CIRCULAR 111

Computerization of the New Mexico Bureau of Mines Mineralogical Museum

by JACQUES RENAULT

RENA MAE BONEM

RONALD RIESE

STATE BUREAU OF MINES AND MINERAL RESOURCES
NEW MEXICO INSTITUTE OF MINING AND TECHNOLOGY
CAMPUS STATION SOCORRO, NEW MEXICO

THE NEW MEXICO BUREAU OF MINES AND MINERAL RESOURCES

Don H. Baker, Jr., *Director*

Full-Time Staff

Joyce M. Aguilar, *Stenographer*
William E. Arnold, *Scientific Illustrator*
Roshan B. Bhappu, *Senior Metallurgist* (on lv.)
Robert A. Bieberman, *Petroleum Geologist*
Lynn A. Brandvold, *Chemist*
Elise Brower, *Assistant Chemist* (on lv.)
Charles E. Chapin, *Geologist*
Richard R. Chavez, *Lab. Assistant*
Lois M. Devlin, *Office Manager*
Jo Drake, *Director's Secretary*
Rousseau H. Flower, *Senior Paleontologist*
Roy W. Foster, *Petroleum Geologist*
William L. Hawks, *Materials Engineer*
Frank E. Kottlowski, *Sr. Geol. & Ass't. Dir.*
Alex. Nicholson, *Geologist-Editor*
Robert L. Price, *Draftsman*
Jacques R. Renault, *Geologist*
John W. Shomaker, *Geologist*
Jackie H. Smith, *Lab. Assistant*
Marilynn Szydlowski, *Secretary*
Robert H. Weber, *Senior Geologist*
Sue Wilks, *Typist*
Max E. Willard, *Economic Geologist*
Juarine W. Wooldridge, *Editorial Clerk*

Part-Time Staff

Martha K. Arnold, *Editorial Assistant*
Gus Blaisdell, *Public Relations*
James A. Brierley, *Ass't Prof. Biology*
Robin C. Lease, *Geologist*
Rufie Montoya, *Dup. Mach. Oper.*
John Reiche, *Instrument Manager*
Ronald Roman, *Research Metallurgist*
W. Kelly Summers, *Geologist*
Frank B. Titus, *Geologist*

Graduate Students

Elise Brower, *Geochemist*
Saul Escalera, *Metallurgist*
Walter W. Fisher, *Metallurgist*
Marsha Koehn, *Geologist*
Che-Chen Liu, *Metallurgist*
Walter H. Pierce, *Geologist*
Haia Roffman, *Geochemist*
David A. Schwab, *Geochemist*

Plus more than 35 undergraduate assistants

CIRCULAR 111

Computerization of the New Mexico Bureau of Mines Mineralogical Museum

by JACQUES RENAULT

Geologist, New Mexico State Bureau of Mines and Mineral Resources

RENA MAE BONEM

Student Assistant, New Mexico Institute of Mining and Technology

RONALD RIESE

Graduate Assistant, New Mexico Institute of Mining and Technology

1970

STATE BUREAU OF MINES AND MINERAL RESOURCES
NEW MEXICO INSTITUTE OF MINING AND TECHNOLOGY
CAMPUS STATION SOCORRO, NEW MEXICO

For sale by the New Mexico Bureau of Mines & Mineral Resources
Campus Station, Socorro, N. Mex. 87801—Price $1. 00

CONTENTS

Page

INTRODUCTION

In a broad sense, a mineralogical museum is a kind of library that serves both as a medium for display of interesting objects and as a repository for reference material. It is in this latter role that it is most useful in scientific investigations.

A major problem in the use of a museum as a reference tool is the retrieval of specimens that possess certain combinations of characteristics. Such retrieval requires complex cross-indexing or sorting and can be time-consuming. For these reasons, a series of computer programs called the NMBMMR Mineralogical Museum System has been developed. This report describes the system and its use. To our knowledge the only other mineralogical collection which has been computerized to date is that of the Smithsonian Institution (Creighton and King, 1969).

The authors wish to acknowledge the help of the New Mexico Institute of Mining and Technology Computer Center staff, especially Tom Nartker and Patricia Myers, for assistance in the construction of the system.

HISTORY OF THE MUSEUM

The mineralogical museum was begun in the early 1900's when C. T. Brown, a member of the Board of Regents, gave 2,565 specimens to the New Mexico Institute of Mining and Technology (then called New Mexico School of Mines). Unfortunately, 50 percent of the Brown Collection was destroyed by a fire in 1928. Until 1964 the collection was maintained by the Geology Department of the School. When the New Mexico State Bureau of Mines and Mineral Resources wing was added onto the Research and Development building in 1962, provision was made for a mineral museum, and a considerable portion of the collection was transferred to the Bureau.

In 1965, a systematic inventory of the entire collection was begun by Edward Bingler, mineralogist for the Bureau. Bingler left in 1967, and this task was taken over by Jacques Renault and student assistants Lorna Goebel and Rena Bonem and graduate assistant Ronald Riese. At that time the remainder of the collection was transferred to the Bureau and divided into four subcollections:

1. A display collection, consisting of spectacular specimens

2. The Dana Collection, consisting of individual minerals classified according to the Dana System
3. A working collection, consisting of specimens that could be destroyed in the course of analytical investigations or traded for new material
4. A backup collection of duplicate material

Although 8,800 specimens had been cataloged, only 6,900 specimens have been recovered and inventoried for the present system; the remainder is presumably lost.

Previously, access to the collection had been by a filing system in which specimens were stored or displayed in various places by species, and specimen data were recorded on cards filed by catalog number in order of acquisition. The information recorded for each specimen consisted of minerals present, locality, donor or source, and comments (such as composition, diffraction pattern, etc.). To examine a specimen that had a desired combination of characteristics, such as mineral assemblage and mining district, it was necessary to search through all the storage locations of all the minerals in the assemblage. In 1969, we began to computerize the on-file mineral data and have now essentially completed the task.

The Bureau is currently preparing X-ray-diffraction patterns and emission spectrograms of material in the museum collection. These new data are being added to computer storage as they are acquired. At present, catalog number, mineral assemblage, donor, geographic location, storage or display location, and comments are recorded on two IBM cards and stored on disc. Additional information, such as correspondence, bibliographic data, and analytical data, is stored on tape.

COMPUTERIZATION

The mineral collection was computerized to facilitate cross-referencing and recovery of specimens with desired characteristics. The specific goal was to be able to search in six categories: mineral assemblage, geographical location, storage or display location, donor or source, comments, and extra data; this goal has been achieved.

The mineralogical museum programs were written for the IBM 360/44 computer at the New Mexico Institute of Mining and Technology Computer Center. This computer has a 32 K core storage. Specimen data are punched on IBM cards and ultimately copied onto disc. The system is writ-

ten in FORTRAN IV; it can accommodate 160 bytes of information per specimen on a two-card format and unlimited information in an "extra-data" format. The system is designed to print six basic kinds of lists that can be displayed in alphabetical order of minerals, numerical order of catalog number, and with or without all of the stored data for each specimen. The basic lists are:

1. All catalog numbers in the collection
2. All mineral species in the collection
3. All geographic locations represented in the collection
4. All donors or sources represented in the collection
5. All storage or display locations
6. Any combination of mineral assemblage, locality, storage, donor, comment entry, and extra data

Structure of the System

Three types of programs' constitute the system: (1) those used to store records of all specimen data on tape and disc (MUSLDR, MUZMGR, and UNBLK), (2) those used to provide and maintain catalogs (RDR, MNLIST, CORRCT, and UPDATE), and (3) those used for searching (MNRL and other single-criterion searches and MULSER). A description of each program is given in appendix A and flow charts are included in appendix C; appendix D contains program listings.

Once the cards for each specimen have been punched and the information stored sequentially on disc, RDR and MNLIST are used to write lists of all stored data in the two-card format by catalog number and by all mineral species in alphabetical order. These lists are used principally for museum bookkeeping. Individual specimen records are corrected or augmented by the program CORRCT. Deletions of old records and additions of new records are accomplished by the program UPDATE. Additional information is stored on tape in the extra-data format and is called for by MULSER.

The single-criterion search programs (MNRL, DONOR, TOWN, CONTRA, etc.) and the multisearch program (MULSER) are used to extract mineral data from disc storage. The program used depends on the type of information desired and whether the search is to be done on the basis of a single criterion (e. g. , mining district) or a combination of criteria (e. g. , galena + fluorite + Hansonburg district).

REFERENCES

Creighton, Reginald, and King, Richard, 1969, The Smithsonian Institution Information Retrieval (SIIR) System for biological and petrological data: Smithsonian Inst. , Inf. Systems Innovations, v. I, n. 1, 25 p.

Kottlowski, F. E. , 1968, Standard stratigraphic computer coding for geologists: Jour. Geol. Education, v. 16, p. 143-145.

APPENDIX A

Descriptions of Programs

MUSLDR

This program and MUZMGR arrange the specimen records according to catalog number. It was necessary to do this in two steps because core sorting in the IBM 360/44 at New Mexico Tech is restricted to 160-byte records of up to 500 specimens.

The first step, MUSLDR, is used to make a tape on which specimen records are arranged by catalog number in blocks of 500 entries. The entries in each successive block are simply whatever groups of 500 records are successively loaded into the computer. After each block is sorted in core, it is written on tape; this process continues until a trailer card with a zero catalog number has been read. See Figure 1 for a flow chart of MUSLDR.

MUZMGR

The second step, MUZMGR, is used to convert the tape made by MUSLDR into a tape in which all records are in order of catalog number. This is accomplished by merging the blocks made by MUSLDR. Three tapes, including tape (1) made by MUSLDR, are required.

Block (1) is copied onto tape (2) and the number of remaining specimens on tape (1) is calculated to determine the number of repetitions necessary to complete the merging of all the blocks. Block (2) is then read into core and each record is compared to the records on tape (2). As the comparison proceeds, the records of block (1) and block (2) are merged in proper sequence onto tape (3). Block (3) is then copied from tape (1) into core and then compared with the records on tape (3). As comparison proceeds, the records of blocks (1), (2), and (3) are merged onto tape (2) in correct sequence. Block (4) is read into core, merged with blocks (1), (2), (3) and copied onto tape (3). This process of reading into core and alternating tapes continues until all of the records on tape (1) are exhausted and merged either onto tape (2) or (3). A message appears on the printer to inform the operator which tape contains the complete sequential file . The MUZMGR flow chart is shown in Figures 2 and 3.

UNBLK

The UNBLK program is used to copy the information on tape onto disc. Records of two specimens consisting of a total of 320 bytes are converted into a single disc record of 360 bytes; the additional 40 bytes are terminal blanks to make the record length compatible with FORTRAN.

Currently, writing two specimen records per disc record is the most economical way of storing information. UNBLK reads the tape records of two specimens, then writes them as a single record on disc. This process continues until all tape records have been read. Figure 4 shows the flow chart for UNBLK.

RDR

This program provides a listing and optional tape copy of the catalog information recorded on the MUSLDR tape.

First a card is read to determine whether a tape copy is desired. If that card has a "0" in the first column, no tape will be made. One sample entry consisting of 160 bytes is read; then if a tape is desired, an unformated record is written on the new tape. Finally, the records are printed. This sequence continues until all records have been read. Figure 5 shows the RDR flow chart.

MNLIST

The MNLIST program is used to produce an alphabetical listing of all mineral species stored on disc.

The procedure followed is to use a blank as the first entry in the list to be made. Then the five mineral entries in each specimen are compared to all entries on the list, and if a mineral is found to be different from those already included on the list it is added. This process continues until all records on the disc have been searched and CORSRT, an on-line alphabetizing subroutine, is called. The alphabetical list is then written on the printer. See Figure 6 for a flow chart.

CORRCT

The program is used for all corrections that do not require addition or deletion of records on the catalog disc; corrections are accomplished by rewriting the appropriate record.

The corrected information is placed in numerical order of catalog numbers. First, one corrected record is read as two cards, then the disc catalog numbers are compared to the correction catalog number until the

disc catalog number is no longer less than the number read. If the two are equivalent, then the record is rewritten with the new information. However, if the disc catalog number is greater than the correction catalog number, then an error in sequencing has occurred and an error message appears on the console typewriter. This condition also occurs in termination as the sequence of reading and corrections continues until the trailer card of "0000" is read. A flow chart is shown in Figure 7.

UPDATE

This program is used for additions of new records and deletions of old records on the museum catalog disc. Simple corrections which can be achieved by merely rewriting the record are accomplished by using the program CORRCT. The UPDATE program must be performed by making a tape and rewriting the disc catalog. UPDATE cards for a deletion contain only the specimen number, and cards for additions are identical to cards used in compiling the catalog.

The procedure followed is to first read the UPDATE cards in order of sample number. Then the original tape is copied onto a new tape until the catalog number of the update is less then or equal to the current tape record. At that point, the old record is omitted or the new record is inserted in numerical sequence on the new tape. This process continues until a trailer card "-1111" is encountered. Figure 8 shows the UPDATE flow chart.

Single-Criterion Searches

MNRL, DONOR, TOWN, etc, are all basically similar and are used as searches for a particular item within one of the criteria used in describing the specimens. These programs were used in building the system and are not used in routine applications because single-criterion searches can be accomplished by MULSER.

To use these programs, the name of the criterion desired is read in, then all of the records on the disc are searched for the information. If a match occurs, the catalog number of that specimen is added to a list which is printed at the completion of the search. Up to 50 items may be searched for at one time and the last search is indicated by the "0000" trailer card.

An example is the search for White Oaks mining district, New Mexico. The MINDST program would be used and WHITE OAKS would appear on a card and be followed by a "0000" card if this is the only search. If other searches are desired, they would follow WHITE OAKS before the "0000" trailer. Then records are read on the disc and the mining district name is compared with WHITE OAKS. The catalog numbers of any matches are stored in a list until the entire disc has been searched. Then all catalog

numbers where matches occurred are printed under a title "MINING DISTRICT SEARCH FOR WHITE OAKS." Figure 9 gives a flow chart for the special case of mineral searches where five names have to be compared (MNRL) and other single-criterion searches are flow charted in Figure 10.

MULSER

This program is designed to search, for a combination of 1 to 15 of the following criteria:

(a) 1 to 5 mineral names
(b) Catalog number
(c) Donor
(d) Geographic location
    (1) Country
    (2) State
    (3) County
    (4) Town
    (5) Mining district

Storage or display location

Comments

Extra data
    (1) Locality data
    (2) Analytical data
    (3) Bibliographic data
    (4) Correspondence
    (5) Miscellaneous

The data deck consists of a card indicating the number of records on the disc, followed by a code card with the number of criteria upon which the search is to be based and ones and zeros in 14 positions, one position for each criterion except extra data, indicating which criteria are to be searched for and which are not to be searched for. The 14 positions on this card are in the order in which the criteria occur in storage. Following this card are cards with the names of the items to be searched for, in order. A card with letters signifying the type of extra data (if any) desired is next and is followed by a card with the number of extra-data entries on the extra-data tape.

The procedure is to search for the first name within the criteria on each record. If the name matches the corresponding entry for the record, the next name (criterion) is compared; if it matches also, the next name is compared. When a match does not occur, the next record on the disc is examined for the first criterion. If a specimen matches all criteria, its cata-

log number is added to a list, which will be printed when all records have been searched.

