICHN,CHNTHS,ICHAIN(MAXPOP),NWDS,IBEGIN,BEGIN1,
9•         •              BEGIN2,BEGIN3,CLSNAM,NOFLD,IPT,IOTWRD,TOTPTS,
9•         •              NCLASS,NUCLS,TOTSUB,TOTFLD,TOTVRT,NOCL,NVRT
9•         •             ,NXTCLS,NOFEAT,MAXCLS,FETVEC(30),SYMMTX(60)
9•         •,VARSIZ,STATKY,ISOKEY,MAPFMT,MAPKEY,SEQUEN(20),PERCEN,SIMERP
9•         •  ,IORDER,INUNIT,INFILE,INITM,BMIN,SUBVEC(60),NOSUB2,CHNVC(30)
9•         •   ,NOCHAN,ERCOMP,NOSEQ
9•          IMPLICIT INTEGER (A-X)
```

```
000540 279L      0001   000650 280L      0001   001604 282L      0001   001630 283L      0001   001691
001665 285L      0001   000210 30L       0001   000232 35L       0001   000461 355G      0001   000241
000474 364G      0001   000250 37L       0001   000507 373G      0001   000522 402G      0001   000301
000304 440F      0000   000306 490F      0001   000055 5L        0001   000330 50L       0000   000311
000320 510F      0000   000322 550F      0000   000325 630F      0000   000331 640F      0000   000261
000346 650F      0000   000364 660F      0000   000401 670F      0000   000475 680F      0000   000501
000510 695F      0001   000353 70L       0000   000520 700F      0000   000526 710F      0000   000531
000544 720F      0000   000556 725F      0000   000570 730F      0000   000576 740F      0001   001661
000271 750F      0000   000277 755F      0001   001710 765G      0001   000376 80L       0001   000421
000073 ASAV      0004   000074 ASAVFL    0003   000115 BEGIN1    0003   000116 BEGIN2    0003   000111
I 000247 BLANK   0004   000055 BMFILE    0004   000056 BMKEY     0000 I 000041 CARD      0000 I 000241
I 000000 CHAR    0003 R 000016 CHNIMS     0003 I 000426 CHNVC     0003   000120 CLSNAM    0000 I 000251
I 000254 COL     0004 I 000037 COMENT     0000 I 000004 COMVEC    0000 I 000257 DASKEY    0004 I 000051
I 000016 DATE    0004 I 000071 DATFIL     0000 I 000244 DBCD      0003 R 000005 DLMIN     0004   000061
000066 DRUMAD    0000 I 000243 EBCD       0000 I 000241 EQUAL     0000 I 000004 EQUVEC    0003 I 000461
000062 ERIPTP    0004 I 000063 ERPKEY     0000 I 000245 FBCD      0003 I 000137 FETVEC    0006 I 000001
I 000000 FLTNUM  0004 I 000000 HEAD       0004 I 000002 HED1      0004 I 000023 HED2      0004   000051
000060 HISKEY    0000 I 000251 I          0000 I 000235 IBCD      0003   000114 IBEGIN    0003   000011
I 000015 ICHN    0000 I 000250 ICNT       0003 I 000326 INFILE    0003 I 000327 INITM     0000   000641
I 000325 INUNIT  0000 I 000006 INVEC      0003 I 000324 IORDER    0000 I 000260 IPCT      0003   000121
000011 IRD       0003 I 000274 ISORKY     0000 I 000262 ISTART    0003 I 000300 ISTOP     0000 I 000251
000012 KPTS      0003 I 000003 KRN        0000 I 000237 LBCD      0003   000001 LNCAT     0000 I 000251
I 000007 MAP     0004   000100 MAPFIL     0003 I 000274 MAPFMT    0003 I 000275 MAPKEY    0004   000051
000064 MAPUNT    0003 I 000136 MAXCLS     0000 I 000240 MBCD      0003 I 000125 NCLASS    0004   000071
000075 NMSTUN    0003 I 000002 NMIN       0003 I 000464 NOCHAN    0003   000132 NOCL      0003   000121
I 000135 NOFEAT  0004   000065 NOFILE     0003   000121 NOFLD     0003   000013 NOPTS     0003 I 000461
I 000425 NOSUB2  0007 I 000000 NUMBER     0003   000133 NVRT      0003   000113 NWDS      0005 I 000001
000134 NXTCLS    0004   000070 PAGSIZ     0000 I 000242 PBCD      0003 R 000322 PERCEN    0003 I 000331
I 000014 PUNCH   0000 I 000261 SAVE1      0000 I 000262 SAVE2     0004 I 000054 SAVTAP    0004   000071
I 000006 SEP     0003 I 000276 SEQUEN     0003 I 000323 SIMERP    0000 I 000137 SLASH     0000 I 000141
I 000010 SPTRIG  0004 I 000072 STAFIL     0003 I 000272 STATKY    0003 R 000004 STDMAX    0003   000001
I 000331 SUBVEC  0003 I 000175 SYMMTX     0003   000130 TOTFLD    0003   000124 TOTPTS    0003   000121
000131 TOTVRT    0003   000123 TOTWRD     0003   000061 TRFORM    0000 I 000236 UBCD      0003   000271
I 000263 ZERO
```

B-138

```
1•          SUBROUTINE SETUP7(ARRAY,TOP,ITIME)
2•     C••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••SETU0010
3•     C                                                                       SETU0020
4•     C•    SETUP7 READS AND ANALYZES ALL CONTROL CARD INPUT FOR THE
5•     C•    ISOCLS PROCESSOR
6•     C                                                                       SETU0060
7•     C••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••SETU0070
8•     C                                                                       SETU0080
9•          INCLUDE COMBK5,LIST
9•          PARAMETER MAXPOP=60,MXFET1=30
9•          COMMON/PASS/STOP,LNCAT,PMIN,KRN,STDMAX,DLMIN,SEP,
9•         •                    MAP,SPTRIG,   IRD,  KPTS, NOPTS, PUNCH,
9•         •              ICHN,CHNTMS,ICHAIN(MAXPOP),NWDS,IBEGIN,BEGIN1,
9•         •             BEGIN2,BEGIN3,CLSNAM,NOFLD,IPT,TOTWRD,TOTPTS,
9•         •             NCLASS,NOCLS,TOTSUB,TOTFLD,TOTVRT,NOCL,NVRT
9•         •            ,NXTCLS,NOFEAT,MAXCLS,FETVEC(30),SYMMTX(60)
9•         *,VARSIZ,STATKY,ISOKEY,MAPFMT,MAPKEY,SEQUEN(20),PERCEN,SIMERP
9•         •  ,IORDER,INUNIT,INFILE,INITM,PMIN,SUBVEC(60),NOSUB2,CHNVC(30)      •NEW
9•         •  ,NOCHAN,ERCOMP,NOSEQ                                              •NEW
9•          IMPLICIT INTEGER (A-X)                                              ••-I
```

```
      END
      INCLUDE COMBK4,LIST
      COMMON/GLOBAL/HEAD(42),MAPTAP,DATAPE,SAVTAP,BNFILE,BMKEY,
     *          HISFIL,HISKEY,TRFORM,ERIPTP,ERPKEY,MAPUNT,NOFILE,
     *   DRUMAD,DRMWDS,PAGSIZ,DATFIL,STAFIL,ASAV,ASAVFL
     *  ,NHSTUN,NHSTFI,SCTRUN,MAPFIL                               •NE•
      END                                                          •o-1
      REAL DLMIN,CHNTMS,SEP,STDMAX,PERCEN
      DIMENSION ARRAY(1),CHAR(2)
      DIMENSION COMVEC(2),EQUVEC(2)
      DATA COMVEC/1,',',/,EQUVEC/1,'='/
      EQUIVALENCE(STOP,ISTOP)
      INCLUDE COMBK4,LIST
      DIMENSION HED1(10),HED2(10),DATE(2),COMENT(10)
      EQUIVALENCE (HED1(1),HEAD(3)),(DATE(1),HEAD(15)),
     *           (HED2(1),HEAD(20)),(COMENT(1),HEAD(32))
      END
       DIMENSION INVEC(27),CARD(62),SLASH(2)                       •NE•
       DATA SLASH/1,'/'/                                           •NE•
      DATA INVEC /'CHANNE','ISTOP','NMIN','KMN','STDMAX','DLMIN',  •o-1
     *'SEP','HED1','HED2','DATE','-END-','COMMEN',
     *'SYMBOL','MEANS','MAP','OPTION','CLASSE','CHAIN',
     *'CLUSTE','FORMAT','DATAFI','STATFI','SEQUEN','PERCEN','MODULE' •NE•
     *,'SUBCLA','PMIN'/                                            •NE•
      DIMENSION SMBLS(60)                                          •o-1
      DATA IBCD/'I'/,UBCD/'U'/,LBCD/'L'/,MBCD/'M'/
      DATA CHAR/1,'='/,  EQUAL/'='/
      DATA PBCD/'P'/,EBCD/'E'/,DBCD/'D'/
      DATA FBCD/'F'/,CBCD/'C'/
      DATA SMBLS /'1','2','3','4','5','6','7','8','9','A','B','C','D',
     1           'E','F','G','H','I','J','K','L','M','N','O','P','Q',
     2           'R','S','T','U','V','W','X','Y','Z','$','*','A','/',
     3           '-','.','+','S','B','=','0','?','>','<',')','(',':',
     4           '!','\',';','=',1H','',',','.',' ' /
       DATA SEQUEN/'S','C','S','C','S','C','S','C','S','C','S','C','S',
     *'C','S','C','S','C','S',0/
      DATA BLANK/' '/                                           SETU0320
      IF(ITIME.NE.1)GO TO 5
C
C     SET UP DEFAULT VALUES FOR INPUT PARAMETERS
C
       ERCOMP=0                                                    •NE•
       IORDER=0                                                    •NE•
      NOSEQ = 2                                                    •NE•
      NOCHAN = 0                                                   •NE•
       SEP=1                                                       •o-1
       NOSUB2=0                                                    •NE•
       PMIN=1                                                      •NE•
        INITM=0                                                    •NE•
      MAPKEY=1
      MAPFMT = 0
      ICNT=0
      PUNCH=0
      DLMIN=3.2
      STDMAX=4.5
      MAP=20
      STATKY=0
      ISTOP=10
```

```
  9.              END
 10.              INCLUDE COMBK6,LIST
 10.              COMMON/GLOBAL/HEAD(42),MAPTAP,DATAPE,SAVTAP,BMFILE,BMKEY,
 10.             .          HISFIL,HISKEY,TRFORM,ERIPTP,ERPKEY,MAPUNT,NOFILE,
 10.             .  DRUMAD,DRMWDS,PAGSIZ,DATFIL,STAFIL,ASAV,ASAVFL
 10.             .  ,NHSTUN,NHSTFI,SCTRUN,MAPFIL                                  .NEW
 10.              END                                                            ..-1
 11.               REAL OLMIN,CHNTHS,SEP,STDMAX,PERCEN
 12.              DIMENSION ARRAY(1),CHAR(2)
 13.              DIMENSION COMVEC(2),EQUVEC(2)
 14.              DATA COMVEC/1,',',/,EQUVEC/1,'='/
 15.              EQUIVALENCE(STOP,ISTOP)
 16.              INCLUDE COMBK4,LIST
 16.              DIMENSION HED1(10),HED2(10),DATE(2),COMENT(10)
 16.              EQUIVALENCE (HED1(1),HEAD(3)),(DATE(1),HEAD(15)),
 16.             .          (HED2(1),HEAD(20)),(COMENT(1),HEAD(32))
 16.              END
 17.               DIMENSION INVEC(27),CARD(62),SLASH(2)                         .NEW
 18.               DATA SLASH/1,'/'/                                             .NEW
 19.              DATA INVEC /'CHANNE','ISTOP','NMIN','KMN','STDMAX','OLMIN',    ..-1
 20.             .'SEP','HED1','HED2','DATE','*END*','COMMEN',
 21.             .'SYMBOL','MEANS','MAP','OPTION','CLASSE','CHAIN',
 22.             .'CLUSTE','FORMAT','DATAFI','STATFI','SEQUEN','PERCEN','MODULE'  .NEW
 23.             .,'SUBCLA','PMIN'/                                               .NEW
 24.              DIMENSION SMBLS(60)                                            ..-1
 25.              DATA IBCD/'I'/,UBCD/'U'/,LBCD/'L'/,MBCD/'M'/
 26.              DATA CHAR/1,'='/,  EQUAL/'='/
 27.              DATA PBCD/'P'/,EBCD/'E'/,DBCD/'D'/
 28.              DATA FBCD/'F'/,CBCD/'C'/
 29.              DATA SMBLS /'1','2','3','4','5','6','7','8','9','A','B','C','D',
 30.                          'E','F','G','H','I','J','K','L','M','N','O','P','Q',
 31.             1            'R','S','T','U','V','W','X','Y','Z','$','*','&','/',
 32.             2            '-','=','+','S','@','=','O','?','>','<',')','(',';',
 33.             3            '!','\',':','%',1H',',','.',' '/
 34.             4 DATA SEQUEN/'S','C','S','C','S','C','S','C','S','C','S','C','S',
 35.             .'C','S','C','S','C','S',0/
 36.              DATA BLANK/' '/                                        SETU0320
 37.              IF(ITIME.NE.1)GO TO 5
 38.         C.
 39.         C.   SET UP DEFAULT VALUES FOR INPUT PARAMETERS
 40.         C.
 41.               ERCOMP=0                                                      .NEW
 42.               IORDER=0                                                      .NEW
 43.              NOSEQ = 2                                                      .NEW
 44.              NOCHAN = 0                                                     .NEW
 45.               SEP=1                                                         ..-1
 46.               NOSUB2=0                                                      .NEW
 47.               PMIN=1                                                        .NEW
 48.               INITM=0                                                       .NEW
 49.              MAPKEY=1
 50.              MAPFMT = 0
 51.              ICNT=0
 52.              PUNCH=0
 53.              OLMIN=3.2
 54.              STDMAX=4.5
 55.              MAP=20
 56.              STATKY=0
 57.              ISTOP=10
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
      NMIN=30                                                                   
      KRN=20                                                                    
      NCLASS=1                                                                  
      MAACLS=MAXPOP                                                             
      ICHN=0                                                                    
      DO 1 I=1,MAXPOP                                                           
    1 SYMMTX(I)=SMBLS(I)                                                        
      NOFEAT    = 0                                                           
      ISTART    = 0                                                           
    5 CONTINUE                                                                  
      IF(ITIME.NE.1)WRITE(6,HEAD)                                               
      WRITE (6,630)                                                 SETU0350    
   10 READ(5,480) CODE,CARD                                                     
      WRITE (6,550)CODE,CARD                                        SETU0370    
      COL=0                                                         SETU0380    
       DO 20 I=1,27                                                      *NEW     
   20 IF (CODE .EQ. INVEC(I)) GO TO (30,50,70,80,90,100,110,130,        **-1     
     *140,150,280,160,170,250,210,240,200,235,230,220,260,270,246,249    *NEW     
     *,256,263,284),I                                                   *NEW     
      WRITE (6,490)CODE,CARD                                        SETU0420**-1
      GO TO 10                                                      SETU0430    
C                                                                   SETU0440    
C     CHANNEL CARD                                                  SETU0460    
C                                                                   SETU0470    
   30 J = NXTCHR(CARD,COL)                                          SETU0480    
      IF (J.EQ.BLANK) GO TO 10                                                  
      IF(ITIME.EQ.1)GO TO 35                                                    
      WRITE(6,640)                                                              
      GO TO 10                                                                  
   35 CONTINUE                                                                  
       IF(J.EQ.'S')GO TO 37                                             *NEW     
       IF(J.EQ.'D')GO TO 43                                             *NEW     
   36 WRITE(6,645)                                                      *NEW     
  645 FORMAT(' ERROR ON CHANNEL CARD')                                  *NEW     
      GO TO 10                                                          *NEW     
   37 M = FIND(CARD,COL,EQUVEC)                                         *NEW     
      IF (M .EQ. -1) GO TO 36                                           *NEW     
      NOCHAN = NUMBER(CARD,COL,CHNVC,NOCHAN)                            *NEW     
      COL = COL - 1                                                     *NEW     
      CALL ORDER(CHNVC,NOCHAN)                                          *NEW     
      GO TO 30                                                          *NEW     
   43 M = FIND(CARD,COL,EQUVEC)                                         *NEW     
      IF (M .EQ. -1) GO TO 36                                           *NEW     
      NOFEAT = NUMBER(CARD,COL,FETVEC,NOFEAT)                           **-1     
      COL = COL - 1                                                     *NEW     
      CALL ORDER(FETVEC,NOFEAT)                                         *NEW     
      GO TO 30                                                          *NEW     
C                                                                   SETU0540**-1
C     ISTOP CARD (MAXIMUM NUMBER OF ITERATIONS)                     SETU0550    
C                                                                   SETU0560    
   50 J = NXTCHR(CARD,COL)                                          SETU0570    
      IF (J.EQ.BLANK) GO TO 10                                      SETU0580    
      COL=COL-1                                                     SETU0590    
      J      = NUMBER(CARD,COL,ISTOP,ISTART)                                    
      GO TO 10                                                      SETU0620    
C                                                                   SETU0720    
C     NMIN CARD (MINIMUM NUMBER OF POINTS PER CLUSTER)              SETU0730    
C                                                                   SETU0740    
```