If extra data exist for a particular specimen beyond that that can be stored on the museum catalog disc, it may be stored on the extra-data tape. In this case, an "X" occurs in the last space of the second specimen-card format, and the nature of the extra data is indicated by L for locality data, A for analytical data, B for bibliographic data, C for correspondence, and M for miscellaneous data in the last five spaces of the comment entry on the second-card format.

If particular kinds of extra data are search criteria, the pertinent entries on the extra-data tape are printed out. The MULSER flow chart is shown in Figure 11.

APPENDIX B

Formats of Data Cards

Primary Data Cards

Two cards are required for each museum specimen. The first card consists of:

1. A 5-digit field for acquisition number
2. Five 12-character fields to accommodate five mineral name s
3. A 15-character field to accommodate the donor name or source

The second card consists of:

1. An 8-character field for country of origin
2. An 8-character field for state of origin
3. An 8-character field for county of origin
4. An 8-character field for town of origin
5. A 16-character field for mining district of origin
6. A 21-character field for comments
7. A 5-character field for extra-data designations
8. A 3-character field for specimen location in the museum
9. A 1-character field which contains an X if extra data for the. specimen exists in museum files

Lack of information is indicated by blanks in the appropriate field on either card.

Retrieval Data Cards

The data deck for retrieval of information from the stored specimen data consists of two control cards and a card for each criterion to be search-ed.

The first card contains the number of specimens to be included in the search, usually the number of museum specimens that have been stored in the system.

The second card consists of:

1. A 2-digit field indicating the number of search criteria
2. A 1-digit field for catalog search

3. A 5-digit field for mineral search
4. A 1-digit field for donor search
5. A 5-digit field for geographic location search
6. A 1-digit field for comment search
7. A 1 -digit field for museum storage-location search
8. A 1-digit field for extra-data search

The cards that follow the second card name each criterion in the following order:

1. One card far catalog-number search
2. Up to five mineral cards
3. One donor card
4. Up to five geographic-location cards in the following order:
    a. Country
    b. State or province
    c. County or equivalent
    d. Town
    e. Mining district
5. One card for comment words to be searched
6. One card for museum storage location to be searched
7. One card for extra-data search, with designations in columns 1 through 5 in the following order:
    a. Locality, designated by "L"
    b. Analytical data, designated by "A"
    c. Bibliographic data, designated by "B"
    d. Correspondence, designated by "C"
    e. Miscellaneous data, designated by "M"

On the second control card, a search for a particular criterion is indicated by "ones" in the appropriate fields. Blanks in the appropriate fields indicate that no search is to be made. If a search is to be made for mineral names, the number of ones in field (3) indicate the number of different minerals in the search.

Names appearing on the criteria search cards are abbreviated so as to be consistent with the way they appear on the primary data cards; that is, according to the method of Kottlowski (1968) and no longer than the corresponding fields on the primary data cards.

APPENDIX C

Flow Charts for Programs

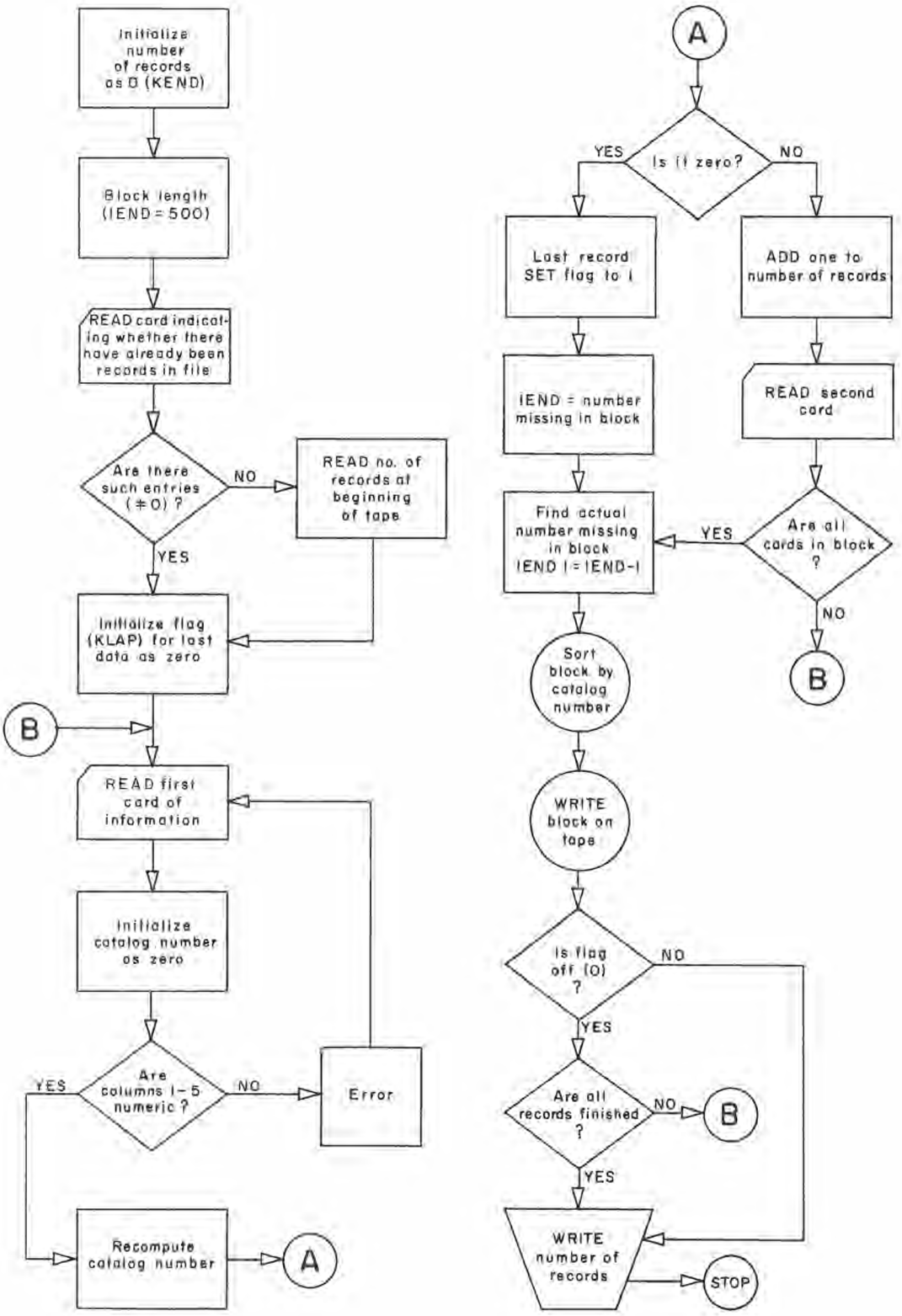

Figure 1. Flow Chart of MUSLDR.

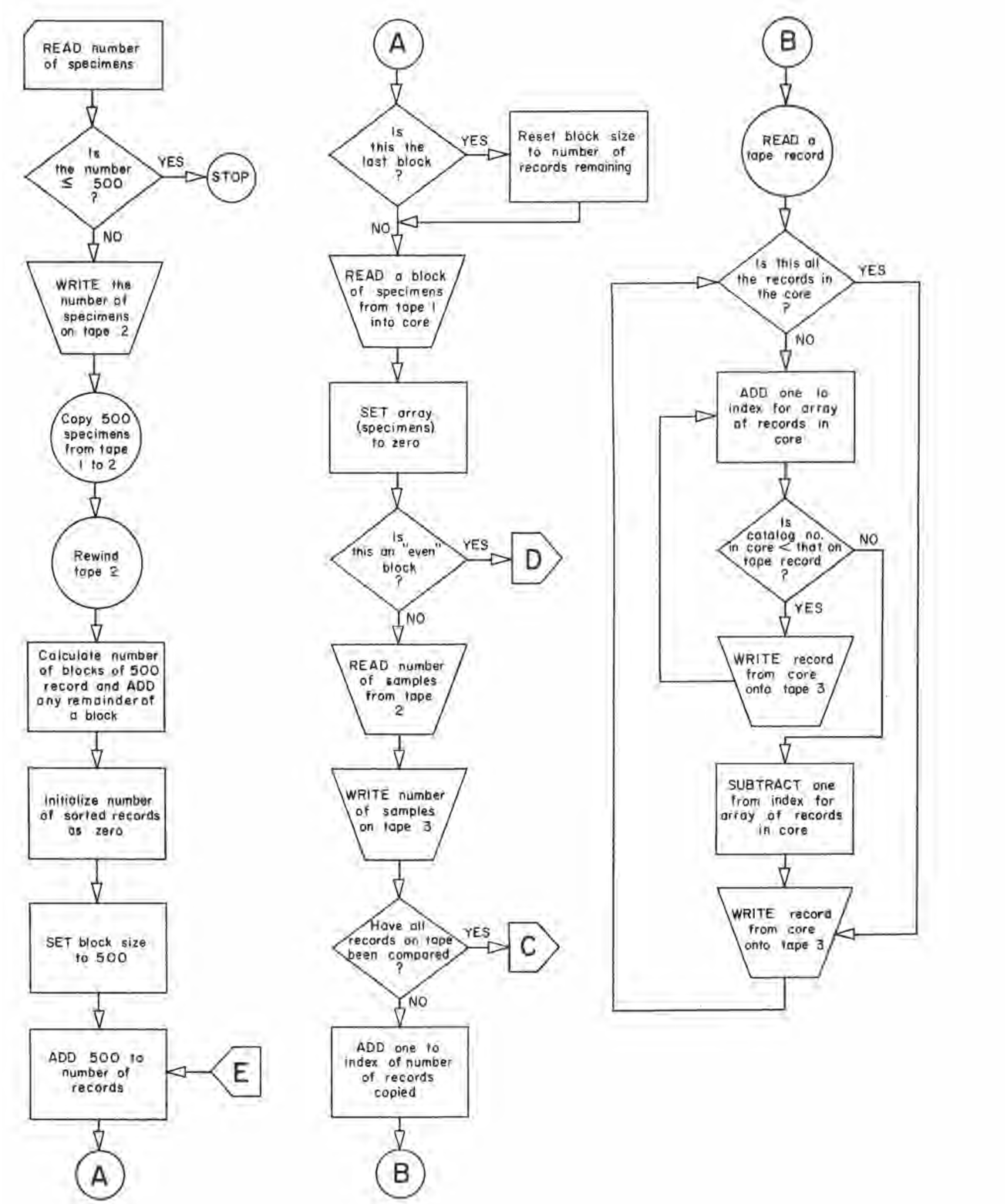

Figure 2. Flow Chart of MUZMGR.

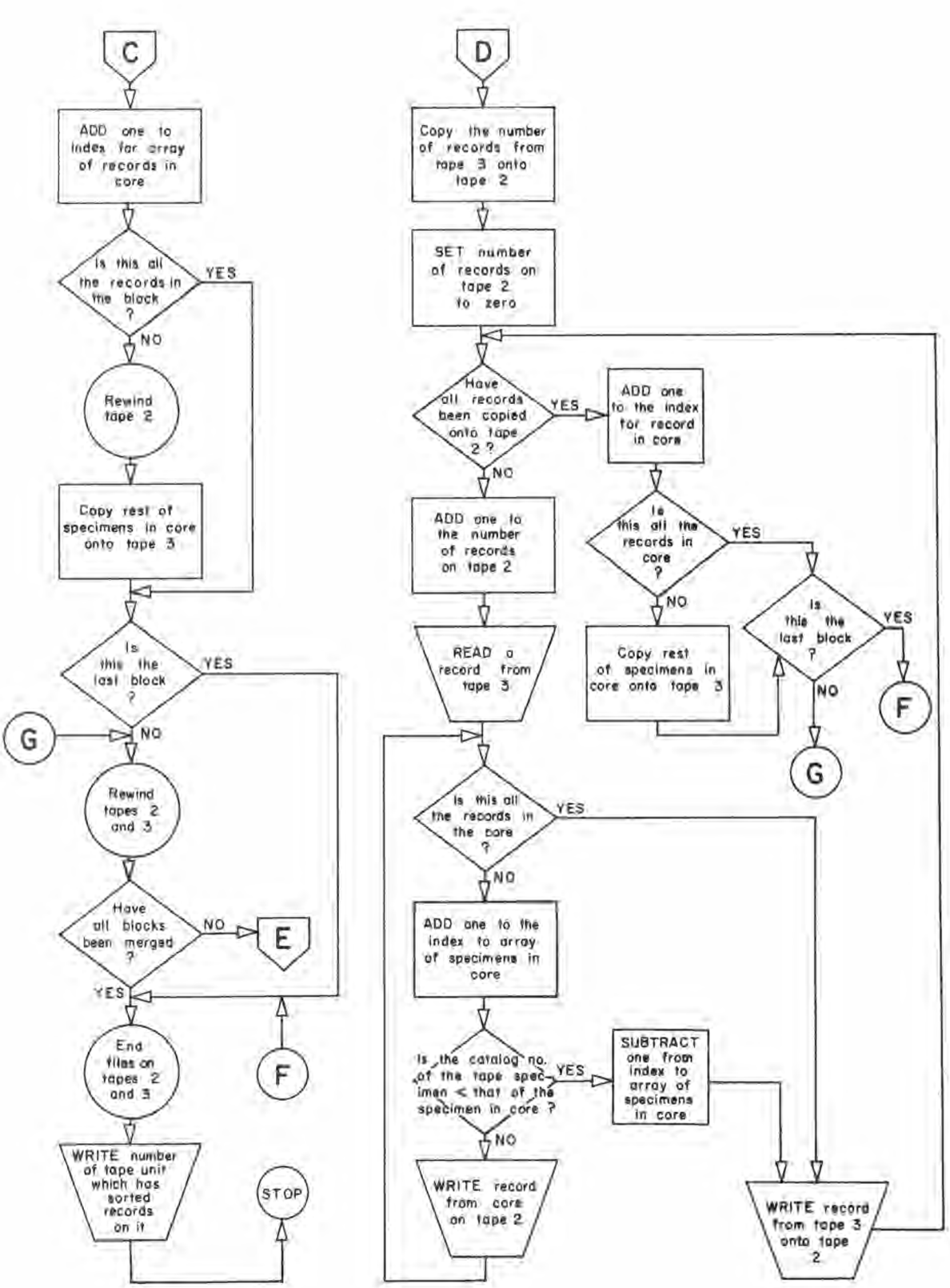

Figure 3. Flow Chart of MUZMGR (cont.).

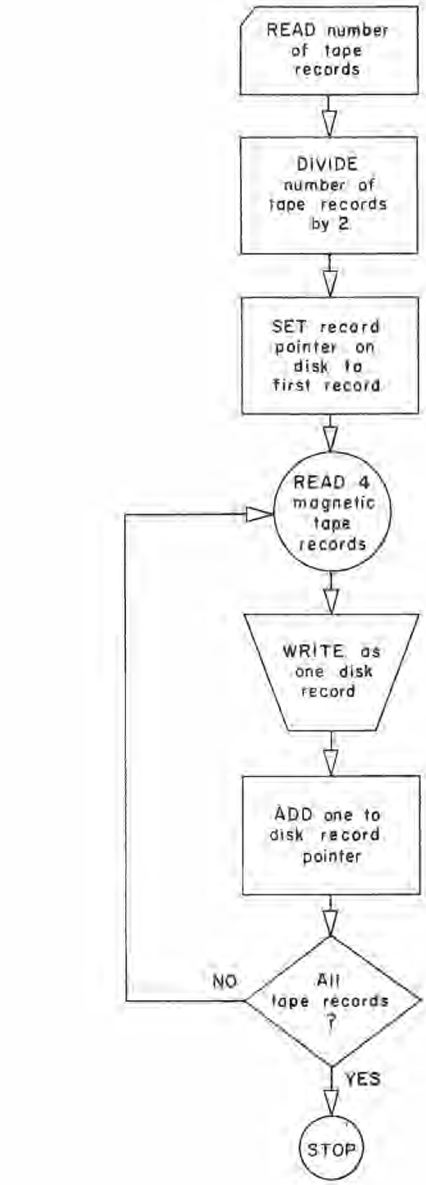

Figure 4. Flow Chart of UNBLK.