```
      NMIN=30                                                                    
      KRN=20                                                                     
      NCLASS=1                                                                   
      MAXCLS=MAXPOP                                                              
      ICHN=0                                                                     
      DO 1 I=1,MAXPOP                                                            
    1 SYMMTX(I)=SMBLS(I)                                                         
      NOFEAT    = 0                                                            
      ISTART    = 0                                                            
    5 CONTINUE                                                                   
      IF(ITIME.NE.1)WRITE(6,HEAD)                                                
      WRITE (6,630)                                                  SETU0350    
   10 READ(5,480) CODE,CARD                                                      
      WRITE (6,550)CODE,CARD                                         SETU0370    
      COL=0                                                          SETU0380    
       DO 20 I=1,27                                                      •NEW     
   20 IF (CODE .EQ. INVEC(I)) GO TO (30,60,70,80,90,100,110,130,        ••-1     
     •140,150,280,160,170,250,210,240,200,235,230,220,260,270,246,249    •NEW     
     •,256,263,284),I                                                    •NEW     
      WRITE (6,490)CODE,CARD                                         SETU0420••-1
      GO TO 10                                                       SETU0430    
C                                                                    SETU0440    
C     CHANNEL CARD                                                               
C                                                                    SETU0460    
   30 J = NXTCHR(CARD,COL)                                           SETU0470    
      IF (J.EQ.BLANK) GO TO 10                                       SETU0480    
      IF(ITIME.EQ.1)GO TO 35                                                     
      WRITE(6,640)                                                               
      GO TO 10                                                                   
   35 CONTINUE                                                                   
       IF(J.EQ.'S')GO TO 37                                              •NEW     
       IF(J.EQ.'D')GO TO 43                                              •NEW     
   36 WRITE(6,645)                                                       •NEW     
  645 FORMAT(' ERROR ON CHANNEL CARD')                                   •NEW     
      GO TO 10                                                           •NEW     
   37 M = FIND(CARD,COL,EQUVEC)                                          •NEW     
      IF (M .EQ. -1) GO TO 36                                            •NEW     
      NOCHAN = NUMBER(CARD,COL,CHNVC,NOCHAN)                             •NEW     
      COL = COL - 1                                                      •NEW     
      CALL ORDER(CHNVC,NOCHAN)                                           •NEW     
      GO TO 30                                                           •NEW     
   43 M = FIND(CARD,COL,EQUVEC)                                          •NEW     
      IF (M .EQ. -1) GO TO 36                                            •NEW     
      NOFEAT = NUMBER(CARD,COL,FETVEC,NOFEAT)                            ••-1     
      COL = COL - 1                                                      •NEW     
      CALL ORDER(FETVEC,NOFEAT)                                          •NEW     
      GO TO 30                                                           •NEW     
C                                                                    SETU0540••-1
C     ISTOP CARD (MAXIMUM NUMBER OF ITERATIONS)                      SETU0550    
C                                                                    SETU0560    
   50 J = NXTCHR(CARD,COL)                                           SETU0570    
      IF (J.EQ.BLANK) GO TO 10                                       SETU0580    
      COL=COL-1                                                      SETU0590    
      J      = NUMBER(CARD,COL,ISTOP,ISTART)                                     
      GO TO 10                                                       SETU0620    
C                                                                    SETU0720    
C     NMIN CARD (MINIMUM NUMBER OF POINTS PER CLUSTER)               SETU0730    
C                                                                    SETU0740    
```

B-3 do

```
      70 J = NXTCHR(CARD,COL)                                          SETU0750
         IF (J.EQ.BLANK) GO TO 10                                      SETU0760
         COL = COL-1                                                   SETU0770
         J      = NUMBER(CARD,COL,NMIN ,ISTART)
         GO TO 10                                                      SETU0800
C
C     KRN CARD (NUMBER OF ITERATIONS PER FULL OUTPUT)                  SETU0820
C                                                                      SETU0830
      80 J = NXTCHR(CARD,COL)                                          SETU0840
         IF (J.EQ.BLANK) GO TO 10                                      SETU0850
         COL = COL-1                                                   SETU0860
         J      = NUMBER(CARD,COL,KRN  ,ISTART)
         GO TO 10                                                      SETU0890
C                                                                      SETU0900
C     STDMAX CARD (MAXIMUM STANDARD DEVIATION PER CLUSTER)             SETU0910
C                                                                      SETU0920
      90 J = FLTNUM(CARD,COL,STDMAX,1)                                 SETU0930
         GO TO 10                                                      SETU0940
C                                                                      SETU0950
C     DLMIN CARD (MINIMUM DISTANCE BETWEEN CLUSTER MEANS)              SETU0960
C                                                                      SETU0970
     100 J = FLTNUM(CARD,COL,DLMIN,1)                                  SETU0980
         GO TO 10                                                      SETU0990
C                                                                      SETU1000
C     SEP CARD  (DISTANCE FOR SPLITTING)                               SETU1010
C                                                                      SETU1020
     110 J = FLTNUM(CARD,COL,SEP,1)                                    SETU1030
         SPLRIG=1                                                      SETU1040
         GO TO 10                                                      SETU1050
C                                                                      SETU1150
C     MED1 CARD                                                        SETU1160
C                                                                      SETU1170
     130 READ (30,500)MED1                                             SETU1180
         GO TO 10                                                      SETU1190
C                                                                      SETU1200
C     MED2,CARD                                                        SETU1210
C                                                                      SETU1220
     140 READ (30,500)MED2                                             SETU1230
         GO TO 10                                                      SETU1240
C                                                                      SETU1250
C     DATE CARD                                                        SETU1260
C                                                                      SETU1270
     150 READ(30,510) DATE
         GO TO 10                                                      SETU1290
C                                                                      SETU1300
C     COMMENT CARD                                                     SETU1310
C                                                                      SETU1320
     160 READ(30,500)COMENT
         GO TO 10                                                      SETU1340
C                                                                      SETU1350
C     SYMBOLS CARD                                                     SETU1360
C                                                                      SETU1370
     170 CONTINUE
     180 ICNT=ICNT + 1                                                 SETU1390
         IF (ICNT.GT.MAXPOP) GO TO 10                                  SETU1400
         SYMMTX(ICNT)=BLANK
     190 M=NXTCHR(CARD,COL)                                            SETU1410
         IF (M.EQ.BLANK) GO TO 10                                      SETU1420
```

B4 41

```
116*      70 J = NXTCHR(CARD,COL)                                          SETU0750
117*         IF (J.EQ.BLANK) GO TO 10                                      SETU0760
118*         COL = COL-1                                                   SETU0770
119*         J      = NUMBER(CARD,COL,NMIN ,ISTART)
120*         GO TO 10                                                      SETU0800
121*    C*                                                                 SETU0810
122*    C*    KRN CARD (NUMBER OF ITERATIONS PER FULL OUTPUT)              SETU0820
123*    C*                                                                 SETU0830
124*      80 J = NXTCHR(CARD,COL)                                          SETU0840
125*         IF (J.EQ.BLANK) GO TO 10                                      SETU0850
126*         COL = COL-1                                                   SETU0860
127*         J      = NUMBER(CARD,COL,KRN  ,ISTART)
128*         GO TO 10                                                      SETU0890
129*    C*                                                                 SETU0900
130*    C*    STDMAX CARD (MAXIMUM STANDARD DEVIATION PER CLUSTER)         SETU0910
131*    C*                                                                 SETU0920
132*      90 J = FLTNUM(CARD,COL,STDMAX,1)                                 SETU0930
133*         GO TO 10                                                      SETU0940
134*    C*                                                                 SETU0950
135*    C*    DLMIN CARD (MINIMUM DISTANCE BETWEEN CLUSTER MEANS)          SETU0960
136*    C*                                                                 SETU0970
137*     100 J = FLTNUM(CARD,COL,DLMIN,1)                                  SETU0980
138*         GO TO 10                                                      SETU0990
139*    C*                                                                 SETU1000
140*    C*    SEP CARD  (DISTANCE FOR SPLITTING)                           SETU1010
141*    C*                                                                 SETU1020
142*     110 J = FLTNUM(CARD,COL,SEP,1)                                    SETU1030
143*         SPLRIG=1                                                      SETU1040
144*         GO TO 10                                                      SETU1050
145*    C*                                                                 SETU1150
146*    C*    MED1 CARD                                                    SETU1160
147*    C*                                                                 SETU1170
148*     130 READ (30,500)MED1                                             SETU1180
149*         GO TO 10                                                      SETU1190
150*    C*                                                                 SETU1200
151*    C*    MED2,CARD                                                    SETU1210
152*    C*                                                                 SETU1220
153*     140 READ (30,500)MED2                                             SETU1230
154*         GO TO 10                                                      SETU1240
155*    C*                                                                 SETU1250
156*    C*    DATE CARD                                                    SETU1260
157*    C*                                                                 SETU1270
158*     150 READ(30,510) DATE
159*         GO TO 10                                                      SETU1290
160*    C*                                                                 SETU1300
161*    C*    COMMENT CARD                                                 SETU1310
162*    C*                                                                 SETU1320
163*     160 READ(30,500)COMENT
164*         GO TO 10                                                      SETU1340
165*    C*                                                                 SETU1350
166*    C*    SYMBOLS CARD                                                 SETU1360
167*    C*                                                                 SETU1370
168*     170 CONTINUE
169*     180 ICNT=ICNT + 1                                                 SETU1390
170*         IF (ICNT.GT.MAXPOP) GO TO 10                                  SETU1400
171*         SYMMTX(ICNT)=BLANK
172*     190 M=NXTCHR(CARD,COL)                                            SETU1410
173*         IF (M.EQ.BLANK) GO TO 10                                      SETU1420
```

```
174•            IF(M.EQ.',')GO TO 180
175•            SYMMTX(ICNT) = M
176•      195   M=NXTCHR(CARD,COL)
177•            IF(M.EQ.BLANK)GO TO 10
178•            IF(M.NE.',')GO TO 195
179•            GO TO 180                                                    SETU1450
180•      C•
181•      C•    MAXCLASS CARD (NO. CLASSES FOR THIS EXECUTION OF ISOCLS- STATISTIC
182•      C•                   FILE WILL BE WRITTEN AFTER 'NCLASS' CLASSES HAVE
183•      C•                   BEEN CLUSTERED)
184•      C•
185•        200 IF(ITIME.EQ.1) GO TO 205
186•            WRITE(6,650)
187•            GO TO 10
188•        205 J=NUMBER(CARD,COL,NCLASS,ISTART)
189•            GO TO 10
190•      C•                                                                 SETU1550
191•      C•    MAP CARD  (NUMBER OF ITERATIONS TO OUTPUT MAP)               SETU1560
192•      C•                                                                 SETU1570
193•        210 J=NXTCHR(CARD,COL)                                           SETU1580
194•            IF (J.EQ.BLANK) GO TO 10                                     SETU1590
195•            COL=COL-1                                                    SETU1600
196•            J          = NUMBER(CARD,COL,MAP    ,ISTART)
197•            GO TO 10                                                     SETU1630
198•      C•                                                                 SETU1640
199•      C•    PUNCH CARD (PUNCH STATISTICS ON CARDS)                       SETU1650
200•      C•                                                                 SETU1660
201•        215 M = FIND(CARD,COL,CHAR)
202•            IF (M .NE. 2)  PUNCH = 1
203•            IF(M.NE.2)GO TO 245
204•            J          = NUMBER(CARD,COL,PUNCH ,ISTART)
205•            GO TO 245
206•      C•
207•      C•    MAP FORMAT CARD
208•      C•
209•        220 M = NXTCHR(CARD,COL)
210•            IF (M .EQ. UBCD)  MAPFMT = 1
211•            IF (M .EQ. LBCD)  MAPFMT = 2
212•            IF (M .EQ. 1H  ) MAPFMT = 1
213•            GO TO 10
214•      C•                                                                 SETU1740
215•      C•    CLUSTERS CARD (MAX. NO. OF CLUSTERS PER CLASS)
216•      C•                                                                 SETU1760
217•        230 J=NXTCHR(CARD,COL)                                           SETU1770
218•            IF (J.EQ.BLANK) GO TO 10                                     SETU1780
219•            COL=COL-1                                                    SETU1790
220•            J          = NUMBER(CARD,COL,MAXCLS,ISTART)
221•            GO TO 10                                                     SETU1820
222•      C•
223•      C•    CHAIN CARD (CHAIN CLUSTERS WHICH ARE DLMIN UNITS APART)
224•        235 ICHN=1
225•            J=FLTNUM(CARD,COL,CHNTHS,1)
226•            GO TO 10
227•      C
228•      C     OPTION CARD
229•      C
230•        240 J = NXTCHR(CARD,COL)
231•            IF (J .EQ. BLANK)  GO TO 10
```

```
174*            IF(M.EQ.',')GO TO 180
175*            SYNMTX(ICNT) = M
176*      195   M=NXTCHR(CARD,COL)
177*            IF(M.EQ.BLANK)GO TO 10
178*            IF(M.NE.',')GO TO 195
179*            GO TO 180                                                    SETU1450
180*      C
181*      C     MAXCLASS CARD (NO. CLASSES FOR THIS EXECUTION OF ISOCLS- STATISTIC
182*      C                    FILE WILL BE WRITTEN AFTER 'NCLASS' CLASSES HAVE
183*      C                    BEEN CLUSTERED)
184*      C
185*      200   IF(ITIME.EQ.1) GO TO 205
186*            WRITE(6,650)
187*            GO TO 10
188*      205   J=NUMBER(CARD,COL,NCLASS,ISTART)
189*            GO TO 10
190*      C                                                                  SETU1550
191*      C     MAP CARD  (NUMBER OF ITERATIONS TO OUTPUT MAP)               SETU1560
192*      C                                                                  SETU1570
193*      210   J=NXTCHR(CARD,COL)                                          SETU1580
194*            IF (J.EQ.BLANK) GO TO 10                                     SETU1590
195*            COL=COL-1                                                    SETU1600
196*            J         = NUMBER(CARD,COL,MAP    ,ISTART)
197*            GO TO 10                                                     SETU1630
198*      C                                                                  SETU1640
199*      C     PUNCH CARD (PUNCH STATISTICS ON CARDS)                       SETU1650
200*      C                                                                  SETU1660
201*      215   M = FINU(CARD,COL,CHAR)
202*            IF (M .NE. 2)  PUNCH = 1
203*            IF(M.NE.2)GO TO 245
204*            J         = NUMBER(CARD,COL,PUNCH ,ISTART)
205*            GO TO 245
206*      C
207*      C     MAP FORMAT CARD
208*      C
209*      220   M = NXTCHR(CARD,COL)
210*            IF (M .EQ. UBCD)  MAPFMT = 1
211*            IF (M .EQ. LBCD)  MAPFMT = 2
212*            IF (M .EQ. 1H  ) MAPFMT = 1
213*            GO TO 10
214*      C                                                                  SETU1740
215*      C     CLUSTERS CARD (MAX. NO. OF CLUSTERS PER CLASS)
216*      C                                                                  SETU1760
217*      230   J=NXTCHR(CARD,COL)                                          SETU1770
218*            IF (J.EQ.BLANK) GO TO 10                                     SETU1780
219*            COL=COL-1                                                    SETU1790
220*            J         = NUMBER(CARD,COL,MAXCLS,ISTART)
221*            GO TO 10                                                     SETU1820
222*      C
223*      C     CHAIN CARD (CHAIN CLUSTERS WHICH ARE DLMIN UNITS APART)
224*      235   ICHN=1
225*            J=FLTNUM(CARD,COL,CHNTHS,1)
226*            GO TO 10
227*      C
228*      C     OPTION CARD
229*      C
230*      240   J = NXTCHR(CARD,COL)
231*            IF (J .EQ. BLANK)  GO TO 10
```

```
C
C       ORDER COLOR KEYS
C
        IF(J.EQ.'O')IORDER=1                                            *NE*
C                                                                       *NE*
C       ERROR COMPUTATION                                               *NE*
C                                                                       *NE*
      IF (J .NE. 'E') GO TO 241                                         *NE*
      J = NXTCHR(CARD,COL+1)                                            *NE*
      IF (J .EQ. 'C') ERCOMP = 1                                        *NE*
C                                                                       *NE*
C       ERIPS CARD                                                      *NE*
      IF (J .EQ. '1') ERPKEY = 1                                        *NE*
  241 CONTINUE                                                          *NE*
C                                                                       *0-1
C       PUNCH CARD.
C
      IF (J .EQ. PBCD)  GO TO 215
C.
C.      STATS
C.
        IF(J.NE.'S')GO TO 242
        J=NXTCHR(CARD,COL)
        IF(J.EQ.'T')STATKY=1
        IF(J.EQ.'I')SIMERP=1
C
C       DAS CARD
C
  242   IF(J.EQ.DBCD)DASKEY=1
C
C       ERIPS CARD
C
C       MEAN VECTOR FOR MAPTAP                                          *0-1
        IF(J.EQ.MBCD)MAPKEY=1
C       CLUSTERS FOR MAPTAP
        IF(J.EQ.CBCD)MAPKEY=2
C.
C.      FIND A COMMA
C.
  245 J=FIND(CARD,COL,COMVEC)
      IF(J.LE.0)GO TO 10
      GO TO 240
C
C       SEQUENCE CARD
C
  246   I=1
  247   M=NXTCHR(CARD,COL)
        IF(M.EQ.' ')GO TO 248
        SEQUEN(I)=M
        I=I+1
        GO TO 247
  248 NOSEQ = I - 1                                                     *NE*
      GO TO 10                                                          *0-1
C
C       PERCENT CARD
C
  249   J=NXTCHR(CARD,COL)
        IF(J.EQ.' ')GO TO 10
```

B 6 43

```
C
C     ORDER COLOR KEYS
C
      IF(J.EQ.'O')IORDER=1                                          •NEW
C                                                                   •NEW
C     ERROR COMPUTATION                                             •NEW
C                                                                   •NEW
      IF (J .NE. 'E') GO TO 241                                     •NEW
      J = NXTCHR(CARD,COL+1)                                        •NEW
      IF (J .EQ. 'C') ERCOMP = 1                                    •NEW
C                                                                   •NEW
C     ERIPS CARD                                                    •NEW
      IF (J .EQ. 'I') ERPKEY = 1                                    •NEW
  241 CONTINUE                                                      •NEW
C                                                                   •••1
C     PUNCH CARD.
C
      IF (J .EQ. PBCD)  GO TO 215
C•
C•    STATS
C•
      IF(J.NE.'S')GO TO 242
      J=NXTCHR(CARD,COL)
      IF(J.EQ.'T')STATKY=1
      IF(J.EQ.'I')SIMERP=1
C
C     DAS CARD
C
  242 IF(J.EQ.DBCD)DASKEY=1
C
C     ERIPS CARD
C
C     MEAN VECTOR FOR MAPTAP                                        •••1
      IF(J.EQ.MBCD)MAPKEY=1
C     CLUSTERS FOR MAPTAP
      IF(J.EQ.CBCD)MAPKEY=2
C•
C•    FIND A COMMA
C•
  245 J=FIND(CARD,COL,COMVEC)
      IF(J.LE.0)GO TO 10
      GO TO 240
C
C     SEQUENCE CARD
C
  246 I=1
  247 N=NXTCHR(CARD,COL)
      IF(N.EQ.' ')GO TO 248
      SEQUEN(I)=N
      I=I+1
      GO TO 247
  246 NOSEQ = I - 1                                                 •NEW
      GO TO 10                                                      •••1
C
C     PERCENT CARD
C
  249 J=NXTCHR(CARD,COL)
      IF(J.EQ.' ')GO TO 10
```

B-6 43

```
      COL=COL-1
      J=NUMBER(CARD,COL,IPCT,J)
      PERCEN=1.-FLOAT(IPCT/100.)
      GO TO 10
C
C     MEANS CARD
C
  250 J = NXTCHR(CARD,COL)
      IF(J .EQ. BLANK)     GO TO 10
      IF(J .EQ. CB(D)      GO TO 255
      IF(J .NE. FB(D)      GO TO 10
      ISUKEY=1
      GO TO 10
  255 ISUKEY=1
      CALL RDMEAN(ARRAY)
      GO TO 10
C
C     READ MODULE DECK AND WRITE TO INPUT STAT UNIT AND FILE.
C
  256 SAVE1=SAVTAP
      SAVE2=STAFIL
      SAVTAP=INUNIT
      STAFIL=INFILE
      CALL CRDSTA(ARRAY,TOP)
      SAVTAP=SAVE1
      STAFIL=SAVE2
      IWITM=1
      GO TO 10
C
C     DATA FILE CARD
C
  260 M = NXTCHR(CARD,COL)
      IF (M .EQ. IH ) GO TO 10
      IF (M .EQ. 'U')  GO TO 265
      IF ( M .EQ. 'F')  GO TO 267
  263 WRITE(6,750)
  750 FORMAT(' ERROR ON DATA FILE CARD')
      GO TO 10
  265 J = FIND(CARD,COL,EQUVEC)
      IF (J.EQ. -1)  GO TO 263
      M = NUMBER(CARD,COL,DATAPE,ZERO)
      COL = COL - 1
      GO TO 260
  267 J = FIND(CARD,COL,EQUVEC)
      IF (J .EQ. -1)  GO TO 263
      M = NUMBER(CARD,COL,DATFIL,ZERO)
      DATFIL = DATFIL - 1
      IF (DATFIL .LT. 0)  DATFIL = 0
      COL = COL - 1
      GO TO 260
C
C     STAT FILE CARD
C
  270 M=NXTCHR(CARD,COL)
  271 IF(M.EQ.'I')GO TO 273
      IF(M.EQ.'U')GO TO 275
      IF(M.EQ.'F')GO TO 277
      IF(M.EQ.'O')GO TO 272
```

```
290*                COL=COL-1
291*                J=NUMBER(CARD,COL,IPCT,J)
292*                PERCEN=1.-FLOAT(IPCT/100.)
293*                GO TO 10
294*          C
295*          C     MEANS CARD
296*          C
297*           250  J = NXTCHR(CARD,COL)
298*                IF(J .EQ. BLANK)     GO TO 10
299*                IF(J .LE. CB(D)      GO TO 255
300*                IF(J .NE. FB(D)      GO TO 10
301*                ISOKEY=1
302*                GO TO 10
303*           255  ISOKEY=1
304*                CALL RDMEAN(ARRAY)
305*                GO TO 10
306*          C•
307*          C•    READ MODULE DECK AND WRITE TO INPUT STAT UNIT AND FILE.                    •NEW
308*          C•                                                                               •NEW
309*           256  SAVE1=SAVTAP                                                               •NEW
310*                SAVE2=STAFIL                                                               •NEW
311*                SAVTAP=INUNIT                                                              •NEW
312*                STAFIL=INFILE                                                              •NEW
313*                CALL CRDSTA(ARRAY,TOP)                                                     •NEW
314*                SAVTAP=SAVE1                                                               •NEW
315*                STAFIL=SAVE2                                                               •NEW
316*                INITM=1                                                                    •NEW
317*                GO TO 10                                                                   •NEW
318*          C
319*          C     DATA FILE CARD
320*          C
321*           260  M = NXTCHR(CARD,COL)
322*                IF (M .EQ. 1H ) GO TO 10
323*                IF (M .EQ. 'U')  GO TO 265
324*                IF ( M .EQ. 'F')  GO TO 267
325*           263  WRITE(6,750)
326*           750  FORMAT(' ERROR ON DATA FILE CARD')
327*                GO TO 10
328*           265  J = FIND(CARD,COL,EQUVEC)
329*                IF (J.EQ. -1)  GO TO 263
330*                M = NUMBER(CARD,COL,DATAPE,ZERO)
331*                COL = COL - 1
332*                GO TO 260
333*           267  J = FIND(CARD,COL,EQUVEC)
334*                IF (J .EQ. -1)  GO TO 263
335*                M = NUMBER(CARD,COL,DATFIL,ZERO)
336*                DATFIL = DATFIL - 1
337*                IF (DATFIL .LT. 0)  DATFIL = 0
338*                COL = COL - 1
339*                GO TO 260
340*          C
341*          C     STAT FILE CARD
342*          C
343*           270  M=NXTCHR(CARD,COL)                                                         •NEW
344*           271  IF(M.EQ.'I')GO TO 278                                                      •NEW
345*                IF(M.EQ.'U')GO TO 275                                                      •NEW
346*                IF(M.EQ.'F')GO TO 277                                                      •NEW
347*                IF(M.EQ.'O')GO TO 272                                                      •NEW
```

```
      IF(M.EQ.',')GO TO 270                                              *NEW
      IF(M.EQ.' ')GO TO 10                                               *NEW
  273 WRITE(6,755)                                                       *NEW
  755 FORMAT(' ERROR ON STATFILE CARD?')                                 *NEW
      GO TO 10                                                           *NEW
  272 J=FIND(CARD,COL,SLASH)                                             *NEW
      IF(J.EQ.-1)GO TO 273                                               *NEW
      INITM=1                                                            *NEW
  274 M=NATCHR(CARD,COL)                                                 *NEW
      IF(M.EQ.',')GO TO 274                                              *NEW
      IF(M.EQ.'F')GO TO 277                                              *NEW
      IF(M.NE.'U')GO TO 271                                              *NEW
  275 J=FIND(CARD,COL,EQUVEC)                                            *NEW
      IF(J.EQ.-1)GO TO 273                                               *NEW
      M=NUMBER(CARD,COL,SAVTAP,ZERO)                                     *NEW
      COL=COL-1                                                          *NEW
      GO TO 274                                                          *NEW
  277 J=FIND(CARD,COL,EQUVEC)                                            *NEW
      IF(J.EQ.-1)GO TO 273                                               *NEW
      M=NUMBER(CARD,COL,STAFIL,ZERO)                                     *NEW
      COL=COL-1                                                          *NEW
      STAFIL=STAFIL-1                                                    *NEW
      IF(STAFIL.LT.0)STAFIL=0                                            *NEW
      GO TO 274                                                          *NEW
  278 J=FIND(CARD,COL,SLASH)                                             *NEW
      IF(J.EQ.-1)GO TO 273                                               *NEW
  279 M=NATCHR(CARD,COL)                                                 *NEW
      IF(M.EQ.',')GO TO 279                                              *NEW
      IF(M.EQ.'F')GO TO 282                                              *NEW
      IF(M.NE.'U')GO TO 271                                              *NEW
  281 J=FIND(CARD,COL,EQUVEC)                                            *NEW
      IF(J.EQ.-1)GO TO 273                                               *NEW
      M=NUMBER(CARD,COL,INUNIT,ZERO)                                     *NEW
      COL=COL-1                                                          *NEW
      GO TO 279                                                          *NEW
  282 J=FIND(CARD,COL,EQUVEC)                                            *NEW
      IF(J.EQ.-1)GO TO 273                                               *NEW
      M=NUMBER(CARD,COL,INFILE,ZERO)                                     *NEW
      COL=COL-1                                                          *NEW
      GO TO 279                                                          *NEW
C                                                                        *NEW
C     SUBCLASSES CARD--USE THE MEANS FOR THESE SUBCLASSES FROM THE       *NEW
C     STAT FILE FOR INITIAL MEANS                                        *NEW
C                                                                        *NEW
  283 NOSUB2=NUMBER(CARD,COL,SUBVEC,NOSUB2)                              *NEW
      GO TO 10                                                           *NEW
C                                                                        *NEW
C     MINIMUM POPULATION FOR STATISTICS PASS.                            *NEW
C                                                                        *NEW
  284 M=NUMBER(CARD,COL,PMIN,ZERO)                                       *NEW
      GO TO 10                                                           *NEW
C                                                                        *-19
C     *END* CARD
C
  280 CONTINUE
      IF (NOFEAT .GT. 0)  GO TO 285
      NOFEAT=30
      DO 261 I=1,30
```

B&L 45

```
      IF(M.EQ.',')GO TO 270                                              *NEW
      IF(M.EQ.' ')GO TO 10                                               *NEW
  273 WRITE(6,755)                                                       *NEW
  755 FORMAT(' ERROR ON STATFILE CARD:')                                 *NEW
      GO TO 10                                                           *NEW
  272 J=FIND(CARD,COL,SLASH)                                             *NEW
      IF(J.EQ.-1)GO TO 273                                               *NEW
      INITM=1                                                            *NEW
  274 M=NXTCHR(CARD,COL)                                                 *NEW
      IF(M.EQ.',')GO TO 274                                              *NEW
      IF(M.EQ.'F')GO TO 277                                              *NEW
      IF(M.NE.'U')GO TO 271                                              *NEW
  275 J=FIND(CARD,COL,EQUVEC)                                            *NEW
      IF(J.EQ.-1)GO TO 273                                               *NEW
      M=NUMBER(CARD,COL,SAVTAP,ZERO)                                     *NEW
      COL=COL-1                                                          *NEW
      GO TO 274                                                          *NEW
  277 J=FIND(CARD,COL,EQUVEC)                                            *NEW
      IF(J.EQ.-1)GO TO 273                                               *NEW
      M=NUMBER(CARD,COL,STAFIL,ZERO)                                     *NEW
      COL=COL-1                                                          *NEW
      STAFIL=STAFIL-1                                                    *NEW
      IF(STAFIL.LT.0)STAFIL=0                                            *NEW
      GO TO 274                                                          *NEW
  278 J=FIND(CARD,COL,SLASH)                                             *NEW
      IF(J.EQ.-1)GO TO 273                                               *NEW
  279 M=NXTCHR(CARD,COL)                                                 *NEW
      IF(M.EQ.',')GO TO 279                                              *NEW
      IF(M.EQ.'F')GO TO 282                                              *NEW
      IF(M.NE.'U')GO TO 271                                              *NEW
  281 J=FIND(CARD,COL,EQUVEC)                                            *NEW
      IF(J.EQ.-1)GO TO 273                                               *NEW
      M=NUMBER(CARD,COL,INUNIT,ZERO)                                     *NEW
      COL=COL-1                                                          *NEW
      GO TO 279                                                          *NEW
  282 J=FIND(CARD,COL,EQUVEC)                                            *NEW
      IF(J.EQ.-1)GO TO 273                                               *NEW
      M=NUMBER(CARD,COL,INFILE,ZERO)                                     *NEW
      COL=COL-1                                                          *NEW
      GO TO 279                                                          *NEW
C                                                                        *NEW
C     SUBCLASSES CARD--USE THE MEANS FOR THESE SUBCLASSES FROM THE       *NEW
C     STAT FILE FOR INITIAL MEANS                                        *NEW
C                                                                        *NEW
  283 NOSUB2=NUMBER(CARD,COL,SUBVEC,NOSUB2)                              *NEW
      GO TO 10                                                           *NEW
C                                                                        *NEW
C     MINIMUM POPULATION FOR STATISTICS PASS.                            *NEW
C                                                                        *NEW
  284 M=NUMBER(CARD,COL,PMIN,ZERO)                                       *NEW
      GO TO 10                                                           *NEW
C                                                                        *-19
C     *END* CARD
C
  280 CONTINUE
      IF (NOFEAT .GT. 0)  GO TO 285
      NOFEAT=30
      DO 261 I=1,30
```

B&45

```
      261 FEIVEC(I)=I
      285 CONTINUE
C
C     PRINT USER REQUEST
C
      WRITE(6,660)
      WRITE(6,670)ISTOP,NMIN,KRN,MAP,MAXCLS,NCLASS,
     *   (FETVEC(I),I=1,NOFEAT)
      WRITE(6,680)DLMIN,STDMAX
      IF(SPTMIG.EQ.1)WRITE(6,690)SEP
       IF(SIMERP.EQ.1)WRITE(6,695)
      IF(ICHN.EQ.1) WRITE(6,710)CHNTMS
      IF(PUNCH.EQ.1) WRITE(6,700)
       IF(ORDER.EQ.1)WRITE(6,715)
       IF(MAPFMT.EQ.1)WRITE(6,720)
       IF(MAPFMT.EQ.2)WRITE(6,725)
      IF(ERPKEY.EQ.1)WRITE(6,730)
      IF (NOFEAT .GT. NMIN)  WRITE(6,740)
      RETURN
C
  480 FORMAT(A6,4X,62A1)
  490 FORMAT(' INVALID INPUT CARD--IGNORED'/I5,A6,4X,62A1)
  500 FORMAT(10X,10A6)
  510 FORMAT(10X,2A6)
  550 FORMAT(5X,A6,4X,62A1)
  630 FORMAT(///' INPUT SUMMARY'///)
  640 FORMAT(' CHANNELS CANNOT BE CHANGED UNTIL THIS EXECTUION OF ISOCLS
     * IS COMPLETED')
  650 FORMAT(' NO. OF CLASSES CANNOT BE CHANGED UNTIL THIS EXECUTION OF
     * ISOCLS IS COMPLETED')
  660 FORMAT(///' YOU HAVE SELECTED THE FOLLOWING PARAMETER VALUES AND
     * OPTIONS'///)
  670 FORMAT(' STOP AFTER',I5,' ITERATION(S)' /
     *       ' ALLOW A MINIMUM OF',I4,' PIXELS PER CLUSTER' /
     *       ' PRINT A CLUSTER SUMMARY EVERY',I5,' ITERATION(S)'/
     *       ' PRINT A CLUSTER MAP EVERY',I5,' ITERATION(S)' /
     *       ' ALLOW A MAXIMUM OF',I5,' CLUSTERS PER CLASS' /
     *       ' THE STATISTICS FILE WILL BE WRITTEN AFTER',I4,' CLASS(ES)
     * HAVE BEEN CLUSTERED' /
     *       ' CHANNELS ARE---', 3013 )
  680 FORMAT(' DLMIN =', F7.3 /' STDMAX=',F7.3)
  690 FORMAT(' SEP   =', F7.3)
  695  FORMAT(1X,'SIMULATE ERIPS CLUSTERING ALGORITHM')
  700 FORMAT(' PUNCH THE MODULE STAT DECK')
  710 FORMAT(' CHAIN CLUSTERS WHICH ARE',F7.3,' UNITS APART')
  715  FORMAT(1X,'ORDER COLOR KEYS')
  720  FORMAT(1X,'WRITE A CLUSTER MAP OUTPUT TAPE IN UNIVERSAL FORMAT')
  725  FORMAT(1X,'WRITE A CLUSTER MAP OUTPUT TAPE IN LARSYS II FORMAT')
  730 FORMAT(' WRITE AN ERIPS INTERFACE TAPE')
  740 FORMAT(///' **WARNING** NMIN IS LESS THAN NO. OF CHANNELS,COVARIANC
     *ES WILL NOT BE INVERTIBLE')
      END                                                    SETU3100

) OF COMPILATION:        NO  DIAGNOSTICS.
/     SYMBOLIC                                  01 FEB 77  05:07:15    0  03360104     14    374
/  CODE    RELOCATABLE                          01 FEB 77  05:07:15    1  03375402     32      1
                                                                       0  03372326     14    122
```


B-9 16

```
406*          261 FETVEC(I)=I
407*          285 CONTINUE
408*          C*
409*          C*     PRINT USER REQUEST
410*          C*
411*              WRITE(6,660)
412*              WRITE(6,670)ISTOP,NMIN,KRN,MAP,MAXCLS,NCLASS,
413*             *    (FETVEC(I),I=1,NOFEAT)
414*              WRITE(6,680)DLMIN,STDMAX
415*              IF(SPTRIG.EQ.1)WRITE(6,690)SEP
416*               IF(SIMERP.EQ.1)WRITE(6,695
417*              IF(ICHN.EQ.1) WRITE(6,710)CHNTMS
418*              IF(PUNCH.EQ.1) WRITE(6,700)
419*               IF(ORDER.EQ.1)WRITE(6,715)
420*               IF(MAPFMT.EQ.1)WRITE(6,720)
421*               IF(MAPFMT.EQ.2)WRITE(6,725)
422*              IF(ERPKEY.EQ.1)WRITE(6,730)
423*              IF (NOFEAT .GT. NMIN)  WRITE(6,740)
424*              RETURN
425*          C
426*          480 FORMAT(A6,4X,62A1)
427*          490 FORMAT(' INVALID INPUT CARD--IGNORED'/T5,A6,4X,62A1)
428*          500 FORMAT(10X,10A6)
429*          510 FORMAT(10X,2A6)
430*          550 FORMAT(5X,A6,4X,62A1)
431*          630 FORMAT(//' INPUT SUMMARY'//)
432*          640 FORMAT(' CHANNELS CANNOT BE CHANGED UNTIL THIS EXECTUION OF ISOCL
433*             * IS COMPLETED')
434*          650 FORMAT(' NO. OF CLASSES CANNOT BE CHANGED UNTIL THIS EXECUTION OF
435*             * ISOCLS IS COMPLETED')
436*          660 FORMAT(///' YOU HAVE SELECTED THE FOLLOWING PARAMETER VALUES AND
437*             * OPTIONS'//)
438*          670 FORMAT(' STOP AFTER',I5,' ITERATION(S)' /
439*             *       ' ALLOW A MINIMUM OF',I4,' PIXELS PER CLUSTER' /
440*             *       ' PRINT A CLUSTER SUMMARY EVERY',I5,' ITERATION(S)'/
441*             *       ' PRINT A CLUSTER MAP EVERY',I5,' ITERATION(S)' /
442*             *       ' ALLOW A MAXIMUM OF',I5,' CLUSTERS PER CLASS' /
443*             *       ' THE STATISTICS FILE WILL BE WRITTEN AFTER',I4,' CLASS(ES)
444*             * HAVE BEEN CLUSTERED' /
445*             *       ' CHANNELS ARE---', 3013 )
446*          680 FORMAT(' DLMIN =', F7.3 / ' STDMAX=',F7.3)
447*          690 FORMAT(' SEP   =', F7.3)
448*          695   FORMAT(1X,'SIMULATE ERIPS CLUSTERING ALGORITHM')
449*          700 FORMAT(' PUNCH THE MODULE STAT DECK')
450*          710 FORMAT(' CHAIN CLUSTERS WHICH ARE',F7.3,' UNITS APART')
451*          715   FORMAT(1X,'ORDER COLOR KEYS')
452*          720   FORMAT(1X,'WRITE A CLUSTER MAP OUTPUT TAPE IN UNIVERSAL FORMAT')
453*          725   FORMAT(1X,'WRITE A CLUSTER MAP OUTPUT TAPE IN LARSYS II FORMAT')
454*          730 FORMAT(' WRITE AN ERIPS INTERFACE TAPE')
455*          740 FORMAT(//' **WARNING** NMIN IS LESS THAN NO. OF CHANNELS,COVARIANC
456*             *CES WILL NOT BE INVERTIBLE')
457*              END                                                    SETU3100