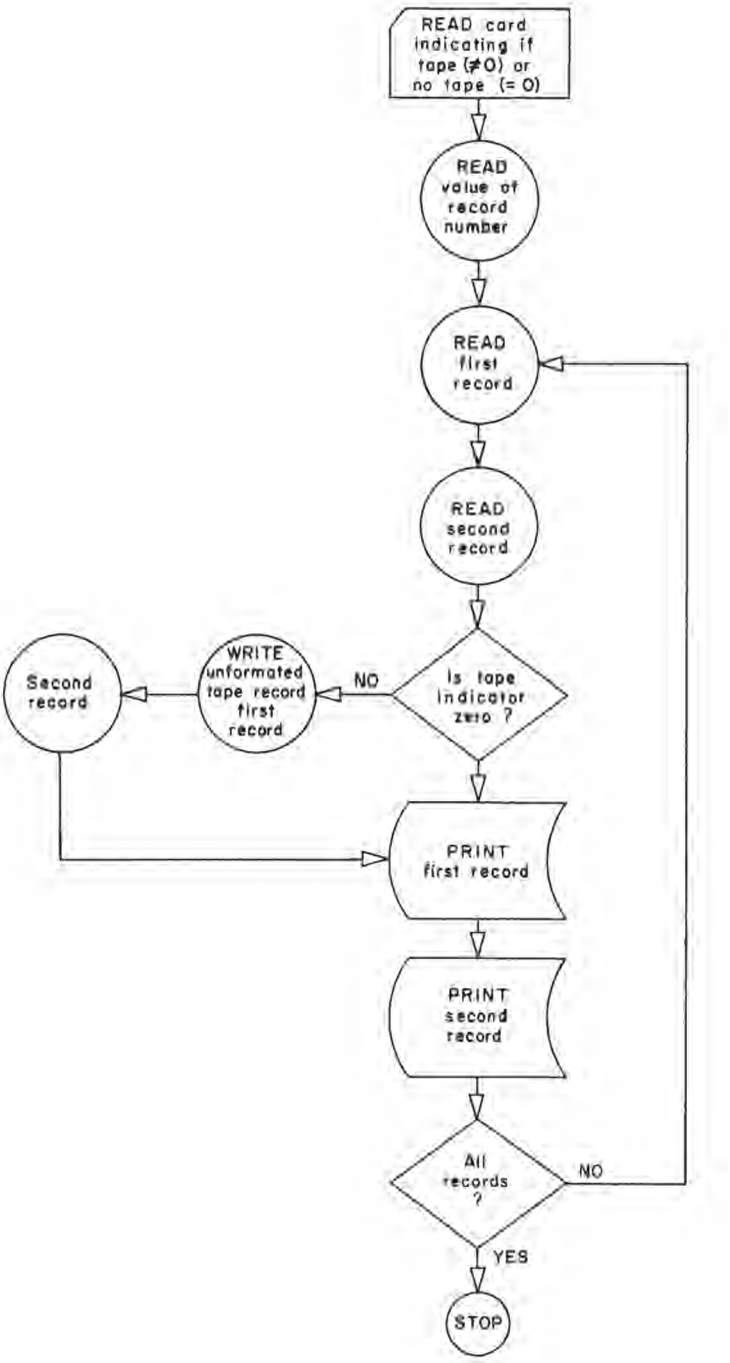

Figure 5. Flow Chart of RDR.

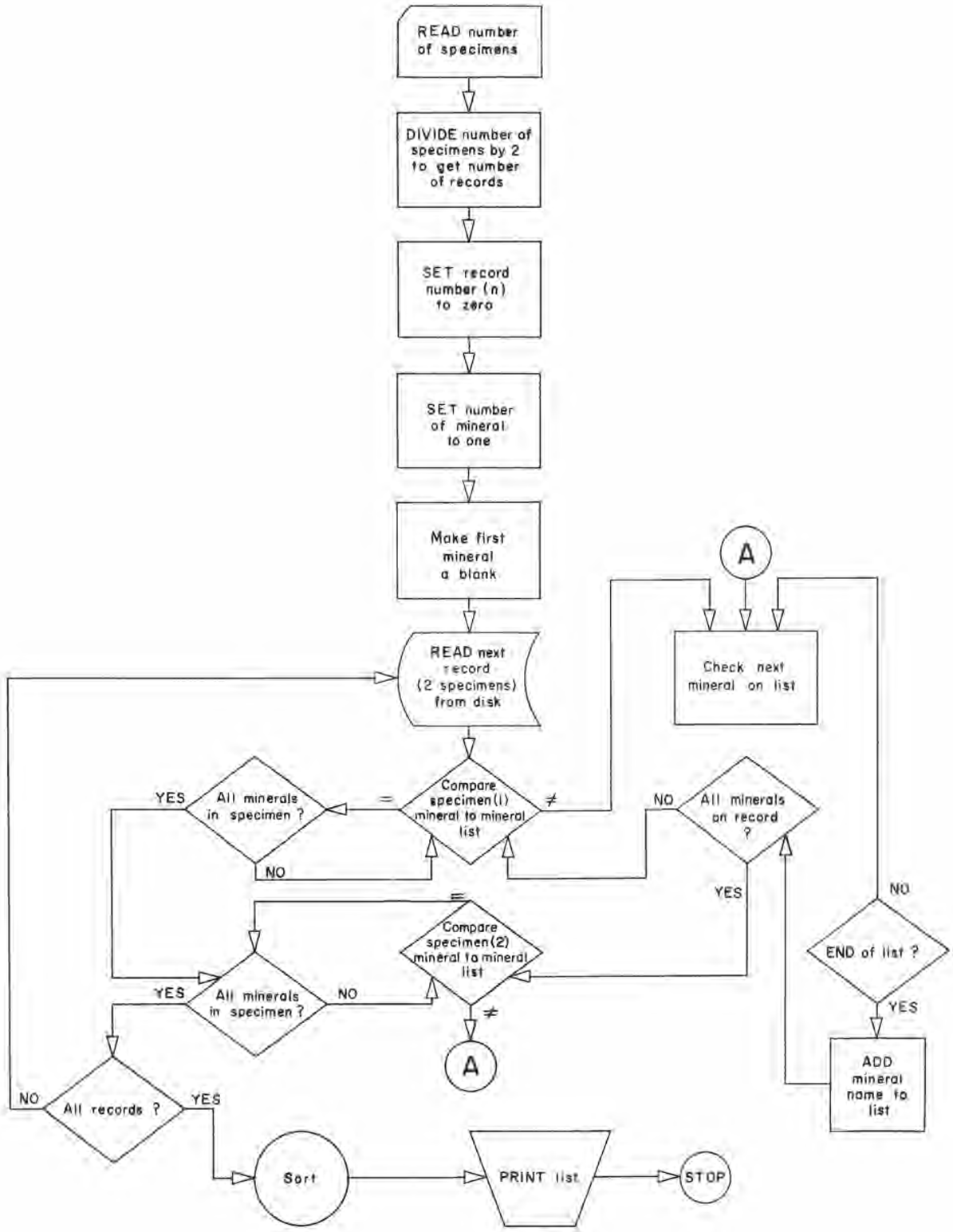

Figure 6. Flow Chart of MNLIST.

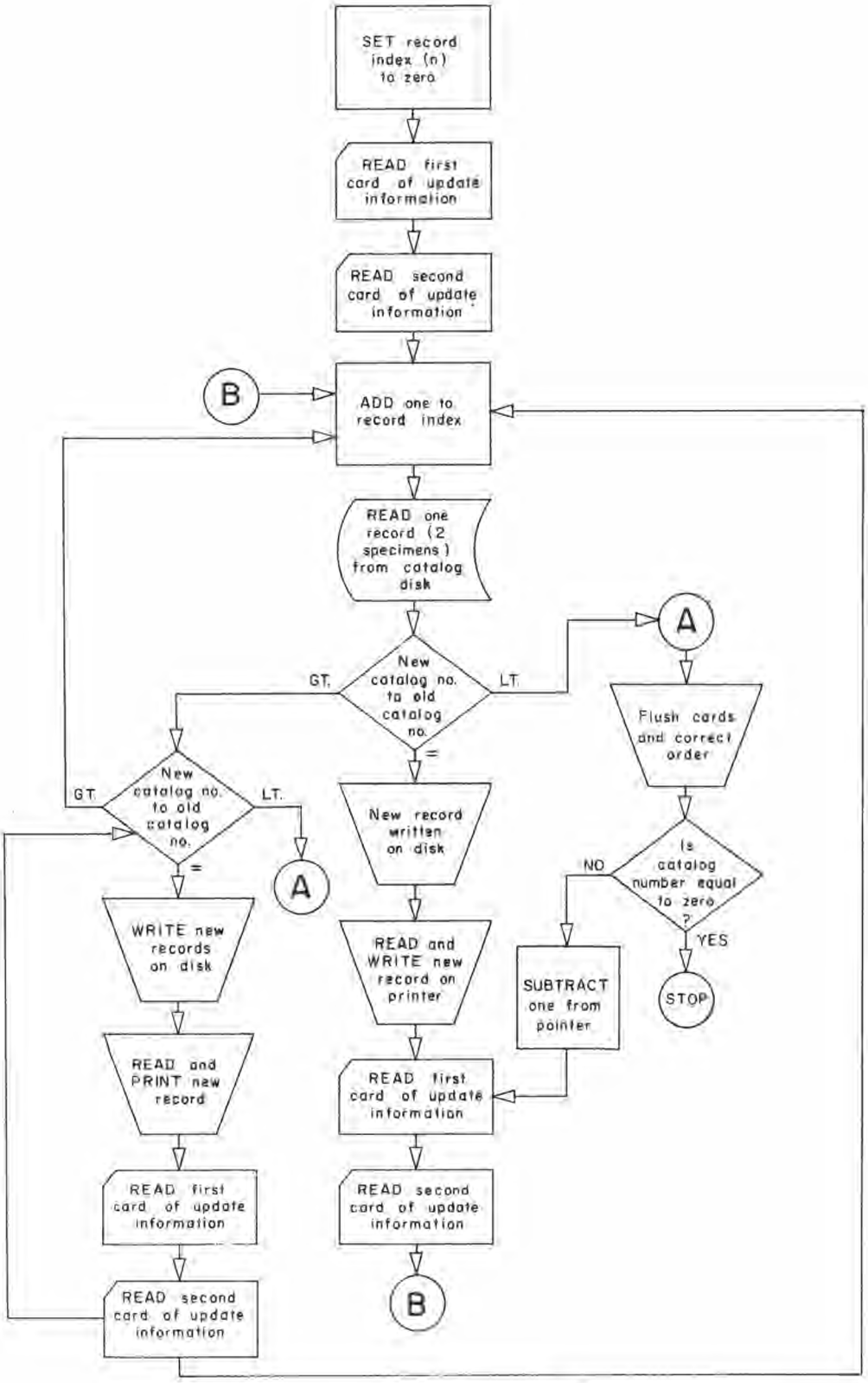

Figure 7. Flow Chart of CORRCT.

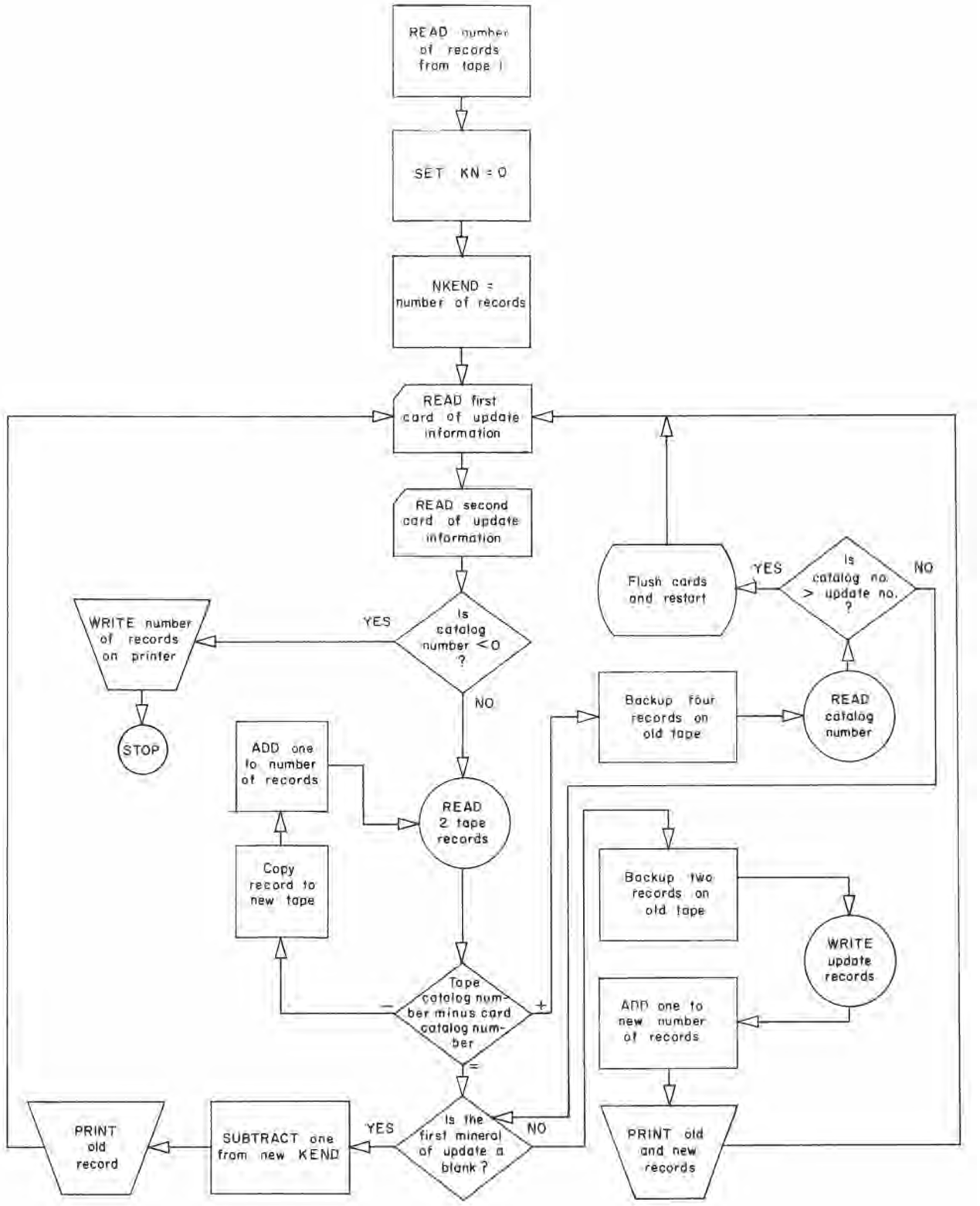

Figure 8. Flow Chart of UPDATE.