) OF COMPILATION:        NO  DIAGNOSTICS.
/      SYMBOLIC                                    01 FEB 77  05:07:15   0  03360104    14   374
/ _ CODE___ RELOCATABLE                            01 FEB 77  05:07:15   1  03375602    32     1
                                                                         0  03372326    14   122
```

APPENDIX C

APPENDIX C

```
GETSTA,GETSTA                                                    01 FEB 77
108 FORTRAN V EXEC II LEVEL 25A -(EXEC8 LEVEL E12D10C10A)
PILATION WAS DONE ON 01 FEB 77 AT 05:08:47

MNVEC,MEANS,COVAR,ITRIG)

UTINE GETSTA    ENTRY POINT 000656

GE USED: CODE(1) 000790; DATA(0) 000276; BLANK_COMMON(2) 000000

NAL REFERENCES (BLOCK, NAME):
    FSBSFL
    CMERR
    EXIT
    NMEMS
    NMOUS
    NIOZS
    NMOUS
    NIOIS
    SORT
    MEANS

GE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

    000131 100F      0000    000161 110F      0001    000100 1346      0001    000112 1456      0001    000130 14
    000155 1726      0000    000200 200F      0001    000201 2016      0000    000203 210F      0000    000213 22
    000237 2236      0001    000264 2306      0001    000272 2346      0000    000217 235F      0001    000314 24
    000332 25L       0001    000355 2636      0001    000360 2666      0001    000376 2766      0001    000343 30
    000415 3106      0001    000424 3136      0001    000521 3406      0001    000542 3466      0001    000552 35
    000571 3626      0001    000061 5L        0001    000409 50L       0001    000121 7L        0000    000142 70
    000145 77L       0001    000162 9L        0000 R  000079 BLANK     0000 I  000000 CMNVC1    0000 I  000105 00
I   000036 00MVEC    0000 I  000130 END       0000 I  000109 I         0000 I  000111 IC        0000 I  000123 IE
I   000126 II        0000    000223 INJPS     0000 I  000122 ISTART    0000 I  000076 ISTATI    0000 I  000113 J
I   000115 JA        0000 I  000121 JJ        0000 I  000120 JJJ       0000 I  000114 JK        0000 I  000116 K
I   000117 L         0000 I  000124 LOOPCT    0000 I  000125 LOOPCI    0000 I  000107 MB        0000 I  000110 MF
I   000112 N         0000 I  000101 NCHAN     0000 I  000075 NF        0000 I  000077 NOCLS     0000 I  000102 NO
I   000100 NOSUB     0000 I  000127 START     0000 I  000103 TOTVRT    0000 I  000106 VARSIZ

    1*                SUBROUTINE GETSTA(UNIT,FILE,MENS,STDEV,NOSUBZ,SUBVEC,NOCHAN,
    2*               . CMNVEC,MEANS,COVAR,ITRIG)                                    0.E0
    3*          C                                                                   0.-1
    4*          C*****************************************************************
    5*          C
    6*          C     SUBROUTINE GETSTA RETRIEVES THE MEANS AND STANDARD DEVIATIONS
    7*          C    . FROM A STATISTICS FILE IN THE LANSYS 'SAVTAP' FORMAT.
    8*          C
    9*          C     INPUT ARGUMENTS:
   10*          C
```

```
GETSTA,GETSTA                                                    01 FEB 77
108 FORTRAN V EXEC II LEVEL 25A  -(EXEC8 LEVEL E12D10D10A)
PILATION WAS DONE ON 01 FEB 77 AT 05:08:47

CHNVEC,MEANS,COVAR,ITRIG)

UTINE GETSTA    ENTRY POINT 000656

GE USED: CODE(1) 000740; DATA(0) 000276; BLANK COMMON(2) 000000

NAL REFERENCES (BLOCK, NAME):
    F$$SFL
    CMERR
    EXIT
    NMEMS
    NMOUS
    NIO2S
    NMOUS
    NIO1S
    SORT
    NERR3S

GE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

   000131 100F      0000   000161 110F      0001   000100 1346      0001   000112 1456      0001   000130 14
   000155 1726      0000   000200 200F      0001   000201 2016      0000   000203 210F      0000   000213 22
   000237 2236      0001   000264 230G      0001   000272 2346      0000   000217 235F      0001   000314 24
   000332 25L       0001   000355 263G      0001   000360 266G      0001   000376 276G      0001   000343 30
   000415 310G      0001   000424 313G      0001   000521 3406      0001   000542 3466      0001   000552 35
   000571 3626      0001   000061 5L        0001   000409 50L       0001   000121 7L        0000   000192 70
   000145 77L       0001   000162 9L        0000 R 000074 BLANK     0000 I 000000 CHNVCI    0000 I 000105 00
I  000036 DUMVEC    0000 I 000130 END       0000 I 000109 I         0000 I 000111 IC        0000 I 000123 IE
I  000126 II        0000   000223 INJPS     0000 I 000122 ISTART    0000 I 000076 ISTATI    0000 I 000113 J
I  000115 JA        0000 I 000121 JJ        0000 I 000120 JJJ       0000 I 000114 JK        0000 I 000116 K
I  000117 L         0000 I 000124 LOOPCT    0000 I 000125 LOOPCI    0000 I 000107 MB        0000 I 000110 MF
I  000112 N         0000 I 000101 NCHAN     0000 I 000075 NF        0000 I 000077 NOCLS     0000 I 000102 NO
I  000100 NOSUB     0000 I 000127 START     0000 I 000103 TOTVRT    0000 I 000106 VARSIZ

    1*            SUBROUTINE GETSTA(UNIT,FILE,MEMS,STDEV,NOSUB2,SUBVEC,NOCHAN,
    2*           * CHNVEC,MEANS,COVAR,ITRIG)                                    *LEN
    3*      C                                                                   **-1
    4*      C*****************************************************************
    5*      C
    6*      C     SUBROUTINE GETSTA RETRIEVES THE MEANS AND STANDARD DEVIATIONS
    7*      C     FROM A STATISTICS FILE IN THE LANSYS 'SAVTAP' FORMAT.
    8*      C
    9*      C     INPUT ARGUMENTS:
   10*      C
```

```
C     UNIT   - FORTRAN UNIT NUMBER FROM WHICH THE STATS ARE TO BE
C              RETRIEVED.
C     FILE   - FILE NO. ON 'UNIT' FROM WHICH THE STATS ARE TO BE
C              RETRIEVED.
C     NOCHAN - NO. OF CHANNELS REQUESTED FROM TRAINING SEGMENT.
C              NOCHAN LESS THAN OR = NO. OF CHANNELS ON STAT FILE
C     CHNVEC - ARRAY CONTAINING ACTUAL CHANNELS REQUESTED FROM
C              TRAINING SEGMENT. MUST BE A SUBSET OF CHANNELS
C              ON STAT FILE
C
C   ITRIG  - IF ITRIG=1, ST. DEV. WILL BE RETURNED ALONG WITH MEANS.
C
C   OUTPUT ARGUMENTS:
C
C     MENS   - ARRAY CONTAINING THE MEAN VECTORS FOR EACH SUBCLASS
C              ( A SUBSET OF THE CHANNELS MAY BE SELECTED, BUT NOT
C              A SUBSET OF THE SUBCLASSES).
C     STDEV  - ARRAY CONTAINING THE SUBSET OF ST. DEV. FOR REQUES-
C              TED CHANNELS IN EACH SUBCLASS
C              IN EACH SUBCLASS.
C     NOSUB  - NUMBER OF SUBCLASSES ON THE STAT FILE
C   CHNVEC - ARRAY CONTAINING ACTUAL CHANNELS REQUESTED FROM
C              TRAINING SEGMENT
C
C*************************************************************
C*************************************************************
C
C  **NOTE**  THE STORAGE ARRAYS PASSED TO THIS SUBROUTINE FOR THE
C            MEANS AND STANDARD DEVIATIONS SHOULD BE SINGLY DIMENSIONED
C            IN THE CALLING ROUTINE. ON OUTPUT THE ITEMS ARE STORED
C            AS FOLLOWS: (SAME FOR STDEV)
C
C                  MEANS(1)        - CHANNEL 1, SUBCLASS 1
C                        (2)       - CHANNEL 2, SUBCLASS 1
C                        3         - CHANNEL 3, SUBCLASS 1
C                        .         -          .
C                        .         -          .
C                       (NOCHAN)   -          NOCHAN,SUBCLASS 1
C                       (NOCHAN+1) - CHANNEL 1, SUBCLASS 2
C                       (NOCHAN+2) - CHANNEL 2, SUBCLASS 2
C                       (NOCHAN+3) - CHANNEL 3, SUBCLASS 2
C                                .   -
C                                .   -
C                       (2*NOCHAN) - CHANNEL NOCHAN OF SUBCLASS 2
C                                .
C                                .
C                                .
C                              ETC.
C                              THRU
C                       (NOCHAN*NOSUB)
C
C*************************************************************
C
      DIMENSION CHNVC1(30),DUMVEC(30),CHNVEC(1)
      DATA BLANK/' '/
       DIMENSION SUBVEC(1)
      DIMENSION MEANS(1),STDEV(1)
      IMPLICIT INTEGER(A-Z)
```

C2 49

```
C*
C*      UNIT   - FORTRAN UNIT NUMBER FROM WHICH THE STATS ARE TO BE
C*               RETRIEVED.
C*      FILE   - FILE NO. ON *UNIT* FROM WHICH THE STATS ARE TO BE
C*               RETRIEVED.
C*      NOCHAN - NO. OF CHANNELS REQUESTED FROM TRAINING SEGMENT.
C*               NOCHAN LESS THAN OR = NO. OF CHANNELS ON STAT FILE
C*      CHNVEC - ARRAY CONTAINING ACTUAL CHANNELS REQUESTED FROM
C*               TRAINING SEGMENT. MUST BE A SUBSET OF CHANNELS
C*               ON STAT FILE
C*
C*    ITRIG    - IF ITRIG=1, ST. DEV. WILL BE RETURNED ALONG WITH MEANS.
C*
C*    OUTPUT ARGUMENTS:
C*
C*       MENS    - ARRAY CONTAINING THE MEAN VECTORS FOR EACH SUBCLASS
C*                 ( A SUBSET OF THE CHANNELS MAY BE SELECTED, BUT NOT
C*                 A SUBSET OF THE SUBCLASSES).
C*       STDEV   - ARRAY CONTAINING THE SUBSET OF ST. DEV. FOR REQUES-
C*                 TED CHANNELS IN EACH SUBCLASS
C*                IN EACH SUBCLASS.
C*      NOSUB  - NUMBER OF SUBCLASSES ON THE STAT FILE
C*    CHNVEC - ARRAY CONTAINING ACTUAL CHANNELS REQUESTED FROM
C*               TRAINING SEGMENT
C*
C*****************************************************************
C*****************************************************************
C*
C*  **NOTE**  THE STORAGE ARRAYS PASSED TO THIS SUBROUTINE FOR THE
C*            MEANS AND STANDARD DEVIATIONS SHOULD BE SINGLY DIMENSIONED
C*            IN THE CALLING ROUTINE. ON OUTPUT THE ITEMS ARE STORED
C*            AS FOLLOWS: (SAME FOR STDEV)
C*
C*                  MEANS(1)      - CHANNEL 1, SUBCLASS 1
C*                       (2)      - CHANNEL 2, SUBCLASS 1
C*                        3       - CHANNEL 3, SUBCLASS 1
C*                        .       -          .
C*                        .       -          .
C*                     (NOCHAN)   -          NOCHAN,SUBCLASS 1
C*                     (NOCHAN+1) - CHANNEL 1, SUBCLASS 2
C*                     (NOCHAN+2) - CHANNEL 2, SUBCLASS 2
C*                     (NOCHAN+3) - CHANNEL 3, SUBCLASS 2
C*                                .
C*                                .
C*                     (2*NOCHAN) - CHANNEL NOCHAN OF SUBCLASS 2
C*                                .
C*                                .
C*                                .
C*                               ETC.
C*                               THRU
C*                     (NOCHAN*NOSUB)
C*
C*****************************************************************
C*
      DIMENSION CHNVC1(30),DUMVEC(30),CHNVEC(1)
      DATA BLANK/' '/
       DIMENSION SUBVEC(1)
      DIMENSION MEANS(1),STDEV(1)
      IMPLICIT INTEGER(A-Z)
```

C-2 49

```
      REAL MEANS,STDEV,COVAR(1965)
      REAL MENS(1)
      REWIND UNIT
      NF=FILE-1
      CALL FSBSFL(UNIT,NF,ISTAT)
      IF(ISTAT.EQ.0)GO TO 5
      WRITE(6,100)UNIT,FILE
  100 FORMAT(1A,'ERROR IN POSITIONING UNIT',I3,' TO FILE', I3)
      CALL CMERR
    5 CONTINUE
      READ(UNIT)NOCLS,NOSUB,NCHAN,NOFLD,TOTVRT,(CHNVC1(I),I=1,NCHAN)
C
C     DEFAULT ALL SUBCLASSES FROM STATISTICS FILE
C
      IF(NOSUB2.NE.0)GO TO 7
      DO 6 I=1,NOSUB
    6 SUBVEC(I)=I
      NOSUB2=NOSUB
    7 CONTINUE
      DO 77 I=1,NOSUB2
      IF (SUBVEC(I) .LE. NOSUB) GO TO 77
      WRITE(6,70)NOSUB
   70 FORMAT(/' REQUESTED SUBCLASS IS NOT ON STAT FILE. STAT FILE CONTA
     .NS ',I3,' SUBCLASSES')
      CALL CMERR
   77 CONTINUE
C
C     DEFAULT -- ALL CHANNELS FROM STAT FILE
C
      IF (NOCHAN .NE. 0)  GO TO 9
      DO 8  I=1,NCHAN
    8 CHNVEC(I) = CHNVC1(I)
      NOCHAN = NCHAN
    9 CONTINUE
C
C     READ PAST THE TRAINING FIELD INFORMATION
C
      DO 10 I=1,NOFLD
      READ(UNIT)DUM
      READ(UNIT)DUM
   10 CONTINUE
      READ(UNIT)DUM
      VARSIZ = NCHAN*(NCHAN+1)/2
      MB=1
      ME = NCHAN
      IC=1
      DO 30 I=1,NOSUB
      READ(UNIT) N,(COVAR(J),J=1,VARSIZ),(MEANS(J),J=MB,ME)
      IF(SUBVEC(IC).NE.I)GO TO 30
      IF(ITRIG.NE.1)GO TO 25
C
C     GET STANDARD DEVIATIONS
C
      JK=0
      DO 20 JA=1,NOCHAN
      JK= JK+ JA
      STDEV(MB+JA-1) = SQRT(COVAR(JK))
   20 CONTINUE
```

```
      REAL MEANS,STDEV,COVAR(465)
      REAL MENS(1)
      REWIND UNIT
      NF=FILE-1
      CALL FSBSFL(UNIT,NF,ISTAT)
      IF(ISTAT.EQ.0)GO TO 5
      WRITE(6,100)UNIT,FILE
  100 FORMAT(1X,'ERROR IN POSITIONING UNIT',I3,' TO FILE', I3)
      CALL CMERR
    5 CONTINUE
      READ(UNIT)NOCLS,NOSUB,NCHAN,NOFLD,TOTVRT,(CHNVC(I),I=1,NCHAN)
C
C     DEFAULT ALL SUBCLASSES FROM STATISTICS FILE
C
      IF(NOSUB2.NE.0)GO TO 7
      DO 6 I=1,NOSUB
    6 SUBVEC(I)=I
      NOSUB2=NOSUB
    7 CONTINUE
      DO 77 I=1,NOSUB2
      IF (SUBVEC(I) .LE. NOSUB) GO TO 77
      WRITE(6,70)NOSUB
   70 FORMAT(/' REQUESTED SUBCLASS IS NOT ON STAT FILE. STAT FILE CONTAI
     .NS ',I3,' SUBCLASSES')
      CALL CMERR
   77 CONTINUE
C
C     DEFAULT -- ALL CHANNELS FROM STAT FILE
C
      IF (NOCHAN .NE. 0) GO TO 9
      DO 8  I=1,NCHAN
    8 CHNVEC(I) = CHNVC(I)
      NOCHAN = NCHAN
    9 CONTINUE
C
C     READ PAST THE TRAINING FIELD INFORMATION
C
      DO 10 I=1,NOFLD
      READ(UNIT)DUM
      READ(UNIT)DUM
   10 CONTINUE
      READ(UNIT)DUM
      VARSIZ = NCHAN*(NCHAN+1)/2
      MB=1
      ME = NCHAN
       IC=1
      DO 30 I=1,NOSUB
      READ(UNIT) N,(COVAR(J),J=1,VARSIZ),(MEANS(J),J=MB,ME)
       IF(SUBVEC(IC).NE.I)GO TO 30
      IF(ITRIG.NE.1)GO TO 25
C
C     GET STANDARD DEVIATIONS
C
      JK=0
      DO 20 JA=1,NOCHAN
      JK= JK+ JA
      STDEV(MB+JA-1) = SQRT(COVAR(JK))
   20 CONTINUE
```

C-57

```
   25 CONTINUE
      IC = IC + 1
      MB = MB + NCHAN
      ME = ME + NCHAN
   30 CONTINUE
C
C     GET SUBSET OF MEANS
C     AND GET SUBSET OF ST. DEV.
C
      DO 50   J=1,NOCHAN
      DO 40   K=1,NCHAN
   40 IF (CHNVEC(J) .EQ. CHNVCI(K))  GO TO 50
      WRITE(6,110)CHNVEC(J),(CHNVCI(L),L=1,NCHAN)
  110 FORMAT(' CHANNEL NO. ',I2,' IS NOT ON TRAINING STAT FILE. CHANNELS
     1 ARE '/ 10X,30(I2,1X))
      CALL EXIT
   50 DUMVEC(J) = K
C
      JJJ = 0
      DO 60 K=1,NOSUBZ
      DO 60  J=1,NOCHAN
      JJ = DUMVEC(J)  + (K-1)*NCHAN
      JJJ = JJJ + 1
      MENS(JJJ) = MEANS(JJ)
   60 IF ( ITRIG .NE. 0) STDEV(JJJ) = STDEV(JJ)
       WRITE(6,200)
  200 FORMAT(//T57,'MEANS')
      ISTART = 1
      IEND = 12
      LOOPCT = NOCHAN/12
      LOOPCI = MOD(NOCHAN,12)
      IF (LOOPCI .GT. 0) LOOPCT = LOOPCT + 1
      IF (LOOPCT .EQ. 1) IEND = NOCHAN
      DO 240 II=1,LOOPCT
      START = ISTART
      END = IEND
      WRITE(6,210)(BLANK,CHNVEC(I),I=ISTART,IEND)
  210 FORMAT(/2A,'CLUSTER',5X,12(A1,'CH(',I2,')',2X))
      DO 230 J=1,NOSUBZ
      WRITE(6,220)SUBVEC(J),(MENS(K),K=START,END)
  220  FORMAT(5X,I2,7X,12(F7.2,1X))
      START = ISTART + NOCHAN*J
      END = IEND + NOCHAN*J
  230 CONTINUE
      WRITE(6,235)
  235 FORMAT(/)
      ISTART = IEND + 1
      IEND = IEND + ISTART - 1
      IF (IEND .GT. NOCHAN) IEND = NOCHAN
  240 CONTINUE
      RETURN
      END
```

```
NO OF COMPILATION:                 NO   DIAGNOSTICS.
.TA         SYMBOLIC                                  01 FEB 77  05:07:44   0  03497327   19   178
.TA  CODE   RELOCATABLE                               01 FEB 77  05:07:44   .  03495655   23     1
                                                                            0  03479253   19    55
```

```
127*        25 CONTINUE
126*           IC = IC + 1
129*           MB = MB + NCHAN
130*           ME = ME + NCHAN
131*        30 CONTINUE
132*     C
133*     C     GET SUBSET OF MEANS
134*     C     AND GET SUBSET OF ST. DEV.
135*     C
136*           DO 50   J=1,NOCHAN
137*           DO 40   K=1,NCHAN
138*        40 IF (CHNVEC(J) .EQ. CHNVC1(K))  GO TO 50
139*           WRITE(6,110)CHNVEC(J),(CHNVC1(L),L=1,NCHAN)
140*       110 FORMAT(' CHANNEL NO. ',I2,' IS NOT ON TRAINING STAT FILE. CHANNELS
141*          1 ARE '/ 10X,30(I2,1X))
142*           CALL EXIT
143*        50 DUMVEC(J) = K
144*     C
145*           JJJ = 0
146*           DO 60 K=1,NOSUB2
147*           DO 60   J=1,NOCHAN
148*           JJ = DUMVEC(J)   + (K-1)*NCHAN
149*           JJJ = JJJ + 1
150*           MENS(JJJ) = MEANS(JJ)
151*        60 IF ( ITRIG .NE. 0) STDEV(JJJ) = STDEV(JJ)
152*           WRITE(6,200)
153*       200 FORMAT(//T57,'MEANS')
154*           ISTART = 1
155*           IEND = 12
156*           LOOPCT = NOCHAN/12
157*           LOOPC1 = MOD(NOCHAN,12)
158*           IF (LOOPC1 .GT. 0) LOOPCT = LOOPCT + 1
159*           IF (LOOPCT .EQ. 1) IEND = NOCHAN
160*           DO 240 II=1,LOOPCT
161*           START = ISTART
162*           END = IEND
163*           WRITE(6,210)(BLANK,CHNVEC(I),I=ISTART,IEND)
164*       210 FORMAT(/2A,'CLUSTER',5X,I2(A1,'CH(',I2,')',2X))
165*           DO 230 J=1,NOSUB2
166*           WRITE(6,220)SUBVEC(J),(MENS(K),K=START,END)
167*       220   FORMAT(5X,I2,7X,12(F7.2,1X))
168*           START = ISTART + NOCHAN*J
169*           END = IEND + NOCHAN*J
170*       230 CONTINUE
171*           WRITE(6,235)
172*       235 FORMAT(/)
173*           ISTART = IEND + 1
174*           IEND = IEND + ISTART - 1
175*           IF (IEND .GT. NOCHAN) IEND = NOCHAN
176*       240 CONTINUE
177*           RETURN
178*           END

NO OF COMPILATION:        NO  DIAGNOSTICS.
:TA          SYMBOLIC                    01 FEB 77  05:07:44    0  03473?7     14    178
:TA  CODE    RELOCATABLE                 01 FEB 77  05:07:44    0  03475655    23      1
                                                                 0  03474253    14     55
```

APPENDIX D

APPENDIX D

```
ESCEN,DESCEN
1108 FORTRAN V EXEC II LEVEL 25A  -(EXEC8 LEVEL E1201001UA)                                         01 FEB 77
MPILATION WAS DONE ON 01 FEB 77 AT 05:07:34

OUTINE DESCEN     ENTRY POINT 000153

IGE USED: CODE(1) 000174; DATA(0) 000027; BLANK COMMON(2) 000000

INAL REFERENCES (BLOCK, NAME)

 1    NERR3S

GE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

   000012 60L        0001    000037 70L        0001    000072 75L        0001   000074 80L        0001   000134
   000005 INJP$      0000 I  000001 J          0000 I  000004 K          0000 R 000000 SAVE       0000 I 000002
 I 000003 SAVE2


              SUBROUTINE DESCEN(SCN,LNCAT,PTR1,PTR2)
              IMPLICIT INTEGER(A-X)
              DIMENSION PTR1(LNCAT),PTR2(LNCAT)
              REAL SCN(LNCAT),SAVE
              J=0
        60    J=J+1
              IF(J.GT.LNCAT)GO TO 90
              IF(J.EQ.LNCAT)GO TO 75
              IF(SCN(J).LT.SCN(J+1))GO TO 70
              GO TO 60
      C
        70    SAVE=SCN(J)
              SCN(J)=SCN(J+1)
              SCN(J+1)=SAVE
      C
              SAVE1=PTR1(J)
              PTR1(J)=PTR1(J+1)
              PTR1(J+1)=SAVE1
      C
              SAVE2=PTR2(J)
              PTR2(J)=PTR2(J+1)
              PTR2(J+1)=SAVE2
        75    K=J
        80    IF(K.EQ.1)GO TO 60
              IF(SCN(K).LT.SCN(K-1))GO TO 60
      C
              SAVE=SCN(K-1)
              SCN(K-1)=SCN(K)
              SCN(K)=SAVE
      C
              SAVE1=PTR1(K-1)
              PTR1(K-1)=PTR1(K)
              PTR1(K)=SAVE1
```

```
ESCEN,DESCEN                                                                    01 FEB 77
1108 FORTRAN V EXEC II LEVEL 25A  -(EXEC8 LEVEL E12.0100104)
MPILATION WAS DONE ON 01 FEB 77 AT 05:07:34

OUTINE DESCEN    ENTRY POINT 000153

GE USED: CODE(1) 000174; DATA(0) 000027; BLANK COMMON(2) 000000

INAL REFERENCES (BLOCK, NAME)
 1  NERR3S

GE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)
    000012 60L      0001  000037 70L      0001  000072 75L      0001  000074 80L      0001  000136
    000005 INJPS    0000 I 000001 J       0000 I 000004 K       0000 R 000000 SAVE    0000 I 000002
 I  000003 SAVE2

                    SUBROUTINE DESCEN(SCN,LNCAT,PTR1,PTR2)
                    IMPLICIT INTEGER(A-X)
                    DIMENSION PTR1(LNCAT),PTR2(LNCAT)
                    REAL SCN(LNCAT),SAVE
                    J=0
          60        J=J+1
                    IF(J.GT.LNCAT)GO TO 90
                    IF(J.EQ.LNCAT)GO TO 75
                    IF(SCN(J).LT.SCN(J+1))GO TO 70
                    GO TO 60
                C
          70        SAVE=SCN(J)
                    SCN(J)=SCN(J+1)
                    SCN(J+1)=SAVE
                C
                    SAVE1=PTR1(J)
                    PTR1(J)=PTR1(J+1)
                    PTR1(J+1)=SAVE1
                C
                    SAVE2=PTR2(J)
                    PTR2(J)=PTR2(J+1)
                    PTR2(J+1)=SAVE2
          75        K=J
          80        IF(K.EQ.1)GO TO 60
                    IF(SCN(K).LT.SCN(K-1))GO TO 60
                C
                    SAVE=SCN(K-1)
                    SCN(K-1)=SCN(K)
                    SCN(K)=SAVE
                C
                    SAVE1=PTR1(K-1)
                    PTR1(K-1)=PTR1(K)
                    PTR1(K)=SAVE1
```

P-153

```
C
      SAVE2=PTR2(K-1)
      PTR2(K-1)=PTR2(K)
      PTR2(K)=SAVE2
      K=K-1
      GO TO 80
   90 CONTINUE
      RETURN
      END
```

END OF COMPILATION: NO DIAGNOSTICS.

```
C
      SAVE2=PTR2(K-1)
      PTR2(K-1)=PTR2(K)
      PTR2(K)=SAVE2
      K=K-1
      GO TO 80
   90 CONTINUE
      RETURN
      END
```

ND OF COMPILATION: NO DIAGNOSTICS.

APPENDIX E

APPENDIX E

```
PL:T.PSPLIT                                                                    01 FEB 77
108 FORTRAN V EXEC II LEVEL 25A  -(EXEC8 LEVEL E:2010010A)
PILATION WAS DONE ON 01 FEB 77 AT 05:07:36

UTINE PSPLIT    ENTRY POINT 000490

GE USED: CODE(1) 000500; DATA(0) 000077; BLANK COMMON(2) 000000

ON BLOCKS:

   PASS   000467

NAL REFERENCES (BLOCK, NAME)

    RREAD
    RWRITE
    CLOCK
    NWDUS
    NIO2S
    SQRT
    NERR3S

GE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

    000303 IC5L     0001   000307 110L     0001   000056 116G     0001   000060 122G     0001   000170 1AC
    000200 165G     0001   000203 171G     0001   000076 20L      0001   000253 210G     0001   000357 233
    000363 237G     0001   000414 247G     0001   000123 25L      0001   000415 252G     0000   000016 30F
    000143 40L      0001   000220 70L      0000 I 000005 ADRES1   0000 I 000006 ADRES2   0003 I 000115 BEG
  I 000116 BEGIN2   0003   000117 BEGIN3   0003   000016 CHNTHS   0003   000426 CMNVC    0003   000120 CLS
  R 000001 DIST     0003   000005 DLMIN    0003   000465 ERCOMP   0003   000137 FETVEC   0000 I 000003 I
    000114 IBEGIN   0000 I 000007 ICCT     0003   000017 ICHAIN   0003   000015 ICHN     0003   000326 INF
    000327 INITM    0000   000030 INJPS    0003   000325 INUNIT   0003   000324 IORDER   0003   000122 IPT
  I 000010 IRC      0003 I 000011 IRD      0003   000273 ISOKEY   0000 I 000012 ISTAT    0000 I 000011 IWR
  I 000004 J        0000 I 000014 K        0000 I 000013 KK       0003 I 000012 KPTS     0003   000003 KRN
  I 000001 LNCAT    0003   000007 MAP      0003   000274 MAPFMT   0003   000275 MAPKEY   0003 I 000136 MAX
    000125 NCLASS   0003   000002 NMIN     0003   000464 NOCHAN   0003   000132 NOCL     0003   000120 NOC
  I 000135 NOFEAT   0003   000121 NOFLD    0003 I 000013 NOPTS    0003   000466 NOSEW    0003   000425 NOS
    000133 NVRT     0003   000113 NWDS     0003   000134 NXTCLS   0003   000322 PERCEN   0003   000330 PMI
    000014 PUNCH    0000 R 000002 RNU      0000 R 000000 SDIST    0003   000006 SEP     0003   000270 SEQ
    000323 SINERP   0003   000010 SPTRIG   0003   000272 STATKY   0003   000004 STDMAX   0003   000000 STO
    000311 SUBVEC   0003   000175 SYMMTX   0000 I 000015 TIME     0003   000130 TOTFLD   0003   000129 TOT
    000127 TOTSUB   0003   000131 TOTVRT   0003   000123 TOTWRD   0003   000271 VARSIZ

1•      C
2•            SUBROUTINE PSPLIT(MEANS,STDEV,N,CLD,C,IPLACE,AVP,AMN,MEN)
3•      C
4•            INCLUDE COMBAS,LIST
4•           PARAMETER MAXPOP=60,MXFET1=30
4•           COMMON/PASS/STOP,LNCAT,NMIN,KRN,STDMAX,DLMIN,SEP,
4•          •                    MAP,SPTRIG,   IRD,  KPTS, NOPTS, PUNCH,
4•          •              ICHN,CHNTHS,ICHAIN(MAXPOP),NWDS,IBEGIN,BEGIN1,
```

```
PLIT,PSPLIT                                                                    01 FEB 77
100 FORTRAN V EXEC II LEVEL 25A -(EXEC8 LEVEL E:2010010A)
PILATION WAS DONE ON 01 FEB 77 AT 05:07:36

UTINE PSPLIT    ENTRY POINT 000450

GE USED: CODE(1) 000500; DATA(0) 000072; BLANK COMMON(2) 000000

ON BLOCKS:
   PASS   000467

NAL REFERENCES (BLOCK, NAME)
   RREAD
   RWRITE
   CLOCK
   NWDUS
   NIO2S
   SQRT
   NERR3S

GE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)
   000303 105L    0001  000307 110L    0001  000056 116G    0001  000060 122G    0001  000170 1AG
   000200 165G    0001  000203 171G    0001  000076 20L     0001  000253 210G    0001  000357 233
   000363 237G    0001  000414 247G    0001  000123 25L     0001  000415 252G    0000  000016 30F
   000143 40L     0001  000220 70L     0000 I 000005 ADRES1   0000 I 000006 ADRES2   0003 I 000115 BEG
 I 000116 BEGIN2  0003  000117 BEGIN3  0003  000016 CHNTHS   0003  000426 CHNVC   0003  000120 CLS
 R 000001 DIST    0003  000005 DLMIN   0003  000465 ERCOMP   0003  000137 FETVEC  0000 I 000003 I
   000114 IBEGIN  0000 I 000007 ICCT    0003  000017 ICHAIN  0003  000015 ICHN    0003  000326 INF
   000327 INITM   0000  000030 INJPS   0003  000325 INUNIT  0003  000324 IORDER  0003  000122 IPT
 I 000010 IRC     0003 I 000011 IRD     0003  000273 ISOKEY  0000 I 000012 ISTAT   0000 I 000011 ISR
 I 000004 J       0000 I 000014 K       0000 I 000013 KK      0003 I 000012 KPTS    0003  000003 KRN
 I 000001 LNCAT   0003  000007 MAP     0003  000274 MAPFMT  0003  000275 MAPKEY  0003 I 000136 MAX
   000125 NCLASS  0003  000002 NMIN    0003  000464 NOCHAN  0003  000132 NOCL    0003  000120 NOC
 I 000135 NOFEAT  0003  000121 NOFLD   0003 I 000013 NOPTS   0003  000466 NOSEW   0003  000425 NOS
   000133 NVAT    0003  000113 NWDS    0003  000134 NXTCLS  0003  000322 PERCEN  0003  000330 PMI
   000014 PUNCH   0000 R 000002 RNG     0000 R 000000 SDIST   0003  000006 SEP     0003  000270 SEQ
   000323 SINERP  0003  000010 SPTRIG  0003  000272 STATKY  0003  000004 STDMAX  0003  000000 STO
   000311 SUBVEC  0003  000175 SYMMTX  0000 I 000015 TIME    0003  000130 TOTFLD  0003  000124 TOT
   000127 TOTSUB  0003  000131 TOTVRT  0003  000123 TOTWRD  0003  000271 VARSIZ

    1•      C
    2•            SUBROUTINE PSPLIT(MEANS,STDEV,N,CLD,C,IPLACE,AVP,AMN,MEN)
    3•      C
    4•            INCLUDE COMBK5,LIST
    4•           PARAMETER MAXPOP=60,MXFET1=30
    4•           COMMON/PASS/STOP,LNCAT,NMIN,KRN,STDMAX,DLMIN,SEP,
    4•          •                    MAP,SPTRIG,   IRD,  KPTS, NOPTS, PUNCH,
    4•          •              ICHN,CHNTHS,ICHAIN(MAXPOP),NWDS,IBEGIN,BEGIN1,
```

```
      *          BEGIN2,BEGIN3,CLSNAM,NOFLD,IPT,TOTRD,TOTPTS,
      *          NCLASS,NOCLS,TOTSUB,TOTFLD,TOTVRT,NOCL,NVRT
      *          ,NXTCLS,NOFEAT,MAXCLS,FETVEC(30),SYMMTX(60)
      *,VARSIZ,STATKY,ISUKEY,MAPFMT,MAPKEY,SEQUEN(20),PERCEN,SIMERR
      *  ,IORDER,INUNIT,INFILE,INITM,PMIN,SUBVEC(60),NOSUB2,CHNVC(30)
      *    ,NUCHAN,ERCOMP,NOSEQ
       IMPLICIT INTEGER (A-X)
       END
        DIMENSION C(NOFEAT,NOPTS),IPLACE(NOPTS),AMN(NOFEAT,MAXCLS)
        DIMENSION STDEV(NOFEAT,MAXCLS),CLD(MAXCLS,MAXCLS),N(MAXCLS)
       DIMENSION AVP(NOFEAT,MAXCLS),MEANS(30,60)
       REAL MEN(NOFEAT,LNCAT)
       REAL AMN,STDEV,AVP,SDIST,DIST,C,RND,MEANS
C
        DO 5 I=1,LNCAT
        N(I)=0
        DO 5 J=1,NOFEAT
        AMN(J,I)=0.0
       STDEV(J,I) = 0.0
    5   AVP(J,I)=0.0
C*
C*        ASSIGN DATA TO CLUSTERS
C*
        ADRES1=BEGIN1
        ADRES2=BEGIN2
        ICCT=NOPTS
        IRC=IRD
   20   IF(IRC.LE.1)ICCT=KPTS
        IF(IRD.EQ.0)GO TO 40
        IWRDS=NOFEAT*ICCT
        CALL RREAD(ADRES1,C,IWRDS,ISTAT)
        ADRES1=ADRES1+IWRDS
   25   IF(ISTAT.EQ.1)GO TO 25
        IF(ISTAT.EQ.0)GO TO 40
        IF(ISTAT.GE.0)GO TO 40
        WRITE(6,30)ISTAT
   30   FORMAT(' ERROR READING DRUM---ISTAT=',I8)
   40   CONTINUE
        DO 100 I=1,ICCT
        KK=1
        SDIST=10.0E+20
        DO 70 J=1,LNCAT
        DIST=0.0
        DO 50 K=1,NOFEAT
   50   DIST=DIST+ABS(MEANS(K,J)-C(K,I))
        IF(DIST-SDIST)60,70,70
   60   KK=J
        SDIST=DIST
   70   CONTINUE
   80   CONTINUE
        N(KK)=N(KK)+1
        IPLACE(I)=KK
        DO 90 K=1,NOFEAT
        AMN(K,KK)=AMN(K,KK)+C(K,I)
        AVP(K,KK)=AVP(K,KK)+C(K,I)*C(K,I)
   90   CONTINUE
  100   CONTINUE
        IF(IRD.EQ.0)GO TO 110
```


E-257