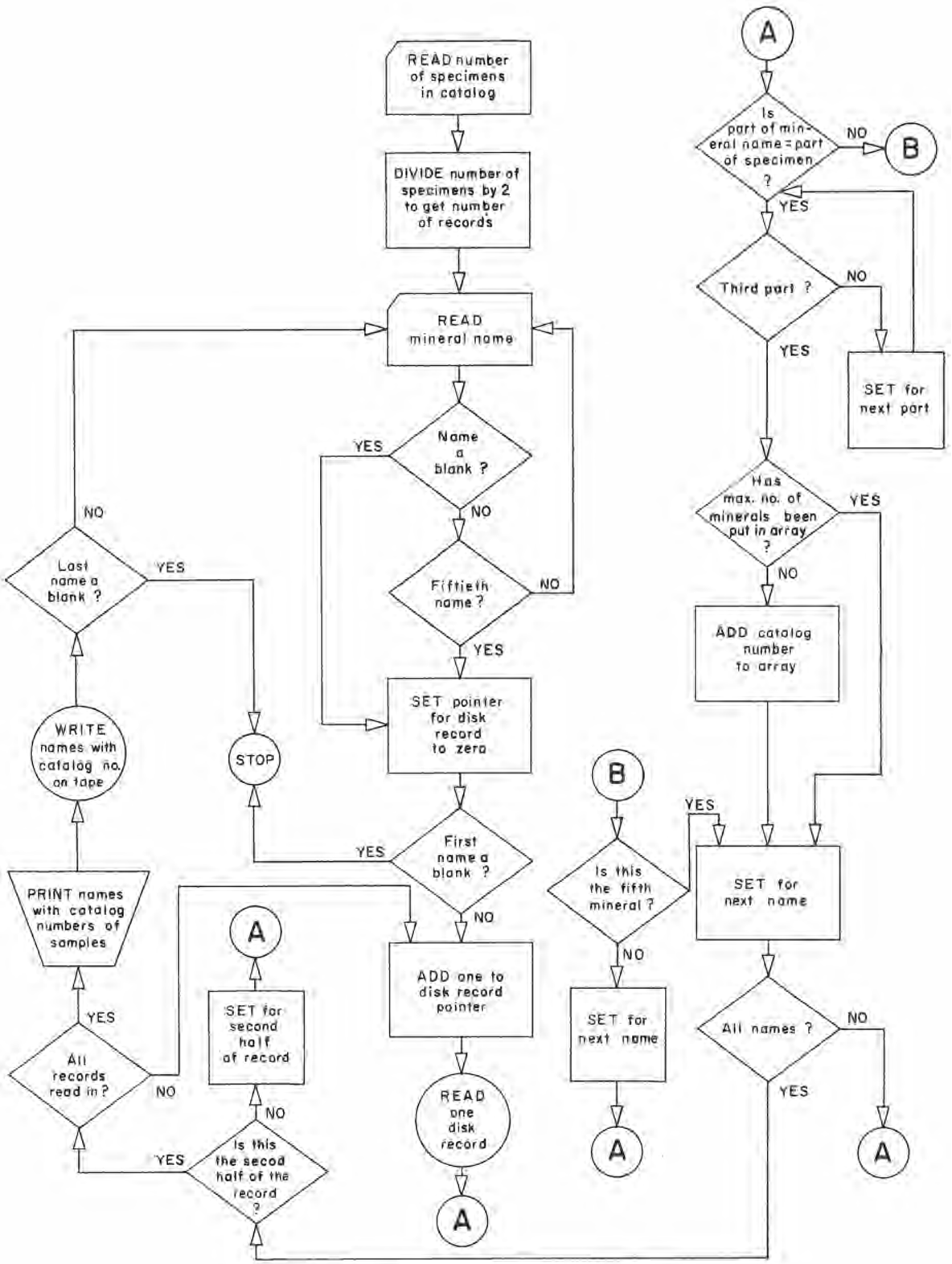

Figure 9. Flow Chart of MNRL.

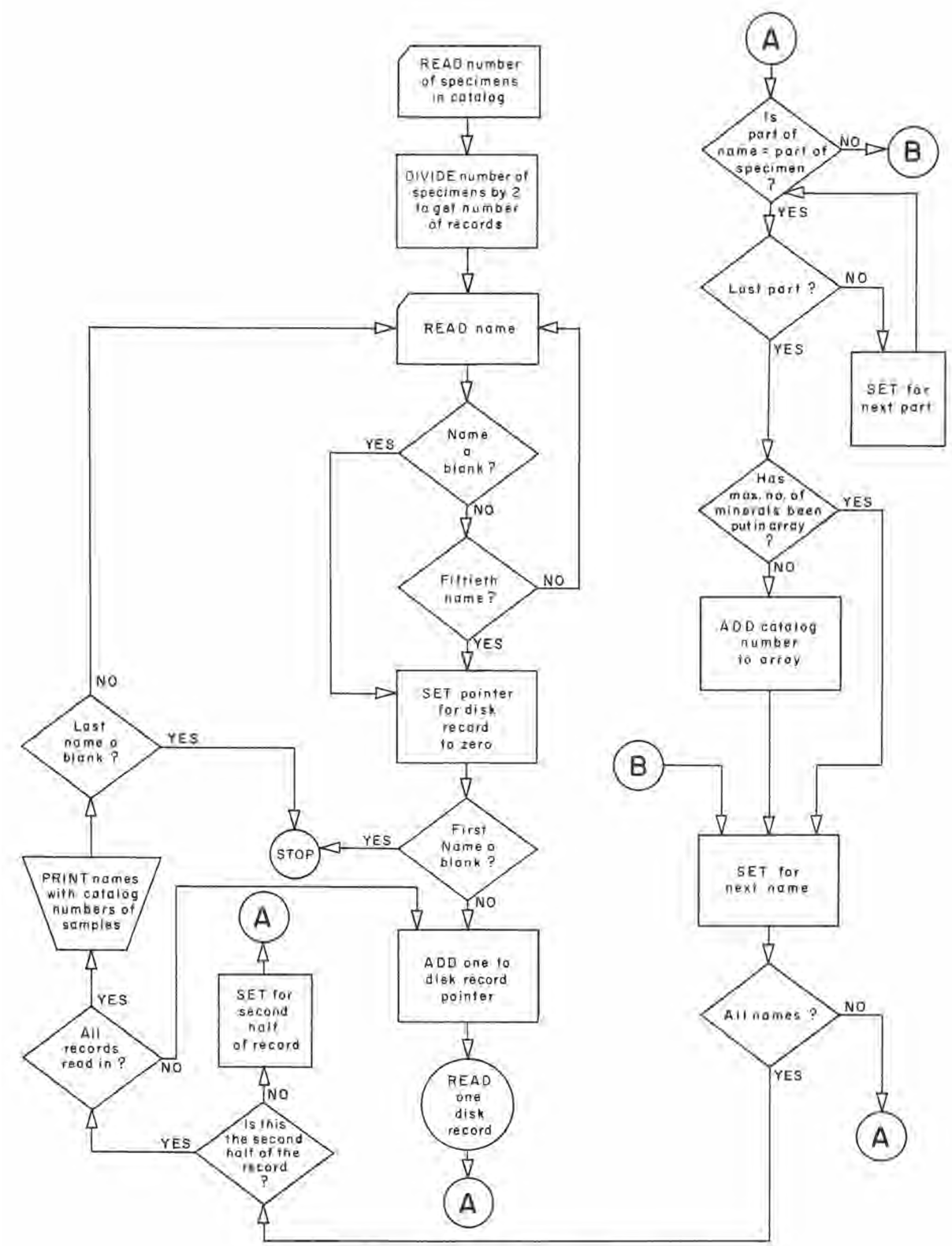

Figure 10. Flow Chart of Single-Criterion Search.

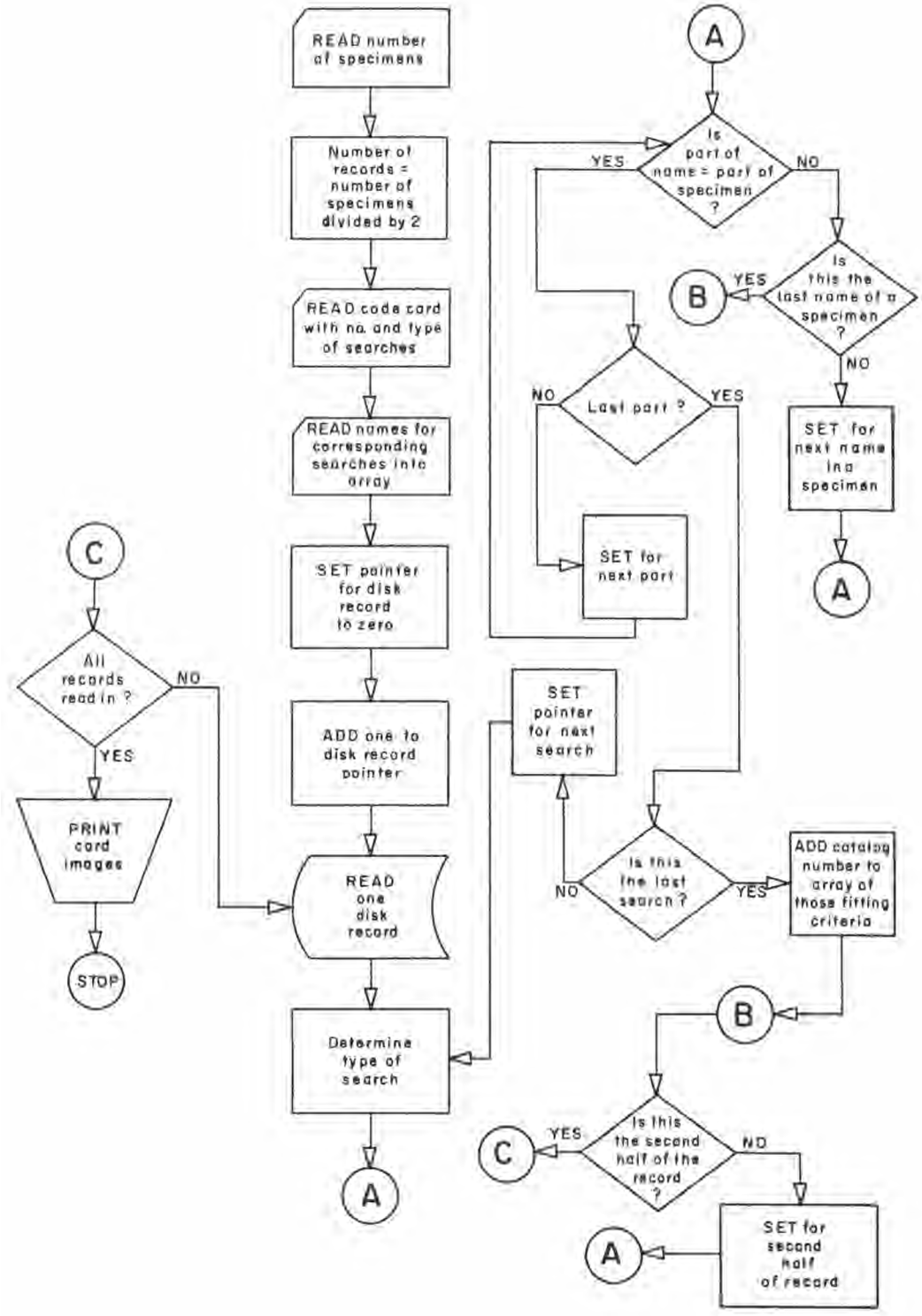

Figure 11. Flow Chart of MULSER.

APPENDIX D

Program Listings and Output Examples

MUSLDR

```
C*************************SORT INTO BLOCKS OF 500*********************************
//MUSLDR EXEC FORTRAN(MAP)
      DIMENSION MINAM(500,3,5),MINDST(500,4),LOC(500),N(11),KATNO(500,5)RWR00010
     $,NN(11)                                                          RWR00020
      INTEGER CATNO(500),DONOR(500,5),CUNTRY(500,2),STAT(500,2),       RWR00030
     $CONTY(500,2),TWN(500,3),COMNTS(500,6),PLAC                       RWR00040
      INTEGER N/'1','2','3','4','5','6','7','8','9','0',' '/          RWR00050
      INTEGER NN/1,2,3,4,5,6,7,8,9,0,0/                                RWR00060
      KEND=0                                                           RWR00180
      IEND=500                                                         RWR00190
      READ(5,1) KLIP                                                   RWR00200
    1 FORMAT(I5)                                                       RWR00210
      IF(KLIP.EQ.0) GO TO 2                                            RWR00220
C                                                                      RWR00230
CCC   KEND IS DETERMINED BY DISPLAYING ADDRESS ON THE CONSOLE AFTER    RWR00240
CCC   THE SYSTEM CANCELS THE PROGRAM                                   RWR00250
C                                                                      RWR00260
      READ(5,1) KEND                                                   RWR00270
      CALL SKPFLM(1)                                                   RWR00280
      CALL BCKALL(1)                                                   RWR00290
      GO TO 2                                                          RWR00300
C                                                                      RWR00310
CCC   THE KEND CARD IS BLANK ONLY WHEN BEGINNING A NEW TAPE.           RWR00320
C                                                                      RWR00330
    2 KLAP=0                                                           RWR00340
C                                                                      RWR00350
CCC   READING PHASE--READS IN 500 SAMPLES AT A TIME.                   RWR00360
C                                                                      RWR00370
      DO 24 INDEX=1,99999                                              RWR00380
      DO 10 IBG=1,IEND                                                 RWR00390
    3 READ(5,4,ERR=3)(KATNO(IBG,J),J=1,5),((MINAM(IBG,J,K),J=1,3),K=1,5)RWR00400
     $,(DONOR(IBG,J),J=1,5)                                            RWR00410
    4 FORMAT(5A1,3A4,3A4,3A4,3A4,3A4,5A3)                              RWR00420
      CATNO(IBG)=0                                                     RWR00430
      DO 8 K=1,5                                                       RWR00440
      DO 5 I=1,11                                                      RWR00450
      IF(KATNO(IBG,K).EQ.N(I)) GO TO 7                                 RWR00460
    5 CONTINUE                                                         RWR00470
      WRITE(15,6)                                                      RWR00480
    6 FORMAT('FLUSH & CORRECT CARD.  REPLACE IN HOPPER IN 1ST  POSITION RWR00490
     $& EOB')                                                          RWR00500
      PAUSE                                                            RWR00510
      GO TO 3                                                          RWR00520
    7 CATNO(IBG)=CATNO(IBG)+NN(I)*10**(5-K)                            RWR00530
    8 CONTINUE                                                         RWR00540
      IF(CATNO(IBG).EQ.0) GO TO 12                                     RWR00550
    9 KEND=KEND+1                                                      RWR00560
   10 READ(5,11,ERR=10) (CUNTRY(IBG,J),J=1,2),(STAT(IBG,J),J=1,2),     RWR00570
     $(CONTY(IBG,J),J=1,2),(TWN(IBG,J),J=1,3),(MINDST(IBG,J),J=1,4),   RWR00580
     $(COMNTS(IBG,J),J=1,6),LOC(IBG)                                   RWR00590
   11 FORMAT(2A4,2A4,2A4,3A4,4A4,6A4,A4)                               RWR00600
      GO TO 13                                                         RWR00610
   12 KLAP=1                                                           RWR00620
      IEND=IEND-(INDEX*500-KEND)                                       RWR00630
C     SORTING PHASE--SORTS 500 SAMPLES AT A TIME                       RWR00640
   13 IEND1=IEND-1                                                     RWR00650
      DO 20 I=1,IEND1                                                  RWR00660
      L=IEND-I                                                         RWR00670
      DO 20 M=1,L                                                      RWR00680
      IF(CATNO(M+1).GE.CATNO(M)) GO TO 20                              RWR00690
      NA=CATNO(M)                                                      RWR00700
      CATNO(M)=CATNO(M+1)                                              RWR00710
      CATNO(M+1)=NA                                                    RWR00720
      DO 14 K1=1,5                                                     RWR00730
      DO 14 J1=1,3                                                     RWR00740
      NA=MINAM(M,J1,K1)                                                RWR00750
      MINAM(M,J1,K1)=MINAM(M+1,J1,K1)                                  RWR00760
   14 MINAM(M+1,J1,K1)=NA                                              RWR00770
      DO 15 J=1,5                                                      RWR00780
      NA=DONOR(M,J1)                                                   RWR00790
      DONOR(M,J1)=DONOR(M+1,J1)                                        RWR00800
   15 DONOR(M+1,J1)=NA                                                 RWR00810
      DO 16 J1=1,2                                                     RWR00820
      NA=CUNTRY(M,J1)                                                  RWR00830
      CUNTRY(M,J1)=CUNTRY(M+1,J1)                                      RWR00840
      CUNTRY(M+1,J1)=NA                                                RWR00850
      NA=STAT(M,J1)                                                    RWR00860
      STAT(M,J1)=STAT(M+1,J1)                                          RWR00870
```