```
4*            *          BEGIN2,BEGIN3,CLSNAM,NOFLD,IPT,TOTRD,TOTPTS,
4*            *          NCLASS,NOCLS,TOTSUB,TOTFLD,TOTVRT,NOCL,NVRT
4*            *          ,NXTCLS,NOFEAT,MAXCLS,FETYPE(30),SYMMTX(60)
4*            *,VARSIZ,STATKY,ISUKEY,MAPFMT,MAPKEY,SEQUEN(20),PERCEN,SIMERR
4*            *  ,IORDER,INUNIT,INFILE,INITM,PMIN,SUBVEC(60),NOSUB2,CHNVC(30)      *NEW
4*            *  ,NOCHAN,ERCOMP,NOSEQ                                              *NEW
4*             IMPLICIT INTEGER (A-X)                                               *--1
4*             END
5*             DIMENSION C(NOFEAT,NOPTS),IPLACE(NOPTS),AMN(NOFEAT,MAXCLS)
6*             DIMENSION STDEV(NOFEAT,MAXCLS),CLD(MAXCLS,MAXCLS),N(MAXCLS)
7*            DIMENSION AVP(NOFEAT,MAXCLS),MEANS(30,60)
8*            REAL MEN(NOFEAT,LNCAT)
9*            REAL AMN,STDEV,AVP,SDIST,DIST,C,RND,MEANS
10*    C
11*            DO 5 I=1,LNCAT
12*            N(I)=0
13*            DO 5 J=1,NOFEAT
14*            AMN(J,I)=0.0
15*           STDEV(J,I) = 0.0
16*     5      AVP(J,I)=0.0
17*    C*
18*    C*       ASSIGN DATA TO CLUSTERS
19*    C*
20*            ADRES1=BEGIN1
21*            ADRES2=BEGIN2
22*            ICCT=NOPTS
23*            IRC=IRD
24*     20     IF(IRC.LE.1)ICCT=KPTS
25*            IF(IRD.EQ.0)GO TO 40
26*            IWRDS=NOFEAT*ICCT
27*            CALL RREAD(ADRES1,C,IWRDS,ISTAT)
28*            ADRES1=ADRES1+IWRDS
29*     25     IF(ISTAT.EQ.1)GO TO 25
30*            IF(ISTAT.EQ.0)GO TO 40
31*            IF(ISTAT.GE.0)GO TO 40
32*            WRITE(6,30)ISTAT
33*     30     FORMAT(' ERROR READING DRUM---ISTAT=',I5)
34*     40     CONTINUE
35*            DO 100 I=1,ICCT
36*            KK=1
37*            SDIST=10.0E+20
38*            DO 70 J=1,LNCAT
39*            DIST=0.0
40*            DO 50 K=1,NOFEAT
41*     50     DIST=DIST+ABS(MEANS(K,J)-C(K,I))
42*            IF(DIST-SDIST)60,70,70
43*     60     KK=J
44*            SDIST=DIST
45*     70     CONTINUE
46*     80     CONTINUE
47*            N(KK)=N(KK)+1
48*            IPLACE(I)=KK
49*            DO 90 K=1,NOFEAT
50*            AMN(K,KK)=AMN(K,KK)+C(K,I)
51*            AVP(K,KK)=AVP(K,KK)+C(K,I)*C(K,I)
52*     90     CONTINUE
53*     100    CONTINUE
54*            IF(IRD.EQ.0)GO TO 110
```

E-257

```
      CALL RWRITE(ADRES2,IPLACE,ICCT,ISTAT)
      ADRES2=ADRES2+ICCT
  105 IF(ISTAT.EQ.1)GO TO 105
  110 IRC=IRC-1
      IF(IRC.GT.0)GO TO 20
      CALL CLOCK(TIME)
      DO 130 K=1,LNCAT
      RND=FLOAT(N(K))
      DO 130 J=1,NOFEAT
      AMN(J,K)=AMN(J,K)/RND
      MEANS(J,K)=AMN(J,K)
  130 STDEV(J,K)=SQRT(AVP(J,K)/RND-AMN(J,K)*AMN(J,K))
C
      DO 140 I=1,LNCAT
      DO 140 J=1,NOFEAT
  140 MEN(J,I) = MEANS(J,I)
C
      RETURN
C
      END
```

```
NO OF COMPILATION:        NO  DIAGNOSTICS.
```

```
      CALL RWRITE(ADRES2,IPLACE,ICCT,ISTAT)
      ADRES2=ADRES2+ICCT
  105 IF(ISTAT.EQ.1)GO TO 105
  110 IRC=IRC-1
      IF(IRC.GT.0)GO TO 20
      CALL CLOCK(TIME)
      DO 130 K=1,LNCAT
      RND=FLOAT(N(K))
      DO 130 J=1,NOFEAT
      AMN(J,K)=AMN(J,K)/RND
      MEANS(J,K)=AMN(J,K)
  130 STDEV(J,K)=SQRT(AVP(J,K)/RND-AMN(J,K)*AMN(J,K))
C
      DO 140 I=1,LNCAT
      DO 140 J=1,NOFEAT
  140 MEN(J,I) = MEANS(J,I)
C
      RETURN
C
      END
```

```
NO OF COMPILATION:        NO  DIAGNOSTICS.
```

APPENDIX F

APPENDIX F

```
 000467 2776      0001   001174 280L      0000   003742 30F       0001   001215 310L      0001   000523
 000526 3156      0001   001221 320L      0000   004052 340F      0001   001253 350L      0001   000639
 004070 360F      0001   000204 40L       0001   001337 400L      0001   000723 4036      0001   001343
 001477 4256      0001   001501 430L      0001   000775 4336      0001   001601 4356      0001   001615
 001606 4416      0001   001636 4466      0001   001617 448L      0000   004105 450F      0001   001617
 004040 502F      0000   004037 503F      0001   001713 530L      0000   004124 540F      0001   001307
 001730 550L      0000   004140 560F      0001   001347 5736      0001   001361 6026      0001   001376
 001421 6166      0001   001450 6246      0001   001651 6516      0001   001576 6616      0001   000267
 001655 7036      0001   001661 7116      0001   002133 7546      0001   002134 7576      0001   000271
I 003715 ADRES1   0000 I 003716 ADRES2   0004 I 000073 ASAV      0004 I 000075 ASAVFL    0003 I 000115
I 000116 BEGIN2   0003 I 000117 BEGIN3   0004 I 000055 BMFILE    0004 I 000056 BMKEY     0003 I 000016
I 000426 CHNVC    0003 I 000120 CLSNAM   0004 I 000053 DATAPE    0004 I 000071 DATFIL    0000 L 003516
R 000102 DIJ      0000 R 000075 DIST      0003 R 000005 DLMIN     0000 R 000100 DMIN      0004 I 000067
I 000066 DRUMAD   0003 I 000465 ENCOMP   0004 I 000062 ERIPTP    0004 I 000063 ERPKEY    0000 R 000104
R 000103 ESUM     0003 I 000137 FETVEC   0004 I 000000 HEAD      0004 I 000057 MISFIL    0004 I 000060
I 003711 I        0003 I 000114 IBEGIN   0000 I 003717 ICCT      0003 I 000017 ICHAIN    0003 I 000015
I 003737 II       0000 I 003731 INC       0003 I 000001 INCAT     0003 I 000326 INFILE    0003 I 000327
  004165 INJPS    0000   004171 INJPS     0003 I 000325 INUNIT    0003 I 000324 IORDER    0003 I 000122
I 003720 IKC      0003 I 000011 IKD       0000 I 003707 ISEQ      0000 I 003517 ISGMA     0003 I 000273
I 003724 ISPLT    0000 I 003722 ISTAT     0000 I 003710 ISTOP     0000 I 003721 IWRDS     0000 I 003712
I 003740 JJ       0000 I 003719 K         0003 I 000135 KDIM      0000 I 003723 KK        0000 I 003741
I 003713 KKT      0003 I 000012 KPTS      0003 I 000003 KRN       0000 I 003730 KX        0000 I 003733
I 003732 LL       0003 I 000001 LNCAT     0003 I 000007 MAP       0004 I 000100 MAPFIL    0003 I 000277
I 000275 MAPKEY   0004 I 000054 MAPTAP    0004 I 000064 MAPUNT    0003 I 000136 MAXCLS    0000 R 000105
I 003726 NCAT     0003 I 000125 NCLASS    0004 I 000070 NHSTFI    0004 I 000075 NHSTUN    0003 I 000002
I 000464 NOCHAN   0003 I 000134 NOCL      0003 I 000126 NOCLS     0000 I 003735 NOCLST    0000 I 003736
I 003734 NOCOMB   0003 I 000135 NOFEAT    0004 I 000065 NOFILE    0003 I 000121 NOFLD     0003 I 000013
I 000466 NOSEQ    0003 I 000425 NOSUB2    0003 I 000133 NVRT      0003 I 000114 NWDS      0003 I 000139
I 000070 PAGSIZ   0003 R 000324 PERCEN    0003 I 000330 PMIN      0000 I 003613 PTR       0003 I 000014
R 000076 RND      0004 I 000054 SAVIAP    0004 I 000077 SCTRUN    0000 R 003515 SDIJ      0000 R 000074
R 000006 SEP      0003 I 000276 SEQUEN    0000 R 000000 SGMA      0003   000009 SGMIN     0003 I 000323
I 003725 SPLFIN   0003 I 000010 SPTRIG    0004 I 000072 STAFIL    0003 I 000272 STATKY    0003 R 000004
I 000000 STOP     0003 I 000331 SUBVEC    0003 I 000175 SYMMTX    0000 R 000077 TEST      0000 R 000101
I 000130 TOTFLD   0003 I 000129 TOTPTS    0003 I 000127 TOTSUB    0003 I 000131 TOTVRT    0003 I 000123
I 000961 TRFORM   0000 I 003727 TRIGI     0003 I 000271 VARSIZ
```

F-I 60

```
 1•            SUBROUTINE ISODAT(C,IPLACE,MEANS,N,STDEV,CLD,FLDINF,AVP,AMN)
 2•            INCLUDE COMBK5,LIST
 2•            PARAMETER MAXPOP=60,MXFET=30
 2•            COMMON/PASS/STOP,LNCAT,NMIN,KRN,STDMAX,DLMIN,SEP,
 2•           •                    MAP,SPTRIG,   IKD,  KPTS, NOPTS, PUNCH,
 2•           •             ICHN,CHNTHS,ICHAIN(MAXPOP),NWDS,IBEGIN,BEGIN1,
 2•           •            BEGIN2,BEGIN3,CLSNAM,NOFLD,IPT,TOTRD,TOTPTS,
 2•           •            NCLASS,NOCLS,TOTSUB,TOTFLD,TOTVRT,NOCL,NVRT
 2•           •            ,NXTCLS,NOFEAT,MAXCLS,FETVEC(30),SYMMTX(60)
 2•           •,VARSIZ,STATKY,ISOKEY,MAPFMT,MAPKEY,SEQUEN(20),PERCEN,SIMERP
 2•           •  ,IORDER,INUNIT,INFILE,INITN,PMIN,SUBVEC(60),NOSUB2,CHNVC(30)      •NE
 2•           •  ,NOCHAN,ENCOMP,NOSEQ                                              •NE
 2•            IMPLICIT INTEGER (A-A)                                              •0-1
 2•            END
 3•            INCLUDE COMBK6,LIST
 3•            COMMON/GLOBAL/HEAD(42),MAPTAP,DATAPE,SAVTAP,BMFILE,BMKEY,
 3•           •            MISFIL,MISKEY,TRFORM,ERIPTP,ERPKEY,MAPUNT,NOFILE,
 3•           •    DRUMAD,DRMWDS,PAGSIZ,DATFIL,STAFIL,ASAV,ASAVFL
 3•           •  ,NMSTUN,NMSTFI,SCTRUN,MAPFIL                                      •NE
```

```
 0000467 277G    0001   001174 280L    0000   003742 30F     0001   001215 310L    0001   000523
 0000526 315G    0001   001221 320L    0000   004052 340F    0001   001253 350L    0001   000639
 0000670 390F    0001   000204 40L     0001   001337 400L    0001   000723 403G    0001   001343
 0001477 425L    0001   001501 430L    0001   000775 433G    0001   001601 435L    0001   001615
 0001006 441G    0001   001036 4466    0001   001617 480L    0000   004105 490F    0001   001617
 0004040 502F    0000   004037 503F    0001   001713 530L    0000   004124 540F    0001   001307
 0001730 550L    0000   004140 560F    0001   001347 573G    0001   001361 602G    0001   001370
 0001421 616G    0001   001450 6246    0001   001551 651G    0001   001576 661G    0001   000267
 0001655 703G    0001   001661 711G    0001   002133 754G    0001   002134 757G    0001   000271
I 003715 ADRES1  0000 I 003716 ADRES2  0004 I 000073 ASAV    0004 I 000074 ASAVFL  0003 I 000115
I 000116 BEGIN2  0003 I 000117 BEGIN3  0004 I 000055 BMFILE  0004 I 000056 BMKEY   0003 I 000016
I 000426 CHNVC   0003 I 000120 CLSNAM  0004 I 000053 DATAPE  0004 I 000071 DATFIL  0000 L 003516
R 000102 DIJ     0000 R 000075 DIST    0003 R 000005 DLMIN   0000 R 000100 DMIN    0004 I 000067
I 000066 DRUMAD  0003 I 000465 ERCOMP  0004 I 000062 ERIPTP  0004 I 000063 ERPKEY  0000 R 000104
R 000103 ESUM    0003 I 000137 FETVEC  0004 I 000000 HEAD    0004 I 000057 HISFIL  0004 I 000060
I 003711 I       0003 I 000114 IBEGIN  0000 I 003717 ICCT    0003 I 000017 ICHAIN  0003 I 000015
I 003737 II      0000 I 003731 INC     0003 I 000001 INCAT   0003 I 000326 INFILE  0003 I 000327
  004165 INJPS   0000   004171 INJPS   0003 I 000325 INUNIT  0003 I 000324 IORDER  0003 I 000122
I 003720 IRC     0003 I 000011 IRD     0000 I 003707 ISEQ    0000 I 003517 ISGMA   0003 I 000273
I 003724 ISPLT   0000 I 003722 ISTAT   0000 I 003710 ISTOP   0000 I 003721 IWRDS   0000 I 003712
I 003740 JJ      0000 I 003714 K       0003 I 000135 KDIM    0000 I 003723 KK      0000 I 003741
I 003713 KKT     0003 I 000012 KPTS    0003 I 000003 KRN     0000 I 003730 KX      0000 I 003733
I 003732 LL      0003 I 000001 LNCAT   0003 I 000007 MAP     0004 I 000100 MAPFIL  0003 I 000279
I 000275 MAPKEY  0004 I 000054 MAPTAP  0004 I 000064 MAPUNT  0003 I 000136 MAXCLS  0000 R 000105
I 003726 NCAT    0003 I 000125 NCLASS  0004 I 000070 NHSTFI  0004 I 000075 NHSTUN  0003 I 000002
I 000464 NOCHAN  0003 I 000134 NOCL    0003 I 000126 NOCLS   0000 I 003735 NOCLST  0000 I 003736
I 003734 NOCOMB  0003 I 000135 NOFEAT  0004 I 000065 NOFILE  0003 I 000121 NOFLD   0003 I 000013
I 000466 NOSEQ   0003 I 000425 NOSUB2  0003 I 000133 NVRT    0003 I 000113 NWDS    0003 I 000134
I 000070 PAGSIZ  0003 R 000324 PERCEN  0003 I 000330 PMIN    0000 I 003613 PTR     0003 I 000014
R 000076 RND     0004 I 000053 SAVTAP  0004 I 000077 SCTRUN  0000 R 003515 SDIJ    0000 R 000074
R 000006 SEP     0003 I 000276 SEQUEN  0000 R 000000 SGMA    0003   000004 SGMIN   0003 I 000323
I 003725 SPLFIN  0003 I 000010 SPTRIG  0004 I 000072 STAFIL  0003 I 000272 STATKY  0003 R 000004
I 000000 STOP    0003 I 000331 SUBVEC  0003 I 000175 SYNTAX  0000 R 000077 TEST    0000 R 000101
I 000130 TOTFLD  0003 I 000129 TOTPTS  0003 I 000127 TOTSUB  0003 I 000131 TOTVRT  0003 I 000123
I 000061 TRFORM  0000 I 003727 TRIGI   0003 I 000271 VARSIZ
```

F-I 60

```
1•          SUBROUTINE ISODAT(C,IPLACE,MEANS,N,STDEV,CLD,FLDINF,AVP,AMN)
2•          INCLUDE COMBK5,LIST
2•          PARAMETER MAXPOP=60,MXFETI=30
2•          COMMON/PASS/STOP,LNCAT,NMIN,KRN,STDMAX,DLMIN,SEP,
2•         •                    MAP,SPTRIG,   IRD,  KPTS, NOPTS, PUNCH,
2•         •              ICHN,CHNTHS,ICHAIN(MAXPOP),NWDS,IBEGIN,BEGIN1,
2•         •             BEGIN2,BEGIN3,CLSNAM,NOFLD,IPT,TOTRD,TOTPTS,
2•         •             NCLASS,NOCLS,TOTSUB,TOTFLD,TOTVRT,NOCL,NVRT
2•         •             ,NXTCLS,NOFEAT,MAXCLS,FETVEC(30),SYNTAX(60)
2•         •,VARSIZ,STATKY,ISOKEY,MAPFMT,MAPKEY,SEQUEN(20),PERCEN,SIMERR
2•         •  ,IWORDER,IRUNIT,IRFILE,INITR,PMIN,SUBVEC(60),NOSUB2,CHNVC(30)      •NEW
2•         •  ,NOCHAN,ERCOMP,NOSEQ                                               •NEW
2•          IMPLICIT INTEGER (A-X)                                               ••-1
2•          END
3•          INCLUDE COMBK6,LIST
3•          COMMON/GLOBAL/HEAD(42),MAPTAP,DATAPE,SAVTAP,BMFILE,BMKEY,
3•         •             HISFIL,HISKEY,TRFORM,ERIPTP,ERPKEY,MAPUNT,NOFILE,
3•         •   DRUMAD,DRMWDS,PAGSIZ,DATFIL,STAFIL,ASAV,ASAVFL
3•         •  ,NHSTUN,NHSTFI,SCTRUN,MAPFIL                                       •NEW
```