MUSLDR (cont.)

```
      STAT(M+1,J1)=NA                                                    RWR00880
      NA=CONTY(M,J1)                                                     RWR00890
      CONTY(M,J1)=CONTY(M+1,J1)                                          RWR00900
   16 CONTY(M+1,J1)=NA                                                   RWR00910
      DO 17 J1=1,3                                                       RWR00920
      NA=TWN(M,J1)                                                       RWR00930
      TWN(M,J1)=TWN(M+1,J1)                                              RWR00940
   17 TWN(M+1,J1)=NA                                                     RWR00950
      DO 18 J1=1,4                                                       RWR00960
      NA=MINDST(M,J1)                                                    RWR00970
      MINDST(M,J1)=MINDST(M+1,J1)                                        RWR00980
   18 MINDST(M+1,J1)=NA                                                  RWR00990
      DO 19 J1=1,6                                                       RWR01000
      NA=COMNTS(M,J1)                                                    RWR01010
      COMNTS(M,J1)=COMNTS(M+1,J1)                                        RWR01020
   19 COMNTS(M+1,J1)=NA                                                  RWR01030
      NA=LOC(M)                                                          RWR01040
      LOC(M)=LOC(M+1)                                                    RWR01050
      LOC(M+1)=NA                                                        RWR01060
   20 CONTINUE                                                           RWR01070
   21 DO 23 I=1,IEND                                                     RWR01080
      WRITE(1,22) CATNO(I),((MINAM(I,J,K),J=1,3),K=1,5),                 RWR01090
     $(DONOR(I,J),J=1,5)                                                 RWR01100
   22 FORMAT(I5,3A4,3A4,3A4,3A4,3A4,5A3)                                 RWR31105
      WRITE(1,11) (COUNTRY(I,J),J=1,2),(STAT(I,J),J=1,2),                RWR0111
     $(CONTY(I,J),J=1,2),(TWN(I,J),J=1,3),(MINDST(I,J),J=1,4),           RWR01120
     $(COMNTS(I,J),J=1,6),LOC(I)                                         RWR01130
   23 CONTINUE                                                           RWR01140
      IF(KLAP)25,24,25                                                   RWR01150
   24 CONTINUE                                                           RWR01160
   25 WRITE(6,1) KEND                                                    RWR01170
      STOP                                                               RWR01180
      END                                                                RWR01190
/*
// EXEC LNKEDT(MAP)
/*
/$ DRC TAPE(1,180)=SCRTCH,16000
/$ DRC PRINT,16000
// EXEC
/*
/&
```

MUZMGR

```
C*************************MERGE SORTED BLOCKS OF 500*****************************
//MUZMGR EXEC FORTRAN
      INTEGER CATNO(500),DONOR(500,5),CUNTRY(500,2),STAT(500,2),          RWR00010
     $CONTY(500,2),TWN(500,3),COMNTS(500,6),CATNOA,DONORA(5),CUNTRA(2),   RWR00020
     $STATA(2),CONTYA(2),TWNA(3),COMNTA(6)                                RWR00030
      DIMENSION MINAM(500,3,5),MINDST(500,4),LOC(500),MINAMA(3,5),        RWR00040
     $MINDTA(4)                                                           RWR00050
      READ(5,1) KEND                                                      RWR00060
    1 FORMAT(I5)                                                          RWR00070
      IF(KEND.LE.500) GO TO 99                                            RWR00080
      WRITE(2,1) KEND                                                     RWR00090
C                                                                         RWR00100
CCC   READS 500 SAMPLES INTO CORE THEN PUTS THEM ON TAPE UNIT 2 (FROM     RWR00110
CCC   TAPE UNIT 1)                                                        RWR00120
C                                                                         RWR00130
      DO 4 I=1,500                                                        RWR00140
      READ(1,2) CATNO(I),((MINAM(I,J,K),J=1,3),K=1,5),                    RWR00150
     $(DONOR(I,J),J=1,5)                                                  RWR00160
    2 FORMAT(I5,3A4,3A4,3A4,3A4,3A4,5A3)                                  RWR00170
      READ(1,3) (CUNTRY(I,J),J=1,2),(STAT(I,J),J=1,2),                    RWR00180
     $(CONTY(I,J),J=1,2),(TWN(I,J),J=1,3),(MINDST(I,J),J=1,4),            RWR00190
     $(COMNTS(I,J),J=1,6),LOC(I)                                          RWR00200
    3 FORMAT(2A4,2A4,2A4,3A4,4A4,6A4,A4)                                  RWR00205
      WRITE(2,2) CATNO(I),((MINAM(I,J,K),J=1,3),K=1,5),                   RWR00210
     $(DONOR(I,J),J=1,5)                                                  RWR00220
    4 WRITE(2,3) (CUNTRY(I,J),J=1,2),(STAT(I,J),J=1,2),                   RWR00230
     $(CONTY(I,J),J=1,2),(TWN(I,J),J=1,3),(MINDST(I,J),J=1,4),            RWR00240
     $(COMNTS(I,J),J=1,6),LOC(I)                                          RWR00250
      REWIND 2                                                            RWR00260
      KFINI=KEND/500-1                                                    RWR00270
      IF(MOD(KEND,500).GT.0) KFINI=KFINI+1                                RWR00280
C                                                                         RWR00290
CCC   MERGING SECTION                                                     RWR00300
C                                                                         RWR00310
      IG=0                                                                RWR00320
      IE=500                                                              RWR00330
      DO 23 INDEX=1,KFINI                                                 RWR00340
      IG=IG+500                                                           RWR00350
      IF(INDEX.EQ.KFINI) GO TO 5                                          RWR00360
      GO TO 6                                                             RWR00370
    5 IE=KEND-IG                                                          RWR00380
C                                                                         RWR00390
CCC   READS A SECOND BLOCK OF 500 SAMPLES INTO CORE (ALSO THE THIRD,      RWR00400
CCC   FORTH, ETC. TO THE (N)TH BLOCK OF 500) FROM TAPE UNIT 1.            RWR00410
C                                                                         RWR00420
    6 DO 7 MG=1,IE                                                        RWR00430
      READ(1,2) CATNO(MG),((MINAM(MG,J,K),J=1,3),K=1,5),                  RWR00440
     $(DONOR(MG,J),J=1,5)                                                 RWR00450
    7 READ(1,3) (CUNTRY(MG,J),J=1,2),(STAT(MG,J),J=1,2),                 RWR00460
     $(CONTY(MG,J),J=1,2),(TWN(MG,J),J=1,3),(MINDST(MG,J),J=1,4),         RWR00470
     $(COMNTS(MG,J),J=1,6),LOC(MG)                                        RWR00480
      KR=0                                                                RWR00490
      IF(MOD(INDEX,2).LE.0) GO TO 15                                      RWR00500
C                                                                         RWR00510
CCC   READS ONE SAMPLE AT A TIME FROM TAPE UNIT 2 AND COMPARES IT WITH    RWR00520
CCC   THOSE IN CORE.                                                      RWR00530
C                                                                         RWR00540
      READ(2,1) KEND                                                      RWR00550
      WRITE(3,1) KEND                                                     RWR00560
      INTAP=0                                                             RWR00570
    8 IF(INTAP.EQ.IG) GO TO 12                                            RWR00580
      INTAP=INTAP+1                                                       RWR00590
      READ(2,2) CATNOA,((MINAMA(J,K),J=1,3),K=1,5),                       RWR00600
     $(DONORA(J),J=1,5)                                                   RWR00610
      READ(2,3) (CUNTRA(J),J=1,2),(STATA(J),J=1,2),                       RWR00620
     $(CONTYA(J),J=1,2),(TWNA(J),J=1,3),(MINDTA(J),J=1,4),                RWR00630
     $(COMNTA(J),J=1,6),LOCA                                              RWR00640
    9 IF(KR.EQ.500) GO TO 11                                              RWR00650
      KR=KR+1                                                             RWR00660
      IF(CATNOA.LT.CATNO(KR)) GO TO 10                                    RWR00670
      WRITE(3,2) CATNO(KR),((MINAM(KR,J,K),J=1,3),K=1,5),                 RWR00680
     $(DONOR(KR,J),J=1,5)                                                 RWR00690
      WRITE(3,3) (CUNTRY(KR,J),J=1,2),(STAT(KR,J),J=1,2),                 RWR00700
     $(CONTY(KR,J),J=1,2),(TWN(KR,J),J=1,3),(MINDST(KR,J),J=1,4),         RWR00710
     $(COMNTS(KR,J),J=1,6),LOC(KR)                                        RWR00720
      GO TO 9                                                             RWR00730
   10 KR=KR-1                                                             RWR00740
   11 WRITE(3,2) CATNOA,((MINAMA(J,K),J=1,3),K=1,5),                      RWR00750
```

MUZMGR (cont.)

```
     $(DONORA(J),J=1,5)                                                    RWR00760
      WRITE(3,3) (CUNTRA(J),J=1,2),(STATA(J),J=1,2),                       RWR00770
     $(CONTYA(J),J=1,2),(TWNA(J),J=1,3),(MINDTA(J),J=1,4),                 RWR00780
     $(COMNTA(J),J=1,6),LOCA                                               RWR00790
      GO TO 8                                                              RWR00800
   12 KR=KR+1                                                              RWR00810
      IF(KR.GT.IE) GO TO 14                                                RWR00820
      REWIND 2                                                             RWR00830
      DO 13 JT=KR,IE                                                       RWR00840
      WRITE(3,2) CATNO(JT),((MINAM(JT,J,K),J=1,3),K=1,5),                  RWR00850
     $(DONOR(JT,J),J=1,5)                                                  RWR00860
   13 WRITE(3,3) (CUNTRY(JT,J),J=1,2),(STAT(JT,J),J=1,2),                  RWR00870
     $(CONTY(JT,J),J=1,2),(TWN(JT,J),J=1,3),(MINDST(JT,J),J=1,4),          RWR00880
     $(COMNTS(JT,J),J=1,6),LOC(JT)                                         RWR00890
   14 IF(INDEX.EQ.KFINI) GO TO 24                                          RWR00900
      REWIND 3                                                             RWR00910
      REWIND 2                                                             RWR00920
      GO TO 23                                                             RWR00930
   15 READ(3,1) KEND                                                       RWR00940
      WRITE(2,1) KEND                                                      RWR00950
      INTAP=0                                                              RWR00960
   16 IF(INTAP.EQ.IG) GO TO 20                                             RWR00970
      INTAP=INTAP+1                                                        RWR00980
      READ(3,2) CATNOA,((MINAMA(J,K),J=1,3),K=1,5),                        RWR00990
     $(DONORA(J),J=1,5)                                                    RWR01000
      READ(3,3) (CUNTRA(J),J=1,2),(STATA(J),J=1,2),                        RWR01010
     $(CONTYA(J),J=1,2),(TWNA(J),J=1,3),(MINDTA(J),J=1,4),                 RWR01020
     $(COMNTA(J),J=1,6),LOCA                                               RWR01030
   17 IF(KR.EQ.500) GO TO 19                                               RWR01040
      KR=KR+1                                                              RWR01050
      IF(CATNOA.LT.CATNO(KR)) GO TO 18                                     RWR01060
      WRITE(2,2) CATNO(KR),((MINAM(KR,J,K),J=1,3),K=1,5),                  RWR01070
     $(DONOR(KR,J),J=1,5)                                                  RWR01080
      WRITE(2,3) (CUNTRY(KR,J),J=1,2),(STAT(KR,J),J=1,2),                  RWR01090
     $(CONTY(KR,J),J=1,2),(TWN(KR,J),J=1,3),(MINDST(KR,J),J=1,4),          RWR01100
     $(COMNTS(KR,J),J=1,6),LOC(KR)                                         RWR01110
      GO TO 17                                                             RWR01120
   18 KR=KR-1                                                              RWR01130
   19 WRITE(2,2) CATNOA,((MINAMA(J,K),J=1,3),K=1,5),                       RWR01140
     $(DONORA(J),J=1,5)                                                    RWR01150
      WRITE(2,3) (CUNTRA(J),J=1,2),(STATA(J),J=1,2),                       RWR01160
     $(CONTYA(J),J=1,2),(TWNA(J),J=1,3),(MINDTA(J),J=1,4),                 RWR01170
     $(COMNTA(J),J=1,6),LOCA                                               RWR01180
      GO TO 16                                                             RWR01190
   20 KR=KR+1                                                              RWR01200
      IF(KR.GT.IE) GO TO 22                                                RWR01210
      REWIND 3                                                             RWR01220
      DO 21 JT=KR,IE                                                       RWR01230
      WRITE(2,2) CATNO(JT),((MINAM(JT,J,K),J=1,3),K=1,5),                  RWR01240
     $(DONOR(JT,J),J=1,5)                                                  RWR01250
   21 WRITE(2,3) (CUNTRY(JT,J),J=1,2),(STAT(JT,J),J=1,2),                  RWR01260
     $(CONTY(JT,J),J=1,2),(TWN(JT,J),J=1,3),(MINDST(JT,J),J=1,4),          RWR01270
     $(COMNTS(JT,J),J=1,6),LOC(JT)                                         RWR01280
   22 IF(INDEX.EQ.KFINI) GO TO 24                                          RWR01290
      REWIND 2                                                             RWR01300
      REWIND 3                                                             RWR01310
   23 CONTINUE                                                             RWR01320
   24 END FILE 2                                                           RWR01330
      END FILE 3                                                           RWR01340
      IF(MOD(KFINI,2).GT.0) GO TO 27                                       RWR/1350
   25 FORMAT('0','NEW MUSEUM FILE IS ON TAPE DRIVE 183')                   RWR01360
      WRITE(6,26)                                                          RWR01370
   26 FORMAT('0','NEW MUSEUM FILE IS ON TAPE DRIVE 182')                   RWR01380
      GO TO 99                                                             RWR01390
   27 WRITE(6,25)                                                          RWR01400
   99 STOP                                                                 RWR01410
      END                                                                  RWR01420
/*
// EXEC LNKEDT
/*
// EXEC
 5258
/*
/&
```