```
      END                                                                   ISOD0020
      EQUIVALENCE (SGMIN,STDMAX)
      REAL MEANS,STDEV,STDMAX,SEP,AVP,C,AMN,SGMA,SDIST,DIST,RND,
     *TEST,DMIN,DLMIN,CLD,TIME,PERCEN,DIJ
       REAL ESUM,ESQT,MEAN(30,60),SDIJ                                      *NEW
      LOGICAL DEL
      DIMENSION AVP(NOFEAT,MAXCLS),ISGMA(MAXPOP)
      DIMENSION C(NOFEAT,NOPTS),IPLACE(NOPTS)
      DIMENSION AMN(NOFEAT,MAXCLS),SGMA(MAXPOP)
      DIMENSION MEANS(NOFEAT,MAXCLS),N(MAXCLS)
      DIMENSION STDEV(NOFEAT,MAXCLS),CLD(MAXCLS,MAXCLS)
      DIMENSION PTR(60)                                                     *NEW
      EQUIVALENCE (KDIM,NOFEAT),(LNCAT,INCAT)                               ISOD0070
      DEL=.FALSE.                                                           *--I
       ISEQ=0
      ISTOP=STOP
      DO 5 I=1,MAXCLS                                                       *--I
      N(I)=0
      DO 5 J=1,NOFEAT
      AMN(J,I)=0.0
    5 AVP(J,I)=0.0
      KKT=1                                                                 ISOD0120*--I
C*                                                                          ISOD0130
C*    ASSIGN DATA TO CLUSTERS                                               ISOD0140
C*                                                                          ISOD0150
   10 CONTINUE                                                              ISOD0160
      DO 15 K=1,LNCAT                                                       *NEW
      DO 15 J=1,NOFEAT                                                      *NEW
   15 MEAN(J,K) = MEANS(J,K)                                                *NEW
      IF (LNCAT.LE.1.AND.KKT.GT.1) GO TO 530                                ISOD0170
      ADRES1=BEGIN1
      ADRES2=BEGIN2
      ICCT=NOPTS                                                            ISOD0210
      IRC=IRD                                                               ISOD0220
   20 IF(IRC.LT.1)ICCT=KPTS                                                 ISOD0230
      IF (IRD.EQ.0) GO TO 40                                                ISOD0240
      IARDS=NOFEAT*ICCT                                                     ISOD0250
      CALL RREAD(ADRES1,C,IARDS,ISTAT)
      ADRES1=ADRES1+IARDS
   25 IF(ISTAT.EQ.1)GO TO 25
      IF(ISTAT.EQ.0)GO TO 40
      IF (ISTAT.GE.0) GO TO 40                                              ISOD0270
      WRITE (6,30)ISTAT                                                     ISOD0280
   30 FORMAT(' ERROR READING DRUM---ISTAT=',I4)                             ISOD0290
   40 CONTINUE                                                              ISOD0300
      DO 100 I=1,ICCT                                                       ISOD0310
      KK=1                                                                  ISOD0320
      IF (LNCAT.LT.2) GO TO 80                                              ISOD0330
      SDIST=10.0E+20                                                        ISOD0340
      DO 70 J=1,LNCAT                                                       ISOD0350
      DIST=0.0                                                              ISOD0360
      DO 50 K=1,NOFEAT                                                      ISOD0370
   50 DIST=DIST + ABS(MEANS(K,J)-C(K,I))                                    ISOD0380
      IF (DIST-SDIST) 60,70,70                                              ISOD0390
   60 KK=J                                                                  ISOD0400
      SDIST=DIST                                                            ISOD0410
   70 CONTINUE                                                              ISOD0420
   80 CONTINUE                                                              ISOD0430
```

Eng 61

```
      END                                                                  ••-1
      EQUIVALENCE (SGMIN,STDMAX)                                    ISOD0020
      REAL MEANS,STDEV,STDMAX,SEP,AVP,C,AMN,SGMA,SDIST,DIST,RND,
     •TEST,DMIN,DLMIN,CLD,TIME,PERCEN,DIJ
       REAL ESUM,ESQT,MEAN(30,60),SDIJ                                  •NEW
      LOGICAL DEL
      DIMENSION AVP(NOFEAT,MAXCLS),ISGMA(MAXPOP)
      DIMENSION C(NOFEAT,NOPTS),IPLACE(NOPTS)
      DIMENSION AMN(NOFEAT,MAXCLS),SGMA(MAXPOP)
      DIMENSION MEANS(NOFEAT,MAXCLS),N(MAXCLS)
      DIMENSION STDEV(NOFEAT,MAXCLS),CLD(MAXCLS,MAXCLS)
      DIMENSION PTR(60)                                                 •NEW
      EQUIVALENCE (KDIM,NOFEAT),(LNCAT,INCAT)                       ISOD0070
      DEL=.FALSE.                                                       ••-1
       ISEQ=0
      ISTOP=STOP
      DO 5 I=1,MAXCLS                                                   ••-1
      N(I)=0
      DO 5 J=1,NOFEAT
      AMN(J,I)=0.0
    5 AVP(J,I)=0.0
      KKT=1                                                         ISOD0120•••-1
C•                                                                  ISOD0130
C•    ASSIGN DATA TO CLUSTERS                                       ISOD0140
C•                                                                  ISOD0150
   10 CONTINUE                                                      ISOD0160
      DO 15 K=1,LNCAT                                                   •NEW
      DO 15 J=1,NOFEAT                                                  •NEW
   15 MEAN(J,K) = MEANS(J,K)                                            •NEW
      IF (LNCAT.LE.1.AND.KKT.GT.1) GO TO 530                        ISOD0170
      ADRES1=BEGIN1
      ADRES2=BEGIN2
      ICCT=NOPTS                                                    ISOD0210
      IRC=IRD                                                       ISOD0220
   20 IF(IRC.LE.1)ICCT=KPTS                                         ISOD0230
      IF (IRD.EQ.0) GO TO 40                                        ISOD0240
      INRDS=NOFEAT•ICCT                                             ISOD0250
      CALL RREAD(ADRES1,C,INRDS,ISTAT)
      ADRES1=ADRES1+INRDS
   25 IF(ISTAT.EQ.1)GO TO 25
      IF(ISTAT.EQ.0)GO TO 40
      IF (ISTAT.GE.0) GO TO 40                                      ISOD0270
      WRITE (6,30)ISTAT                                             ISOD0280
   30 FORMAT(' ERROR READING DRUM---ISTAT=',I4)                     ISOD0290
   40 CONTINUE                                                      ISOD0300
      DO 100 I=1,ICCT                                               ISOD0310
      KK=1                                                          ISOD0320
      IF (LNCAT.LT.2) GO TO 80                                      ISOD0330
      SDIST=10.0E+20                                                ISOD0340
      DO 70 J=1,LNCAT                                               ISOD0350
      DIST=0.0                                                      ISOD0360
      DO 50 K=1,NOFEAT                                              ISOD0370
   50 DIST=DIST + ABS(MEANS(K,J)-C(K,I))                            ISOD0380
      IF (DIST-SDIST) 60,70,70                                      ISOD0390
   60 KK=J                                                          ISOD0400
      SDIST=DIST                                                    ISOD0410
   70 CONTINUE                                                      ISOD0420
   80 CONTINUE                                                      ISOD0430
```

E-2 61

```
      N(KK)=N(KK)+1                                                    ISODO940
      IPLACE(I)=KK                                                     ISODO950
      DO 90 K=1,NOFEAT                                                 ISODO970
       AMN(K,KK)=AMN(K,KK)+C(K,I)
       AVP(K,KK)=AVP(K,KK)+C(K,I)*C(K,I)
   90 CONTINUE                                                         ISODO500
  100 CONTINUE                                                         ISODO510
      IF (IRD.EQ.0) GO TO 110                                          ISODO520
      CALL RWRITE(ADRES2,IPLACE,ICCT,ISTAT)
      ADRES2=ADRES2+ICCT
  105 IF(ISTAT.EQ.1)GO TO 105
  110 IRC=IRC-1                                                        ISODO540
      IF (IRC.GT.0) GO TO 20                                           ISODO550
      CALL CLOCK(TIME)                                                 ISODO560
      IF(MOD(KKT,KRN).EQ.0)WRITE(6,120)KKT,TIME
  120 FORMAT(' CUMULATIVE TIME AFTER ASSIGNING DATA TO CLUSTERS FOR ITE=
     +ATION',I4,' IS',F10.6)
      DO 130 K=1,LNCAT                                                 ISODO590
       RND=FLOAT(N(K))
      DO 130 J=1,NOFEAT                                                ISODO610
       AMN(J,K)=AMN(J,K)/RND
       MEANS(J,K)=AMN(J,K)
  130  STDEV(J,K)=SQRT(AVP(J,K)/RND-AMN(J,K)*AMN(J,K))
       IF(ERCOMP.NE.1)GO TO 135                                                    *NEW
       ESUM=0.0                                                                    *NEW
       DO 132 J=1,NOFEAT                                                           *NEW
       DO 132 K=1,LNCAT                                                            *NEW
       ESUM=ESUM+N(K)*(STDEV(J,K))**2/TOTPTS                                       *NEW
  132  CONTINUE                                                                    *NEW
       ESQT=SQRT(ESUM)                                                             *NEW
       WRITE(6,133)ESQT                                                            *NEW
  133  FORMAT(1X,////'  ERCOMP = ',F7.3/)                                          *NEW
C                                                                      ISODO640
C     CALCULATE DISTANCES BETWEEN CLUSTER CENTERS                      ISODO650
C                                                                      ISODO660
  135  CALL CLDIST(CLD,STDEV,MEANS)                                               *NEW
C                                                                      ISODO680**-1
C     IF STOP EQUALS ZERO RETURN AFTER INITIAL CLASSIFICATION
C
      IF(STOP.EQ.0)RETURN
      IF (MOD(KKT,MAP)) 150,140,150                                    ISODO710
  140 CALL PRINT(KKT,IPLACE,MEANS,STDEV,CLD,FLOINF,N)
      GO TO 161                                                                   *NEW
  150 IF (MOD(KKT,KRN))161,160,161                                               *NEW
  160 CONTINUE                                                         ISODO750**-2
      CALL PRINT(KKT,IPLACE,MEANS,STDEV,CLD,FLOINF,N)
  161 CONTINUE                                                                   *NEW
C                                                                                *NEW
C     FOR ITERATION N CHECK N(K) AGAINST PMIN * NOFEAT                           *NEW
C                                                                                *NEW
      IF (ISEW .NE. NOSEQ) GO TO 169                                             *NEW
      ISTOP = 0                                                                  *NEW
  162 DO 165  K=1,LNCAT                                                          *NEW
      IF ( N(K) - (PMIN*NOFEAT)) 167,165,165                                     *NEW
  165 CONTINUE                                                                   *NEW
      IF (DEL) CALL PSPLIT(MEAN,STDEV,N,CLD,C,IPLACE,AVP,AMN,MEANS)              *NEW
      IF (DEL) CALL CLDIST(CLD,STDEV,MEANS)                                      *NEW
      RETURN                                                                     *NEW
```

```
      N(KK)=N(KK)+1                                                    ISODO440
      IPLACE(I)=KK                                                     ISODO450
      DO 90 K=1,NOFEAT                                                 ISODO470
       AMN(K,KK)=AMN(K,KK)+C(K,I)
       AVP(K,KK)=AVP(K,KK)+C(K,I)*C(K,I)
   90 CONTINUE                                                         ISODO500
  100 CONTINUE                                                         ISODO510
      IF (IRD.EQ.0) GO TO 110                                          ISODO520
      CALL RWRITE(ADRES2,IPLACE,ICCT,IST,I)
      ADRES2=ADRES2+ICCT
  105 IF(ISTAT.EQ.1)GO TO 105
  110 IRC=IRC-1                                                        ISODO540
      IF (IRC.GT.0) GO TO 20                                           ISODO550
      CALL CLOCK(TIME)                                                 ISODO560
      IF(MOD(KKT,KRN).EQ.0)WRITE(6,120)KKT,TIME
  120 FORMAT(' CUMULATIVE TIME AFTER ASSIGNING DATA TO CLUSTERS FOR ITE
     *ATION',I4,' IS',F10.6)
      DO 130 K=1,LNCAT                                                 ISODO590
       RND=FLOAT(N(K))
      DO 130 J=1,NOFEAT                                                ISODO610
       AMN(J,K)=AMN(J,K)/RND
       MEANS(J,K)=AMN(J,K)
  130  STDEV(J,K)=SQRT(AVP(J,K)/RND-AMN(J,K)*AMN(J,K))
       IF(ERCOMP.NE.1)GO TO 135                                                    *NEW
       ESUM=0.0                                                                    *NEW
       DO 132 J=1,NOFEAT                                                           *NEW
       DO 132 K=1,LNCAT                                                            *NEW
       ESUM=ESUM+N(K)*(STDEV(J,K))**2/TOTPTS                                       *NEW
  132  CONTINUE                                                                    *NEW
       ESQT=SQRT(ESUM)                                                             *NEW
       WRITE(6,133)ESQT                                                            *NEW
  133  FORMAT(1X,////'  ERCOMP = ',F7.3/)                                          *NEW
C                                                                      ISODO640
C     CALCULATE DISTANCES BETWEEN CLUSTER CENTERS                      ISODO650
C                                                                      ISODO660
  135  CALL CLDIST(CLD,STDEV,MEANS)                                               *NEW
C                                                                      ISODO680***-1
C     IF STOP EQUALS ZERO RETURN AFTER INITIAL CLASSIFICATION
C
      IF(STOP.EQ.0)RETURN
      IF (MOD(KKT,MAP)) 150,140,150                                    ISODO710
  140 CALL PRINT(KKT,IPLACE,MEANS,STDEV,CLD,FLDINF,N)
      GO TO 161                                                                   *NEW
  150 IF (MOD(KKT,KRN))161,160,161                                               *NEW
  160 CONTINUE                                                         ISODO750***-2
      CALL PRINT(KKT,IPLACE,MEANS,STDEV,CLD,FLDINF,N)
  161 CONTINUE                                                                    *NEW
C                                                                                 *NEW
C     FOR ITERATION N CHECK N(K) AGAINST PMIN * NOFEAT                            *NEW
C                                                                                 *NEW
      IF (ISEW .NE. NOSEQ) GO TO 167                                              *NEW
      ISTOP = 0                                                                   *NEW
  162 DO 165  K=1,LNCAT                                                           *NEW
      IF ( N(K) - (PMIN*NOFEAT)) 167,165,165                                      *NEW
  165 CONTINUE                                                                    *NEW
      IF (DEL) CALL PSPLIT(MEAN,STDEV,N,CLD,C,IPLACE,AVP,AMN,MEANS)               *NEW
      IF (DEL) CALL CLDIST(CLD,STDEV,MEANS)                                       *NEW
      RETURN                                                                      *NEW
```

```
      167 WRITE(6,168)K,N(K),PMIN,NOFEAT                                 *NEW
      168 FORMAT(/' CLUSTER',I3,' REMOVED FOR HAVING ONLY',I6,' POINTS*'/ *NEW
     *' MIN. POINTS IS  (',I4,' + ',I2,')'/                              *NEW
          CALL DELETE(K)                                                 *NEW
          DEL = .TRUE.                                                   *NEW
          GO TO 164                                                      *NEW
      169 CONTINUE                                                       *NEW
      170 CONTINUE
C                                                                        ISOD0780
C     ON ITERATIONS 1 THRU N-1 CHECK N(K) AGAINST NMIN                   *NEW
C                                                                        ISOD0800**-1
          DO 180 K=1,INCAT                                               ISOD0810
          IF (N(K)-NMIN) 190,180,180                                     ISOD0820
      180 CONTINUE                                                       ISOD0830
          IF(DEL)CALL CLDIST(CLD,STDEV,MEANS)
          GO TO 220                                                      ISOD0850
      190 IF (MOD(KKT,KRN)) 200,,200                                     ISOD0860
          WRITE (6,210)K,N(K),NMIN                                       ISOD0870
      200 CALL DELETE(K)                                                 ISOD0880
          DEL=.TRUE.
          GO TO 170                                                      ISOD0890
      210 FORMAT('0 CLUSTER ',I2,' REMOVED FOR HAVING  ONLY ',I6,        ISOD0900
     2       ' ELEMENTS. MIN. NO. ELEMENTS IS ',I6)                      ISOD0910
      220 CONTINUE                                                       ISOD0920
C                                                                        ISOD0930
C                                                                        ISOD0960**-1
                                                                         ISOD0970
C     SPLIT ITERATION                                                    ISOD0980
C                                                                        ISOD0990
C                                                                        ISOD1000
           DO 225 I=1,INCAT                                              *NEW
      225  PTR(I)=I                                                      *NEW
           ISPLT=0                                                       ISOD1030
           DO 260 K=1,INCAT                                              ISOD1040
C                                                                        ISOD1050
C     FIND MAXIMUM STANDARD DEVIATION PER CLUSTER                        ISOD1060
C                                                                        ISOD1070
           SGMA(K)=STDEV(1,K)                                            ISOD1080
           ISGMA(K)=1                                                    ISOD1090
           DO 250 J=2,KDIM                                               ISOD1100
           IF (STDEV(J,K)-SGMA(K)) 250,240,240                           ISOD1110
      240  ISGMA(K)=J                                                    ISOD1120
           SGMA(K)=STDEV(J,K)                                            ISOD1130
      250  CONTINUE                                                      ISOD1140
           IF(SGMA(K).GE.STDMAX)ISPLT=ISPLT+1                            ISOD1150
      260  CONTINUE                                                      ISOD1160
           IF(2*INCAT.GT.MAXCLS)CALL DESCEN(SGMA,LNCAT,ISGMA,PTR)        *NEW
           TEST=FLOAT(ISPLT)/FLOAT(LNCAT)                                ISOD1170
           IF(TEST.LE.PERCEN)SPLFIN=1
           IF (KKT .ST. ISTOP) SPLFIN = 1                                *NEW
           IF (SPLFIN .EQ. 0) GO TO 270                                  *NEW
           IF (MOD(KKT,KRN) .EQ. 0) WRITE(6,503)                         *NEW
      503  FORMAT( /)                                                    *NEW
           IF(MOD(KKT,KRN).EQ.0)WRITE(6,502)                             *NEW**-9
      502  FORMAT(1X,'USER INPUT-SPLIT-COMBINE SEQUENCE OF ITERATIONS.')
           ISEQ=ISEQ+1
           IF (SEQUEN(ISEQ) .EQ. 'S') GO TO 270                          *NEW
           IF (SEQUEN(ISEQ) .EQ. 'C') GO TO 410                          *NEW
```

E4 63