UNBLK

```
C      **********************UNBLOCK-REBLOCK*****************************
//UNBLK EXEC FORTRAN
      DEFINE FILE 1(3155,360,E,KDSK)
      INTEGER CATNOA(2),DONORA(5,2),CUNTRA(2,2),STATA(2,2),CONTYA(2,2),
     $TWNA(3,2),COMNTA(6,2),MINAMA(3,5,2),MINDIA(4,2),LOCA(2)
      READ(5,1)KEND
    1 FORMAT(I5)
      NEND=KEND/2
      N=1
      DO 7 I=1,NEND
      READ(2,3,ERR=8)CATNOA(1),((MINAMA(J,K,1),J=1,3),K=1,5),
     $(DONORA(J,1),J=1,5)
    3 FORMAT(I5,5(3A4),5A3)
      READ(2,4,ERR=8)(CUNTRA(J,1),J=1,2),(STATA(J,1),J=1,2),
     $(CONTYA(J,1),J=1,2),(TWNA(J,1),J=1,3),(MINDIA(J,1),J=1,4),
     $(COMNTA(J,1),J=1,6),LOCA(1)
    4 FORMAT(3(2A4),3A4,4A4,6A4,A4)
      READ(2,3,ERR=8)CATNOA(2),((MINAMA(J,K,2),J=1,3),K=1,5),
     $(DONORA(J,2),J=1,5)
      READ(2,4,ERR=8)(CUNTRA(J,2),J=1,2),(STATA(J,2),J=1,2),
     $(CONTYA(J,2),J=1,2),(TWNA(J,2),J=1,3),(MINDIA(J,2),J=1,4),
     $(COMNTA(J,2),J=1,6),LOCA(2)
    5 WRITE(1'N,6)CATNOA(1),((MINAMA(J,K,1),J=1,3),K=1,5),
     $(DONORA(J,1),J=1,5),(CUNTRA(J,1),J=1,2),(STATA(J,1),J=1,2),
     $(CONTYA(J,1),J=1,2),(TWNA(J,1),J=1,3),(MINDIA(J,1),J=1,4),
     $(COMNTA(J,1),J=1,6),LOCA(1),CATNOA(2),
     $((MINAMA(J,K,2),J=1,3),K=1,5),(DONORA(J,2),J=1,5),
     $(CUNTRA(J,2),J=1,2),(STATA(J,2),J=1,2),
     $(CONTYA(J,2),J=1,2),(TWNA(J,2),J=1,3),(MINDIA(J,2),J=1,4),
     $(COMNTA(J,2),J=1,6),LOCA(2)
    6 FORMAT(2(I5,5(3A4),5A3,3(2A4),3A4,4A4,6A4,A4))
      N=N+1
    7 CONTINUE
      READ(5,3,ERR=8)CATNOA(1),((MINAMA(J,K,1),J=1,3),K=1,5),
     $(DONORA(J,1),J=1,5)
      READ(5,4,ERR=8)(CUNTRA(J,1),J=1,2),(STATA(J,1),J=1,2),
     $(CONTYA(J,1),J=1,2),(TWNA(J,1),J=1,3),(MINDIA(J,1),J=1,4),
     $(COMNTA(J,1),J=1,6),LOCA(1)
      READ(5,3,ERR=8)CATNOA(2),((MINAMA(J,K,2),J=1,3),K=1,5),
     $(DONORA(J,2),J=1,5)
      READ(5,4,ERR=8)(CUNTRA(J,2),J=1,2),(STATA(J,2),J=1,2),
     $(CONTYA(J,2),J=1,2),(TWNA(J,2),J=1,3),(MINDIA(J,2),J=1,4),
     $(COMNTA(J,2),J=1,6),LOCA(2)
      WRITE(1'N,6)CATNOA(1),((MINAMA(J,K,1),J=1,3),K=1,5),
     $(DONORA(J,1),J=1,5),(CUNTRA(J,1),J=1,2),(STATA(J,1),J=1,2),
     $(CONTYA(J,1),J=1,2),(TWNA(J,1),J=1,3),(MINDIA(J,1),J=1,4),
     $(COMNTA(J,1),J=1,6),LOCA(1),CATNOA(2),
     $((MINAMA(J,K,2),J=1,3),K=1,5),(DONORA(J,2),J=1,5),
     $(CUNTRA(J,2),J=1,2),(STATA(J,2),J=1,2),
     $(CONTYA(J,2),J=1,2),(TWNA(J,2),J=1,3),(MINDIA(J,2),J=1,4),
     $(COMNTA(J,2),J=1,6),LOCA(2)
      GO TO 9
    8 PAUSE 'READ ERROR ON TAPE OR CARD READER'
      WRITE(6,1)N
    9 STOP
      END
/*
// EXEC LNKEDT
/*
/$ DRC DISK(1,190)=MUS001(SDSBUM),16000
/$ DRC TAPE(2,180)=000128,16000
// EXEC
 5273
/*
/&
```

RDR

```
C*************************COPY MUSEUM TAPE*********************************
//RDR  EXEC FORTRAN
       INTEGER CATNOA,DONORA(5),CUNTRA(2),STATA(2),CONTYA(2),TWNA(3),     RWR00010
      $COMNTA(6)                                                          RWR00020
       DIMENSION MINAMA(3,5),MINDTA(4)                                    RWR00030
       READ(5,1) MAKTAP                                                   RWR00120
       READ(1,1) KEND                                                     RWR00130
     1 FORMAT(I5)                                                         RWR00140
       DO 8 I=1,KEND                                                      RWR00150
       READ(1,2) CATNOA,((MINAMA(J,K),J=1,3),K=1,5),(DONORA(J),J=1,5)     RWR00160
     2 FORMAT(I5,3A4,3A4,3A4,3A4,3A4,5A3)                                 RWR00170
       READ(1,3) (CUNTRA(J),J=1,2),(STATA(J),J=1,2),(CONTYA(J),J=1,2),    RWR00180
      $(TWNA(J),J=1,3),(MINDTA(J),J=1,4),(COMNTA(J),J=1,6),LOCA           RWR00190
     3 FORMAT(2A4,2A4,2A4,3A4,4A4,6A4,A4)                                 RWR00200
       IF(MAKTAP.EQ.0) GO TO 4                                            RWR00210
       WRITE(2)    CATNOA,((MINAMA(J,K),J=1,3),K=1,5),(DONORA(J),J=1,5)   RWR00220
       WRITE(2)    (CUNTRA(J),J=1,2),(STATA(J),J=1,2),(CONTYA(J),J=1,2),  RWR00230
      $(TWNA(J),J=1,3),(MINDTA(J),J=1,4),(COMNTA(J),J=1,6),LOCA           RWR00240
     4 WRITE(6,5) CATNOA,((MINAMA(J,K),J=1,3),K=1,5),(DONORA(J),J=1,5)    RWR00250
     5 FORMAT(' ',I5,2X,3A4,2X,3A4,2X,3A4,2X,3A4,2X,3A4,2X,5A3)           RWR00260
     6 WRITE(6,7) (CUNTRA(J),J=1,2),(STATA(J),J=1,2),(CONTYA(J),J=1,2),   RWR00270
      $(TWNA(J),J=1,3),(MINDTA(J),J=1,4),(COMNTA(J),J=1,6),LOCA           RWR00280
     7 FORMAT(' ',5X,2A4,2X,2A4,2X,2A4,2X,3A4,2X,4A4,2X,6A4,2X,A4/)       RWR00290
     8 CONTINUE                                                           RWR00300
       STOP                                                               RWR00310
       END                                                                RWR00320
// EXEC LNKEDT
//SYS002 ACCESS SDS002,180='MASTER'
//SYS003 ACCESS SDS003,182='COPYWO'
/*
// EXEC
  1
/*
/&
```

```
C      ********************ALPHABETICAL MINERAL LISTING*****************<
//MNLIST EXEC FORTRAN
       DIMENSION MINAM1(3,5),MINAM2(3,5),KNTRY1(2),KNTRY2(2),
      #MDST1(4),MDST2(4),MLOG(3,6000)
       INTEGER DNR1(5),DNR2(5),CATNO1,CATNO2,CNTY1(2),CNTY2(2),
      #STAT1(2),STAT2(2),TWN1(3),TWN2(3),COMNT1(6),COMNT2(6)
       DEFINE FILE 9(3155,360,E,KDSK)
       DATA KBLNK/'    '/
C
C
       READ(5,1)MAXSPL
     1 FORMAT(I6)
       MAXREC=MAXSPL/2
       N=0
       J=1
       MLOG(1,1)=KBLNK
       MLOG(2,1)=KBLNK
       MLOG(3,1)=KBLNK
       DO 14 INDEX=1,MAXREC
       N=N+1
       READ(9'N,2) CATNO1,((MINAM1(I,K),I=1,3),K=1,5),(DNR1(I),I=1,5),
      #     (KNTRY1(I),I=1,2),(STAT1(I),I=1,2),(CNTY1(I),I=1,2),
      #     (TWN1(I),I=1,3),(MDST1(I),I=1,4),(COMNT1(I),I=1,6),LOC1,
      #CATNO2,((MINAM2(I,K),I=1,3),K=1,5),(DNR2(I),I=1,5),
      #     (KNTRY2(I),I=1,2),(STAT2(I),I=1,2),(CNTY2(I),I=1,2),
      #     (TWN2(I),I=1,3),(MDST2(I),I=1,4),(COMNT2(I),I=1,6),LOC2
     2 FORMAT(2(I5,5(3A4),5A3,3(2A4),3A4,4A4,6A4,A4))
       DO 7 NN=1,5
       IF(MINAM1(1,NN).EQ.KBLNK) GO TO 8
       DO 5 L=1,J
       IF(MINAM1(1,NN).NE.MLOG(1,L)) GO TO 5
     3 DO 4 M=2,3
       IF(MINAM1(M,NN).NE.MLOG(M,L)) GO TO 5
     4 CONTINUE
       GO TO 7
     5 CONTINUE
       J=J+1
       IF(J.GT.6000) GO TO 15
       DO 6 K=1,3
     6 MLOG(K,J)=MINAM1(K,NN)
     7 CONTINUE
     8 DO 13 NN=1,5
       IF(MINAM2(1,NN).EQ.KBLNK) GO TO 14
       DO 11 L=1,J
       IF(MINAM2(1,NN).NE.MLOG(1,L)) GO TO 11
     9 DO 10 M=2,3
       IF(MINAM2(M,NN).NE.MLOG(M,L)) GO TO 11
    10 CONTINUE
       GO TO 13
    11 CONTINUE
       J=J+1
       IF(J.GT.6000) GO TO 15
       DO 12 K=1,3
    12 MLOG(K,J)=MINAM2(K,NN)
    13 CONTINUE
    14 CONTINUE
    15 WRITE(6,16)CATNO1
    16 FORMAT('0','ENDING CATALOG NUMBER WAS ',I5)
       J=J-1
       CALL CORSRT(MLOG,0,12,12,J)
       WRITE(6,17)((MLOG(MN,MM),MN=1,3),MM=1,J)
    17 FORMAT(' ',8(3X,3A4))
       STOP
       END
/*
// EXEC LNKEDT
/*
/$ DRC PRINT,16000
/$ DRC DISK(9,190)=MUS001(SDSBUM),16000
// EXEC
  5274
/*
/&
```

CORRCT

```
C      ************************CORRECTION OF DISK*****************************
//CORRCT EXEC FORTRAN
       INTEGER CATNOA(2),DONORA(5,2),CUNTRA(2,2),STATA(2,2),CONTYA(2,2),
      $TWNA(3,2),COMNTA(6,2),MINAMA(3,5,2),MINDIA(4,2),LOCA(2),
      $NMINAM(3,5),NDONOR(5),
      $NCUNTR(2),NSTAT(2),NCONTY(2),NTWNA(3),NCOMNT(6),NMINDI(4),NLOCA
       DEFINE FILE 1(3155,360,E,KDSK)
       N=0
   17  READ(5,1,ERR=17)NCATNO,((NMINAM(J,K),J=1,3),K=1,5),
      $(NDONOR(J),J=1,5)
    1  FORMAT(I5,5(3A4),5A3)
       READ(5,2)(NCUNTR(J),J=1,2),(NSTAT(J),J=1,2),(NCONTY(J),J=1,2),
      $(NTWNA(J),J=1,3),
      $(NMINDI(J),J=1,4),(NCOMNT(J),J=1,6),NLOCA
    2  FORMAT(3(2A4),3A4,4A4,6A4,A4)
    3  FORMAT('0',I5,5(3A4),5A3)
    4  FORMAT(' ',3(2A4),3A4,4A4,6A4,A4)
    5  N=N+1
       READ(1'N,6)CATNOA(1),((MINAMA(J,K,1),J=1,3),K=1,5),
      $(DONORA(J,1),J=1,5),(CUNTRA(J,1),J=1,2),(STATA(J,1),J=1,2),
      $(CONTYA(J,1),J=1,2),(TWNA(J,1),J=1,3),(MINDIA(J,1),J=1,4),
      $(COMNTA(J,1),J=1,6),LOCA(1),CATNOA(2),
      $((MINAMA(J,K,2),J=1,3),K=1,5),(DONORA(J,2),J=1,5),
      $(CUNTRA(J,2),J=1,2),(STATA(J,2),J=1,2),
      $(CONTYA(J,2),J=1,2),(TWNA(J,2),J=1,3),(MINDIA(J,2),J=1,4),
      $(COMNTA(J,2),J=1,6),LOCA(2)
    6  FORMAT(2(I5,5(3A4),5A3,3(2A4),3A4,4A4,6A4,A4))
       IF(CATNOA(1)-NCATNO) 7,8,12
    7  IF(CATNOA(2)-NCATNO) 5,9,12
    8  WRITE(1'N,6)NCATNO,((NMINAM(J,K),J=1,3),K=1,5),
      $(NDONOR(J),J=1,5),(NCUNTR(J),J=1,2),(NSTAT(J),J=1,2),
      $(NCONTY(J),J=1,2),(NTWNA(J),J=1,3),(NMINDI(J),J=1,4),
      $(NCOMNT(J),J=1,6),NLOCA,CATNOA(2),
      $((MINAMA(J,K,2),J=1,3),K=1,5),(DONORA(J,2),J=1,5),
      $(CUNTRA(J,2),J=1,2),(STATA(J,2),J=1,2),
      $(CONTYA(J,2),J=1,2),(TWNA(J,2),J=1,3),(MINDIA(J,2),J=1,4),
      $(COMNTA(J,2),J=1,6),LOCA(2)
       READ(1'N,6)CATNOA(1),((MINAMA(J,K,1),J=1,3),K=1,5),
      $(DONORA(J,1),J=1,5),(CUNTRA(J,1),J=1,2),(STATA(J,1),J=1,2),
      $(CONTYA(J,1),J=1,2),(TWNA(J,1),J=1,3),(MINDIA(J,1),J=1,4),
      $(COMNTA(J,1),J=1,6),LOCA(1),CATNOA(2),
      $((MINAMA(J,K,2),J=1,3),K=1,5),(DONORA(J,2),J=1,5),
      $(CUNTRA(J,2),J=1,2),(STATA(J,2),J=1,2),
      $(CONTYA(J,2),J=1,2),(TWNA(J,2),J=1,3),(MINDIA(J,2),J=1,4),
      $(COMNTA(J,2),J=1,6),LOCA(2)
       WRITE(6,3)CATNOA(1),((MINAMA(J,K,1),J=1,3),K=1,5),
      $(DONORA(J,1),J=1,5)
       WRITE(6,4)(CUNTRA(J,1),J=1,2),(STATA(J,1),J=1,2),
      $(CONTYA(J,1),J=1,2),(TWNA(J,1),J=1,3),(MINDIA(J,1),J=1,4),
      $(COMNTA(J,1),J=1,6),LOCA(1)
       READ(5,1,ERR=11)NCATNO,((NMINAM(J,K),J=1,3),K=1,5),
      $(NDONOR(J),J=1,5)
       READ(5,2)(NCUNTR(J),J=1,2),(NSTAT(J),J=1,2),(NCONTY(J),J=1,2),
      $(NTWNA(J),J=1,3),
      $(NMINDI(J),J=1,4),(NCOMNT(J),J=1,6),NLOCA
       GO TO 7
    9  WRITE(1'N,6)CATNOA(1),((MINAMA(J,K,1),J=1,3),K=1,5),
      $(DONORA(J,1),J=1,5),(CUNTRA(J,1),J=1,2),(STATA(J,1),J=1,2),
      $(CONTYA(J,1),J=1,2),(TWNA(J,1),J=1,3),(MINDIA(J,1),J=1,4),
      $(COMNTA(J,1),J=1,6),LOCA(1),NCATNO,
      $((NMINAM(J,K),J=1,3),K=1,5),(NDONOR(J),J=1,5),
      $(NCUNTR(J),J=1,2),(NSTAT(J),J=1,2),
      $(NCONTY(J),J=1,2),(NTWNA(J),J=1,3),(NMINDI(J),J=1,4),
      $(NCOMNT(J),J=1,6),NLOCA
       READ(1'N,6)CATNOA(1),((MINAMA(J,K,1),J=1,3),K=1,5),
      $(DONORA(J,1),J=1,5),(CUNTRA(J,1),J=1,2),(STATA(J,1),J=1,2),
      $(CONTYA(J,1),J=1,2),(TWNA(J,1),J=1,3),(MINDIA(J,1),J=1,4),
      $(COMNTA(J,1),J=1,6),LOCA(1),CATNOA(2),
      $((MINAMA(J,K,2),J=1,3),K=1,5),(DONORA(J,2),J=1,5),
      $(CUNTRA(J,2),J=1,2),(STATA(J,2),J=1,2),
      $(CONTYA(J,2),J=1,2),(TWNA(J,2),J=1,3),(MINDIA(J,2),J=1,4),
      $(COMNTA(J,2),J=1,6),LOCA(2)
       WRITE(6,3)CATNOA(2),((MINAMA(J,K,2),J=1,3),K=1,5),
      $(DONORA(J,2),J=1,5)
       WRITE(6,4)(CUNTRA(J,2),J=1,2),(STATA(J,2),J=1,2),
      $(CONTYA(J,2),J=1,2),(TWNA(J,2),J=1,3),(MINDIA(J,2),J=1,4),
      $(COMNTA(J,2),J=1,6),LOCA(2)
   10  READ(5,1,ERR=11)NCATNO,((NMINAM(J,K),J=1,3),K=1,5),
```

CORRCT (cont.)

```
      READ(5,2)(NCUNTR(J),J=1,2),(NSTAT(J),J=1,2),(NCONTY(J),J=1,2),
     $(NTWNA(J),J=1,3),
     $(NMINDI(J),J=1,4),(NCOMNT(J),J=1,6),NLOCA
      GO TO 5
   11 PAUSE 'CARD FORMAT ERROR'
      GO TO 10
   12 PAUSE 'FLUSH CARDS AND CHECK ORDER'
      IF(NCATNO.EQ.00000) GO TO 13
      N=N-1
      GO TO 10
   13 STOP
      END
/*
// EXEC LNKEDT
/*
/$ DRC DISK(1,190)=MUS001(SDSBUM),16000
// EXEC
00000

/*
/&
```

UPDATE

```
C**********************TAPE UPDATE*****************************************
//UPDATE EXEC FORTRAN
      INTEGER CATNOA,DONORA(5),CUNTRA(2),STATA(2),CONTYA(2),TWNA(3),
     $COMNTA(6),MINAMA(3,5),MINDIA(4),LOCA,KEND,NMINAM(3,5),NDONOR(5),
     $NCUNTR(2),NSTAT(2),NCONTY(2),NTWNA(3),NCOMNT(6),NMINDI(4),NLOCA,
     $CATNO1
      DATA KBLIK/'    '/
      READ(1,1)KEND
    1 FORMAT(I5)
      KN=0
    2 FORMAT('0',I5,5(3A4),5A3)
    3 FORMAT(' ',3(2A4),3A4,4A4,6A4,A4)
      WRITE(6,4)
    4 FORMAT(51X,'ADDITIONS AND DELETIONS TO THE MUSEUM FILE')
      NKEND=KEND
      DO 13 I=1,99999
      READ(5,5,ERR=9)NCATNO,((NMINAM(J,K),J=1,3),K=1,5),
     $(NDONOR(J),J=1,5)
    5 FORMAT(I5,5(3A4),5A3)
      READ(5,6)(NCUNTR(J),J=1,2),(NSTAT(J),J=1,2),(NCONTY(J),J=1,2),
     $(NTWNA(J),J=1,3),
     $(NMINDI(J),J=1,4),(NCOMNT(J),J=1,6),NLOCA
    6 FORMAT(3(2A4),3A4,4A4,6A4,A4)
      IF(NCATNO.LT.0) GO TO 14
      READ(1,5)CATNOA,((MINAMA(J,K),J=1,3),K=1,5),(DONORA(J),J=1,5)
      READ(1,6)(CUNTRA(J),J=1,2),(STATA(J),J=1,2),(CONTYA(J),J=1,2),
     $(TWNA(J),J=1,3),(MINDIA(J),J=1,4),(COMNTA(J),J=1,6),LOCA
      IF(CATNOA-NCATNO) 12,8,10
    7 CALL BACKUP(1)
      CALL BACKUP(1)
      WRITE(2,5)NCATNO,((NMINAM(J,K),J=1,3),K=1,5),(NDONOR(J),J=1,5)
      WRITE(2,6)(NCUNTR(J),J=1,2),(NSTAT(J),J=1,2),(NCONTY(J),J=1,2),
     $(NTWNA(J),J=1,3),
     $(NMINDI(J),J=1,4),(NCOMNT(J),J=1,6),NLOCA
      KN=KN+1
      NKEND=NKEND+1
      CALL BACKUP(2)
      CALL BACKUP(2)
      WRITE(6,2)CATNOA,((MINAMA(J,K),J=1,3),K=1,5),(DONORA(J),J=1,5)
      WRITE(6,3)(CUNTRA(J),J=1,2),(STATA(J),J=1,2),(CONTYA(J),J=1,2),
     $(TWNA(J),J=1,3),(MINDIA(J),J=1,4),(COMNTA(J),J=1,6),LOCA
      READ(2,5)NCATNO,((NMINAM(J,K),J=1,3),K=1,5),(NDONOR(J),J=1,5)
      READ(2,6)(NCUNTR(J),J=1,2),(NSTAT(J),J=1,2),(NCONTY(J),J=1,2),
     $(NTWNA(J),J=1,3),
     $(NMINDI(J),J=1,4),(NCOMNT(J),J=1,6),NLOCA
      WRITE(6,2)NCATNO,((NMINAM(J,K),J=1,3),K=1,5),(NDONOR(J),J=1,5)
      WRITE(6,3) (NCUNTR(J),J=1,2),(NSTAT(J),J=1,2),(NCONTY(J),J=1,2),
     $(NTWNA(J),J=1,3),
     $(NMINDI(J),J=1,4),(NCOMNT(J),J=1,6),NLOCA
      GO TO 13
    8 IF(NMINAM(1,1).NE.KBLIK) GO TO 7
      NKEND=NKEND-1
      KN=KN-1
      WRITE(6,2)CATNOA,((MINAMA(J,K),J=1,3),K=1,5),(DONORA(J),J=1,5)
      WRITE(6,3)(CUNTRA(J),J=1,2),(STATA(J),J=1,2),(CONTYA(J),J=1,2),
     $(TWNA(J),J=1,3),(MINDIA(J),J=1,4),(COMNTA(J),J=1,6),LOCA
      WRITE(6,2)NCATNO,((NMINAM(J,K),J=1,3),K=1,5),(NDONOR(J),J=1,5)
      WRITE(6,3)(NCUNTR(J),J=1,2),(NSTAT(J),J=1,2),(NCONTY(J),J=1,2),
     $(NTWNA(J),J=1,3),
     $(NMINDI(J),J=1,4),(NCOMNT(J),J=1,6),NLOCA
      GO TO 13
    9 PAUSE 'FLUSH AND CHECK CARD ORDER'
      GO TO 13
   10 CALL BACKUP(1)
      CALL BACKUP(1)
      CALL BACKUP(1)
      CALL BACKUP(1)
      READ(1,11)CATNO1
   11 FORMAT(I5,75X///)
      IF(CATNO1.GT.NCATNO) GO TO 9
      GO TO 8
   12 WRITE(2,5)CATNOA,((MINAMA(J,K),J=1,3),K=1,5),(DONORA(J),J=1,5)
      WRITE(2,6)(CUNTRA(J),J=1,2),(STATA(J),J=1,2),(CONTYA(J),J=1,2),
     $(TWNA(J),J=1,3),(MINDIA(J),J=1,4),(COMNTA(J),J=1,6),LOCA
      KN=KN+1
      READ(1,5)CATNOA,((MINAMA(J,K),J=1,3),K=1,5),(DONORA(J),J=1,5)
      READ(1,6)(CUNTRA(J),J=1,2),(STATA(J),J=1,2),(CONTYA(J),J=1,2),
     $(TWNA(J),J=1,3),(MINDIA(J),J=1,4),(COMNTA(J),J=1,6),LOCA
```

UPDATE (cont.)

```
      IF(CATNOA-NCATNO) 12,8,7
   13 CONTINUE
   14 KN=KN+1
      WRITE(6,15)NKEND
   15 FORMAT('0','THE NEW VALUE OF KEND IS ',I6)
   16 STOP
      END
/*
// EXEC LNKEDT
/$ ORC TAPE(2,182)=DD0128,16000
/$ ORC TAPE(1,180)=DD0129,16000
/*
// EXEC
-1111

/*
/&
```

MNRL

```
C      ************************SINGLE MINERAL SEARCH*****************************
//MNRL  EXEC FORTRAN
      DIMENSION NAME(6,50), MINAM1(3,5), MINAM2(3,5)
      INTEGER*2 CATNO1,CATNO2,LOG(900,50),J(50)
      DEFINE FILE 9(3155,360,E,KDSK)
      DATA KBLNK/'    '/
      READ(5,1) MAXSPL
    1 FORMAT(I6)
      MAXREC=MAXSPL/2
    2 DO 5 KDIT=1,50
      KTIC=KDIT
    3 READ(5,4,ERR=3) (NAME(I,KDIT),I=1,3)
    4 FORMAT(3A4)
      IF(NAME(1,KDIT).EQ.KBLNK) GO TO 6
      ITIC=KDIT
      J(ITIC)=0
    5 CONTINUE
    6 N=0
      IF(NAME(1,1).EQ.KBLNK) GO TO 16
      DO 12 INDEX=1,MAXREC
      N=N+1
      READ(9'N,7) CATNO1,((MINAM1(I,K),I=1,3),K=1,5),
     #CATNO2,((MINAM2(I,K),I=1,3),K=1,5)
    7 FORMAT(I5,5(3A4),15X,80X,I5,5(3A4))
      DO 12 LLG=1,ITIC
      DO 9 K=1,5
      DO 8 I=1,3
      IF(NAME(I,LLG).NE.MINAM1(I,K)) GO TO 9
    8 CONTINUE
      IF(J(LLG).EQ.900) GO TO 12
      J(LLG)=J(LLG)+1
      LOG(J(LLG),LLG)=CATNO1
    9 CONTINUE
      DO 11 K=1,5
      DO 10 I=1,3
      IF(NAME(I,LLG).NE.MINAM2(I,K)) GO TO 11
   10 CONTINUE
      IF(J(LLG).EQ.900) GO TO 12
      J(LLG)=J(LLG)+1
      LOG(J(LLG),LLG)=CATNO2
   11 CONTINUE
   12 CONTINUE
      DO 15 IWRT=1,ITIC
      WRITE(6,13)(NAME(I,IWRT),I=1,3)
   13 FORMAT('-',39X,'RESULTS OF MINERAL SEARCH FOR ',3A4)
      WRITE(2,100)(NAME(I,IWRT),I=1,3),J(IWRT)
  100 FORMAT(3A4,3X,I3)
      JJJJ=J(IWRT)
      WRITE(6,14)(LOG(I,IWRT),I=1,JJJJ)
   14 FORMAT(' ',20I6)
      WRITE(2,200)(LOG(I,IWRT),I=1,JJJJ)
  200 FORMAT(20I6)
   15 CONTINUE
      IF(NAME(1,KTIC).EQ.KBLNK) GO TO 16
      GO TO 2
   16 STOP
      END
/*
// EXEC LNKEDT
/*
/$ DRC PRINT,16000
/$ DRC DISK(9,190)=MUS001(SDSBUM),32000
/$ DRC TAPE(2,181)=MUS002,16000
// EXEC
  5274
C                          FOLLOWING CARDS  MINERALS TO BE SEARCHED FOR

/*
/&
```

DONOR

```
C      ************************DONOR SEARCH*****************************
//DONOR EXEC FORTRAN
       DIMENSION NAME(6,50)
       INTEGER DNR1(5), DNR2(5)
       INTEGER*2 CATNO1,CATNO2,LOG(900,50),J(50)
       DEFINE FILE 9(3155,360,E,KDSK)
       DATA KBLNK/'    '/
       READ(5,1) MAXSPL
     1 FORMAT(I6)
       MAXREC=MAXSPL/2
     2 DO 5 KDIT=1,50
       KTIC=KDIT
     3 READ(5,4,ERR=3) (NAME(I,KDIT),I=1,5)
     4 FORMAT(5A3)
       IF(NAME(1,KDIT).EQ.KBLNK) GO TO 6
       ITIC=KDIT
       J(ITIC)=0
     5 CONTINUE
     6 N=0
       IF(NAME(1,1).EQ.KBLNK) GO TO 16
       DO 12 INDEX=1,MAXREC
       N=N+1
       READ(9'N,7)CATNO1,(DNR1(I),I=1,5),CATNO2,(DNR2(I),I=1,5)
     7 FORMAT(I5,60X,5A3,80X,I5,60X,5A3)
       DO 12 LLG=1,ITIC
       DO 8 I=1,5
       IF(NAME(I,LLG).NE.DNR1(I)) GO TO 9
     8 CONTINUE
       IF(J(LLG).EQ.900) GO TO 12
       J(LLG)=J(LLG)+1
       LOG(J(LLG),LLG)=CATNO1
     9 CONTINUE
       DO 10 I=1,5
       IF(NAME(I,LLG).NE.DNR2(I)) GO TO 11
    10 CONTINUE
       IF(J(LLG).EQ.900) GO TO 12
       J(LLG)=J(LLG)+1
       LOG(J(LLG),LLG)=CATNO2
    11 CONTINUE
    12 CONTINUE
       DO 15 IWRT=1,ITIC
       WRITE(6,13)(NAME(I,IWRT),I=1,5)
    13 FORMAT('-',39X,'RESULTS OF DONOR SEARCH FOR ',5A3)
       JJJJ=J(IWRT)
       WRITE(6,14)(LOG(I,IWRT),I=1,JJJJ)
    14 FORMAT(' ',2016)
    15 CONTINUE
       IF(NAME(1,KTIC).EQ.KBLNK) GO TO 16
       GO TO 2
    16 STOP
       END
/*
// EXEC LNKEDT
/*
/$ ORG DISK(9,190)=MUSOO1(SDSBUM),16000
/$ ORG PRINT,16000
// EXEC
  5274

/*
/&
```