```
      167 WRITE(6,168)K,N(K),PMIN,NOFEAT                                *NEW
      168 FORMAT(/' CLUSTER',I3,' REMOVED FOR HAVING ONLY',I6,' POINTS*'/
     *' MIN. POINTS IS  (',I4,' + ',I2,')')                             *NEW
          CALL DELETE(K)                                                *NEW
          DEL = .TRUE.                                                  *NEW
          GO TO 164                                                     *NEW
      169 CONTINUE                                                      *NEW
      170 CONTINUE
C                                                                       ISOD0780
C     ON ITERATIONS 1 THRU N-1 CHECK N(K) AGAINST NMIN                  *NEW
C                                                                       ISOD0800**-1
          DO 180 K=1,INCAT                                              ISOD0810
          IF (N(K)-NMIN) 190,180,180                                    ISOD0820
      180 CONTINUE                                                      ISOD0830
          IF(DEL)CALL CLDIST(CLD,STDEV,MEANS)
          GO TO 220                                                     ISOD0850
      190 IF (MOD(KKT,KRN)) 200,,200                                    ISOD0860
          WRITE (6,210)K,N(K),NMIN                                      ISOD0870
      200 CALL DELETE(K)                                                ISOD0880
          DEL=.TRUE.
          GO TO 170                                                     ISOD0890
      210 FORMAT('0 CLUSTER ',I2,' REMOVED FOR HAVING  ONLY ',I6,       ISOD0900
     2       ' ELEMENTS, MIN: NO: ELEMENTS IS ',I6)                     ISOD0910
      220 CONTINUE                                                      ISOD0920
C                                                                       ISOD0930
C                                                                       ISOD0960**-1
                                                                        ISOD0970
C     SPLIT ITERATION                                                   ISOD0980
C                                                                       ISOD0990
C                                                                       ISOD1000
           DO 225 I=1,INCAT                                             *NEW
      225   PTR(I)=I                                                    *NEW
          ISPLT=0                                                       ISOD1030
          DO 260 K=1,INCAT                                              ISOD1040
C                                                                       ISOD1050
C     FIND MAXIMUM STANDARD DEVIATION PER CLUSTER                       ISOD1060
C                                                                       ISOD1070
          SGMA(K)=STDEV(1,K)                                            ISOD1080
          ISGMA(K)=1                                                    ISOD1090
          DO 250 J=2,KDIM                                               ISOD1100
          IF (STDEV(J,K)-SGMA(K)) 250,240,240                           ISOD1110
      240 ISGMA(K)=J                                                    ISOD1120
          SGMA(K)=STDEV(J,K)                                            ISOD1130
      250 CONTINUE                                                      ISOD1140
          IF(SGMA(K).GE.STDMAX)ISPLT=ISPLT+1                            ISOD1150
      260 CONTINUE                                                      ISOD1160
           IF(2*INCAT.GT.MAXCLS)CALL DESCEN(SGMA,LNCAT,ISGMA,PTR)       *NEW
          TEST=FLOAT(ISPLT)/FLOAT(LNCAT)                                ISOD1170
           IF(TEST.LE.PERCEN)SPLFIX=1                                   *NEW
           IF (KKT .EQ. ISTOP) SPLFIX = 1                               *NEW
           IF (SPLFIX .EQ. 0) GO TO 270                                 *NEW
           IF (MOD(KKT,KRN) .EQ. 0) WRITE(6,503)                        *NEW
      503 FORMAT( /)                                                    *NEW
           IF(MOD(KKT,KRN).EQ.0)WRITE(6,502)                            **-9
      502  FORMAT(1X,'USER INPUT-SPLIT-COMBINE SEQUENCE OF ITERATIONS:')
           ISEQ=ISEQ+1
          IF (SEQUEN(ISEQ) .EQ. 'S') GO TO 270                          *NEW
          IF (SEQUEN(ISEQ) .EQ. 'C') GO TO 410                          *NEW
```

E-4 63

```
      C
      C     IS SPLITTING REQUIRED                                      ISOD1200
      C                                                                ISOD1210
      C                                                                ISOD1220
  270 K=1                                                              ISOD1230
      NCAT=INCAT                                                       ISOD1240
  280 IF (K-NCAT) 290,290,500                                          ISOD1250
  290 IF (STDMAX-SGMA(K)) 300,300,310                                  ISOD1260
  300 IF (N(K)-(NMIN+NMIN+2)) 310,310,320                              ISOD1270
  310 K=K+1                                                            ISOD1280
      GO TO 280                                                        ISOD1290
      C                                                                ISOD1300
      C     SPLIT CLUSTER K                                            ISOD1310
      C                                                                ISOD1320
  320 TRIG1=1                                                          ISOD1330
      DEL=.TRUE.                                                       *NEW
      KK=ISGMA(K)                                                      ISOD1340
  330 INCAT=INCAT+1                                                    ISOD1350
      IF (INCAT.LE.MAXCLS) GO TO 350                                   ISOD1360
      IF(MOD(KKT,KRN).EQ.0)WRITE(6,340)KKT                             *NEW
  340 FORMAT(/' MAXIMUM CLUSTERS ON ITERATION',I9/' SPLITTING REQUIRED H*--2
     *UT NOT PERFORMED'/)                                              ISOD1390
      LNCAT=MAXCLS                                                     ISOD1400
      GO TO 500                                                        ISOD1410
  350 INC=INCAT                                                        ISOD1420
      LL=PTR(K)                                                        *NEW
  360 DO 370 I=1,KDIM                                                  ISOD1440
  370   AMN(I,INC)=AMN(I,LL)                                           *NEW
  380   AMN(KK,LL)=AMN(KK,LL)+SEP*SGMA(K)                              *NEW
        AMN(KK,INC)=AMN(KK,INC)-SEP*SGMA(K)                            *--2
      SGMA(K)=0.0                                                      ISOD1500
      IF (MOD(KKT,KRN)) 400,,400                                       ISOD1510
      WRITE(6,390)LL,KK,INC                                            *NEW
  390 FORMAT('0 CLUSTER ',I2,' IS SPLIT IN THE ',I2,'TH PARAMETER INTO C*--1
     2LUSTER ',I2)                                                     ISOD1540
  400 CONTINUE                                                         ISOD1550
      K=K+1                                                            ISOD1560
      GO TO 280                                                        ISOD1570
      C                                                                ISOD1580
      C     EVEN ITERATION                                             ISOD1590
      C                                                                ISOD1600
      C     ARE CLUSTERS TO BE COMBINED                                ISOD1610
      C                                                                ISOD1620
  410 CONTINUE                                                         ISOD1630
      DO 405 L=1,LNCAT                                                 *NEW
  405   PTR(L)=1                                                       *NEW
      C                                                                *NEW
      NOCOMB=0                                                         *NEW
      NUCLST=LNCAT-1                                                   *NEW
      DO 440 L=1,NUCLST,2                                              *NEW
      NOCLTR = LNCAT - 1                                               *NEW
      KK=0                                                             *NEW
      DMIN=DLMIN                                                       *NEW
      DO 430 I=1,NOCLTR                                                *NEW
      C                                                                *NEW
      IF(PTR(I).EQ.0)GO TO 430                                         *NEW
      II=I+1                                                           *NEW
      DO 425 J=II,LNCAT                                                *NEW
      IF(PTR(J).EQ.0)GO TO 425                                         *NEW
```

```
      C
      C     IS SPLITTING REQUIRED                                          ISOD1200
      C                                                                    ISOD1210
      C                                                                    ISOD1220
  270 K=1                                                                  ISOD1230
      NCAT=INCAT                                                           ISOD1240
  280 IF (K-NCAT) 290,290,500                                              ISOD1250
  290 IF (STDMAX-SGMA(K)) 300,300,310                                      ISOD1260
  300 IF (N(K)-(NMIN+NMIN+2)) 310,310,320                                  ISOD1270
  310 K=K+1                                                                ISOD1280
      GO TO 280                                                            ISOD1290
      C                                                                    ISOD1300
      C     SPLIT CLUSTER K                                                ISOD1310
      C                                                                    ISOD1320
  320 TRIG1=1                                                              ISOD1330
       DEL=.TRUE.
      KK=ISGMA(K)                                                          ISOD1340
  330 INCAT=INCAT+1                                                        ISOD1350
      IF (INCAT.LE.MAXCLS) GO TO 350                                       ISOD1360
      IF(MOD(KKT,KRN).EQ.0)WRITE(6,340)KKT
  340 FORMAT(/' MAXIMUM CLUSTERS OR ITERATION',I4/' SPLITTING REQUIRED B  ISOD1380
     *UT NOT PERFORMED'/)                                                 ISOD1390
      LNCAT=MAXCLS                                                         ISOD1400
      GO TO 500                                                            ISOD1410
  350 INC=INCAT                                                            ISOD1420
       LL=PTR(K)                                                                *NEW
  360 DO 370 I=1,KDIM                                                      ISOD1440
  370   AMN(I,INC)=AMN(I,LL)                                                   *NEW
  380   AMN(KK,LL)=AMN(KK,LL)+SEP*SGMA(K)                                      *NEW
       AMN(KK,INC)=AMN(KK,INC)-SEP*SGMA(K)                                     *E-2
      SGMA(K)=0.0                                                          ISOD1500
      IF (MOD(KKT,KRN)) 400,,400                                           ISOD1510
       WRITE(6,390)LL,KK,INC                                                   *NEW
  390 FORMAT('0 CLUSTER ',I2,' IS SPLIT IN THE ',I2,'IN PARAMETER INTO C   *E-1
     2LUSTER ',I2)                                                         ISOD1540
  400 CONTINUE                                                             ISOD1550
      K=K+1                                                                ISOD1560
      GO TO 280                                                            ISOD1570
      C                                                                    ISOD1580
      C     EVEN ITERATION                                                 ISOD1590
      C                                                                    ISOD1600
      C     ARE CLUSTERS TO BE COMBINED                                    ISOD1610
      C                                                                    ISOD1620
  410 CONTINUE                                                             ISOD1630
       DO 405 L=1,LNCAT                                                        *NEW
  405  PTR(L)=1                                                                *NEW
      C                                                                        *NEW
       NOCOMB=0                                                                *NEW
       NOCLST=LNCAT-1                                                          *NEW
       DO 440 L=1,NOCLST,2                                                     *NEW
      NOCLTR = LNCAT - 1                                                       *NEW
       KK=0                                                                    *NEW
       DMIN=DLMIN                                                              *NEW
      DO 430 I=1,NOCLTR                                                        *NEW
      C                                                                        *NEW
       IF(PTR(I).EQ.0)GO TO 430                                                *NEW
       II=I+1                                                                  *NEW
       DO 425 J=II,LNCAT                                                       *NEW
       IF(PTR(J).EQ.0)GO TO 425                                                *NEW
```

```
235•          SDIJ = 0.0                                                                    •NEW
236•           DO 420 JJ=1,KDIM                                                             •NEW
237•          SDIJ=SDIJ+((AMN(JJ,I)-AMN(JJ,J))**2/(STDEV(JJ,I)*STDEV(JJ,J)))               •NEW
238•      420 CONTINUE                                                                      •NEW
239•          DIJ=SQRT(SDIJ)                                                               •NEW
240•      C                                                                                •NEW
241•          IF(DIJ.GT.DMIN)GO TO 425                                                     •NEW
242•          DMIN=DIJ                                                                     •NEW
243•          KK=I                                                                         •NEW
244•          KKK=J                                                                        •NEW
245•      425 CONTINUE                                                                     •NEW
246•      430 CONTINUE                                                                     •NEW
247•      C                                                                                •NEW
248•          IF(KK.EQ.0)GO TO 480                                                         •NEW
249•          PTR(KK)=0                                                                    •NEW
250•      C                                                                                •NEW
251•      C     COMBINE CLUSTERS KK AND KKK                                                •NEW
252•      C                                                                                •NEW
253•          DEL=.TRUE.                                                                   •NEW
254•          RND=1.0 /FLOAT(N(KK)+N(KKK))                                                 •NEW
255•      C                                                                                •NEW
256•          DO 460 K=1,KDIM                                                              •NEW
257•      460 AMN(K,KK)=(N(KK)*AMN(K,KK)+N(KKK)*AMN(K,KKK))*RND                            •NEW
258•      C                                                                                •NEW
259•          CALL DELETE(KKK)                                                             •NEW
260•          IF (KKK .EQ. (LNCAT+1)) GO TO 435                                            •NEW
261•      C                                                                                •NEW
262•      C     MOVE POINTERS UP                                                           •NEW
263•      C                                                                                •NEW
264•          DO 175 K=KKK,LNCAT                                                           •NEW
265•      175 PTR(K) = PTR(K+1)                                                            •NEW
266•      C                                                                                •NEW
267•      435 IF(MOD(KKT,KRN))440•,440                                                     •NEW
268•          WRITE(6,490)KK,KKK,KK                                                        •NEW
269•      440 CONTINUE                                                                     •NEW
270•      C                                                                                •NEW
271•      480 CONTINUE                                                                     •NEW
272•      490 FORMAT(' CLUSTERS ',I2,' AND ',I2,' HAVE BEEN COMBINED INTO CLUST ISOD1980•-31
273•         2ER ',I2)                                                          ISOD1990
274•      C•                                                                    ISOD2000
275•      C•    REINITIALIZE                                                    ISOD2010
276•      C•                                                                    ISOD2020
277•      500 CONTINUE                                                          ISOD2030
278•          DO 510 J=1,MAXCLS                                                 ISOD2040
279•          N(J)=1                                                            ISOD2050
280•          SGMA(J)=0.0                                                       ISOD2070
281•          ISGMA(J)=0                                                        ISOD2080
282•          DO 510 K=1,KDIM                                                   ISOD2090
283•          AVP(K,J)=0.0                                                      ISOD2100
284•          STDEV(K,J)=0.0                                                    ISOD2110
285•          MEANS(K,J)=AMN(K,J)                                               ISOD2120
286•          AMN(K,J)=0.0                                                      ISOD2130
287•      510 CONTINUE                                                          ISOD2140
288•          KKT=KKT+1                                                         ISOD2160
289•          DEL=.FALSE.                                                       ISOD2170
290•          GO TO 10                                                          ISOD2180
291•      C                                                                     ISOD2230•.-8
292•      530 IF (KKT.NE.2) GO TO 530
```

```
235*          SDIJ = 0.0                                                                    *NEW
236*          DO 420 JJ=1,KDIM                                                              *NEW
237*          SDIJ=SDIJ+((AMN(JJ,I)-AMN(JJ,J))**2/(STDEV(JJ,I)*STDEV(JJ,J)))               *NEW
238*    420   CONTINUE                                                                      *NEW
239*          DIJ=SQRT(SDIJ)                                                                *NEW
240*    C                                                                                   *NEW
241*          IF(DIJ.GT.DMIN)GO TO 425                                                      *NEW
242*          DMIN=DIJ                                                                      *NEW
243*          KK=I                                                                          *NEW
244*          KKK=J                                                                         *NEW
245*    425   CONTINUE                                                                      *NEW
246*    430   CONTINUE                                                                      *NEW
247*    C                                                                                   *NEW
248*          IF(KK.EQ.0)GO TO 480                                                          *NEW
249*          PTR(KK)=0                                                                     *NEW
250*    C                                                                                   *NEW
251*    C       COMBINE CLUSTERS KK AND KKK                                                 *NEW
252*    C                                                                                   *NEW
253*          DEL=.TRUE.                                                                    *NEW
254*          RND=1.0 /FLOAT(N(KK)+N(KKK))                                                  *NEW
255*    C                                                                                   *NEW
256*          DO 460 K=1,KDIM                                                               *NEW
257*    460   AMN(K,KK)=(N(KK)*AMN(K,KK)+N(KKK)*AMN(K,KKK))*RND                            *NEW
258*    C                                                                                   *NEW
259*          CALL DELETE(KKK)                                                              *NEW
260*          IF (KKK .EQ. (LNCAT+1)) GO TO 535                                             *NEW
261*    C                                                                                   *NEW
262*    C     MOVE POINTERS UP                                                              *NEW
263*    C                                                                                   *NEW
264*          DO 175 K=KKK,LNCAT                                                            *NEW
265*      175 PTR(K) = PTR(K+1)                                                             *NEW
266*    C                                                                                   *NEW
267*      435   IF(MOD(KKT,KRN))440,,440                                                   *NEW
268*          WRITE(6,490)KK,KKK,KK                                                         *NEW
269*      440   CONTINUE                                                                    *NEW
270*    C                                                                                   *NEW
271*      480   CONTINUE                                                                    *NEW
272*      490 FORMAT(' CLUSTERS ',I2,' AND ',I2,' HAVE BEEN COMBINED INTO CLUST ISOD1980    *-31
273*         2ER ',I2)                                                          ISOD1990
274*    C                                                                       ISOD2000
275*    C     REINITIALIZE                                                      ISOD2010
276*    C                                                                       ISOD2020
277*      500 CONTINUE                                                          ISOD2030
278*          DO 510 J=1,MAXCLS                                                 ISOD2040
279*          N(J)=1                                                            ISOD2050
280*          SGMA(J)=0.0                                                       ISOD2070
281*          ISGMA(J)=0                                                        ISOD2080
282*          DO 510 K=1,KDIM                                                   ISOD2090
283*          AVP(K,J)=0.0                                                      ISOD2100
284*          STDEV(K,J)=0.0                                                    ISOD2110
285*          MEANS(K,J)=AMN(K,J)                                               ISOD2120
286*          AMN(K,J)=0.0                                                      ISOD2130
287*      510 CONTINUE                                                          ISOD2140
288*          KKT=KKT+1                                                         ISOD2160
289*          DEL=.FALSE.                                                       ISOD2170
290*          GO TO 10                                                          ISOD2180
291*    C                                                                       ISOD2230
292*      530 IF (KKT.NE.2) GO TO 550                                           *-8
```

E-6 65