CONTRA

```
C      ************************COUNTRY SEARCH*************************.
//CONTRA EXEC FORTRAN
       DIMENSION NAME(6,50),KNTRY1(2),KNTRY2(2)
       INTEGER*2 CATNO1,CATNO2,LOG(900,50),J(50)
       DEFINE FILE 9(3155,360,E,KDSK)
       DATA KBLNK/'    '/
       READ(5,1) MAXSPL
     1 FORMAT(I6)
       MAXREC=MAXSPL/2
     2 DO 5 KDIT=1,50
       KTIC=KDIT
     3 READ(5,4,ERR=3) (NAME(I,KDIT),I=1,2)
     4 FORMAT(2A4)
       IF(NAME(1,KDIT).EQ.KBLNK) GO TO 6
       ITIC=KDIT
       J(ITIC)=0
     5 CONTINUE
     6 N=0
       IF(NAME(1,1).EQ.KBLNK) GO TO 16
       DO 12 INDEX=1,MAXREC
       N=N+1
       READ(9'N,7)CATNO1,(KNTRY1(I),I=1,2),CATNO2,(KNTRY2(I),I=1,2)
     7 FORMAT(I5,75X,2A4,72X,I5,75X,2A4)
       DO 12 LLG=1,ITIC
       DO 8 I=1,2
       IF(NAME(I,LLG).NE.KNTRY1(I)) GO TO 9
     8 CONTINUE
       IF(J(LLG).EQ.900) GO TO 12
       J(LLG)=J(LLG)+1
       LOG(J(LLG),LLG)=CATNO1
     9 CONTINUE
       DO 10 I=1,2
       IF(NAME(I,LLG).NE.KNTRY2(I)) GO TO 11
    10 CONTINUE
       IF(J(LLG).EQ.900) GO TO 12
       J(LLG)=J(LLG)+1
       LOG(J(LLG),LLG)=CATNO2
    11 CONTINUE
    12 CONTINUE
       DO 15 IWRT=1,ITIC
       WRITE(6,13)(NAME(I,IWRT),I=1,2)
    13 FORMAT('-',41X,'RESULTS OF COUNTRY SEARCH FOR ',2A4)
       JJJJ=J(IWRT)
       WRITE(6,14)(LOG(I,IWRT),I=1,JJJJ)
    14 FORMAT(' ',2016)
    15 CONTINUE
       IF(NAME(1,KTIC).EQ.KBLNK) GO TO 16
       GO TO 2
    16 STOP
       END
/*
// EXEC LNKEDT
/*
/$ DRC PRINT,16000
/$ DRC DISK(9,190)=MUSOO1(SDSBUM),16000
// EXEC
  5274

/*
/&
```

```
C      ************************STATE SEARCH*****************************
//STATE EXEC FORTRAN
      DIMENSION NAME(6,50)
      INTEGER STAT1(2),STAT2(2)
      INTEGER*2 CATNO1,CATNO2,LOG(900,50),J(50)
      DEFINE FILE 9(3155,360,E,KDSK)
      DATA KBLNK/'    '/
      READ(5,1) MAXSPL
    1 FORMAT(I6)
      MAXREC=MAXSPL/2
    2 DO 5 KDIT=1,50
      KTIC=KDIT
    3 READ(5,4,ERR=3) (NAME(I,KDIT),I=1,2)
    4 FORMAT(2A4)
      IF(NAME(1,KDIT).EQ.KBLNK) GO TO 6
      ITIC=KDIT
      J(ITIC)=0
    5 CONTINUE
    6 N=0
      IF(NAME(1,1).EQ.KBLNK) GO TO 16
      DO 12 INDEX=1,MAXREC
      N=N+1
      READ(9'N,7)CATNO1,(STAT1(I),I=1,2),CATNO2,(STAT2(I),I=1,2)
    7 FORMAT(I5,75X,8X,2A4,64X,I5,75X,8X,2A4)
      DO 12 LLG=1,ITIC
      DO 8 I=1,2
      IF(NAME(I,LLG).NE.STAT1(I)) GO TO 9
    8 CONTINUE
      IF(J(LLG).EQ.900) GO TO 12
      J(LLG)=J(LLG)+1
      LOG(J(LLG),LLG)=CATNO1
    9 CONTINUE
      DO 10 I=1,2
      IF(NAME(I,LLG).NE.STAT2(I)) GO TO 11
   10 CONTINUE
      IF(J(LLG).EQ.900) GO TO 12
      J(LLG)=J(LLG)+1
      LOG(J(LLG),LLG)=CATNO2
   11 CONTINUE
   12 CONTINUE
      DO 15 IWRT=1,ITIC
      WRITE(6,13)(NAME(I,IWRT),I=1,2)
   13 FORMAT('-',42X,'RESULTS OF STATE SEARCH FOR ',2A4)
      JJJJ=J(IWRT)
      WRITE(6,14)(LOG(I,IWRT),I=1,JJJJ)
   14 FORMAT(' ',2016)
   15 CONTINUE
      IF(NAME(1,KTIC).EQ.KBLNK) GO TO 16
      GO TO 2
   16 STOP
      END
/*
// EXEC LNKEDT
/*
/$ DRC PRINT,16000
/$ DRC DISK(9,190)=MUS001(SDSBUM),16000
// EXEC
  5274

/*
/&
```

COUNTY

```
C      ************************COUNTY SEARCH*************************
//COUNTY EXEC FORTRAN
       DIMENSION NAME(6,50)
       INTEGER CNTY1(2),CNTY2(2)
       INTEGER*2 CATNO1,CATNO2,LOG(900,50),J(50)
       DEFINE FILE 9(3155,360,E,KDSK)
       DATA KBLNK/'    '/
       READ(5,1) MAXSPL
     1 FORMAT(I6)
       MAXREC=MAXSPL/2
     2 DO 5 KDIT=1,50
       KTIC=KDIT
     3 READ(5,4,ERR=3) (NAME(I,KDIT),I=1,2)
     4 FORMAT(2A4)
       IF(NAME(1,KDIT).EQ.KBLNK) GO TO 6
       ITIC=KDIT
       J(ITIC)=0
     5 CONTINUE
     6 N=0
       IF(NAME(1,1).EQ.KBLNK) GO TO 16
       DO 12 INDEX=1,MAXREC
       N=N+1
       READ(9'N,7)CATNO1,(CNTY1(I),I=1,2),CATNO2,(CNTY2(I),I=1,2)
     7 FORMAT(I5,75X,16X,2A4,56X,I5,75X,16X,2A4)
       DO 12 LLG=1,ITIC
       DO 8 I=1,2
       IF(NAME(I,LLG).NE.CNTY1(I)) GO TO 9
     8 CONTINUE
       IF(J(LLG).EQ.900) GO TO 12
       J(LLG)=J(LLG)+1
       LOG(J(LLG),LLG)=CATNO1
     9 CONTINUE
       DO 10 I=1,2
       IF(NAME(I,LLG).NE.CNTY2(I)) GO TO 11
    10 CONTINUE
       IF(J(LLG).EQ.900) GO TO 12
       J(LLG)=J(LLG)+1
       LOG(J(LLG),LLG)=CATNO2
    11 CONTINUE
    12 CONTINUE
       DO 15 IWRT=1,ITIC
       WRITE(6,13)(NAME(I,IWRT),I=1,2)
    13 FORMAT('-',41X,'RESULTS OF COUNTY SEARCH FOR ',2A4)
       JJJJ=J(IWRT)
       WRITE(6,14)(LOG(I,IWRT),I=1,JJJJ)
    14 FORMAT(' ',20I6)
    15 CONTINUE
       IF(NAME(1,KTIC).EQ.KBLNK) GO TO 16
       GO TO 2
    16 STOP
       END
/*
// EXEC LNKEDT
/*
/$ ORG PRINT,16000
/$ ORG DISK(9,190)=MUS001(SDSBUM),16000
//  EXEC
  5274

/*
/&
```

```
C      ************************TOWN SEARCH*********************
//TOWN EXEC FORTRAN
      DIMENSION NAME(6,50)
      INTEGER TWN1(3),TWN2(3)
      INTEGER*2 CATNO1,CATNO2,LOG(900,50),J(50)
      DEFINE FILE 9(3155,360,E,KDSK)
      DATA KBLNK/'    '/
      READ(5,1) MAXSPL
    1 FORMAT(I6)
      MAXREC=MAXSPL/2
    2 DO 5 KDIT=1,50
      KTIC=KDIT
    3 READ(5,4,ERR=3) (NAME(I,KDIT),I=1,3)
    4 FORMAT(3A4)
      IF(NAME(1,KDIT).EQ.KBLNK) GO TO 6
      ITIC=KDIT
      J(ITIC)=0
    5 CONTINUE
    6 N=0
      IF(NAME(1,1).EQ.KBLNK) GO TO 16
      DO 12 INDEX=1,MAXREC
      N=N+1
      READ(9'N,7)CATNO1,(TWN1(I),I=1,3),CATNO2,(TWN2(I),I=1,3)
    7 FORMAT(I5,75X,24X,3A4,44X,I5,75X,24X,3A4)
      DO 12 LLG=1,ITIC
      DO 8 I=1,3
      IF(NAME(I,LLG).NE.TWN1(I)) GO TO 9
    8 CONTINUE
      IF(J(LLG).EQ.900) GO TO 12
      J(LLG)=J(LLG)+1
      LOG(J(LLG),LLG)=CATNO1
    9 CONTINUE
      DO 10 I=1,3
      IF(NAME(I,LLG).NE.TWN2(I)) GO TO 11
   10 CONTINUE
      IF(J(LLG).EQ.900) GO TO 12
      J(LLG)=J(LLG)+1
      LOG(J(LLG),LLG)=CATNO2
   11 CONTINUE
   12 CONTINUE
      DO 15 IWRT=1,ITIC
      WRITE(6,13)(NAME(I,IWRT),I=1,3)
   13 FORMAT('-',41X,'RESULTS OF TOWN SEARCH FOR ',3A4)
      JJJJ=J(IWRT)
      WRITE(6,14)(LOG(I,IWRT),I=1,JJJJ)
   14 FORMAT(' ',20I6)
   15 CONTINUE
      IF(NAME(1,KTIC).EQ.KBLNK) GO TO 16
      GO TO 2
   16 STOP
      END
/*
// EXEC LNKEDT
/*
/$ DRC PRINT,16000
/$ DRC DISK(9,190)=MUS001(SDSBUM),16000
// EXEC
  5274

/*
/&
```

MINDST

```
C      **********************MINING DISTRICT SEARCH*********************
//MINDST EXEC FORTRAN
       DIMENSION NAME(6,50),MDST1(4),MDST2(4)
       INTEGER*2 CATNO1,CATNO2,LOG(900,50),J(50)
       DEFINE FILE 9(3155,360,E,KDSK)
       DATA KBLNK/'    '/
       READ(5,1) MAXSPL
     1 FORMAT(I6)
       MAXREC=MAXSPL/2
     2 DO 5 KDIT=1,5
       KTIC=KDIT
     3 READ(5,4,ERR=3)(NAME(I,ITIC),I=1,4)
     4 FORMAT(4A4)
       IF(NAME(1,KDIT).EQ.KBLNK) GO TO 6
       ITIC=KDIT
       J(ITIC)=0
     5 CONTINUE
     6 N=0
       IF(NAME(1,1).EQ.KBLNK) GO TO 16
       DO 12 INDEX=1,MAXREC
       N=N+1
       READ(9'N,7)CATNO1,(MDST1(I),I=1,4),CATNO2,(MDST2(I),I=1,4)
     7 FORMAT(I5,75X,36X,4A4,28X,I5,75X,36X,4A4)
       DO 12 LLG=1,ITIC
       DO 8 I=1,4
       IF(NAME(I,LLG).NE.MDST1(I)) GO TO 9
     8 CONTINUE
       IF(J(LLG).EQ.900) GO TO 12
       J(LLG)=J(LLG)+1
       LOG(J(LLG),LLG)=CATNO1
     9 CONTINUE
       DO 10 I=1,4
       IF(NAME(I,LLG).NE.MDST2(I)) GO TO 11
    10 CONTINUE
       IF(J(LLG).EQ.900) GO TO 12
       J(LLG)=J(LLG)+1
       LOG(J(LLG),LLG)=CATNO2
    11 CONTINUE
    12 CONTINUE
       DO 15 IWRT=1,ITIC
       WRITE(6,13)(NAME(I,IWRT),I=1,4)
    13 FORMAT('-',33X,'RESULTS OF MINING DISTRICT SEARCH FOR ',4A4)
       JJJJ=J(IWRT)
       WRITE(6,14)(LOG(I,IWRT),I=1,JJJJ)
    14 FORMAT(' ',2016)
    15 CONTINUE
       IF(NAME(1,KTIC).EQ.KBLNK) GO TO 16
       GO TO 2
    16 STOP
       END
/*
// EXEC LNKEDT
/*
/$ DRC PRINT,16000
/$ DRC DISK(9,190)=MUS001(SDSBUM),16000
// EXEC
  5274
TYRONE
/*
/&
```

STLOC

```
C      ************************LOCATION SEARCH**********************
//STLOC EXEC FORTRAN
       DIMENSION NAME(6,50)
       INTEGER*2 CATNO1,CATNO2,LOG(900,50),J(50)
       DEFINE FILE 9(3155,360,E,KDSK)
       DATA KBLNK/'    '/
       READ(5,1) MAXSPL
     1 FORMAT(I6)
       MAXREC=MAXSPL/2
     2 DO 5 KDIT=1,50
       KTIC=KDIT
     3 READ(5,4,ERR=3) (NAME(1,KDIT))
     4 FORMAT(A4)
       IF(NAME(1,KDIT).EQ.KBLNK) GO TO 6
       ITIC=KDIT
       J(ITIC)=0
     5 CONTINUE
     6 N=0
       IF(NAME(1,1).EQ.KBLNK) GO TO 13
       DO 9 INDEX=1,MAXREC
       N=N+1
       READ(9'N,7)CATNO1,LOC1,CATNO2,LOC2
     7 FORMAT(I5,75X,76X,A4,I5,75X,76X,A4)
       DO 9 LLG=1,ITIC
       IF(NAME(1,LLG).NE.LOC1) GO TO 8
       IF(J(LLG).EQ.900) GO TO 9
       J(LLG)=J(LLG)+1
       LOG(J(LLG),LLG)=CATNO1
     8 IF(NAME(1,LLG).NE.LOC2) GO TO 9
       IF(J(LLG).EQ.900) GO TO 9
       J(LLG)=J(LLG)+1
       LOG(J(LLG),LLG)=CATNO2
     9 CONTINUE
       DO 12 IWRT=1,ITIC
       WRITE(6,10)NAME(1,IWRT)
    10 FORMAT('-',33X,'RESULTS OF STORAGE LOCATION SEARCH FOR ',A4)
       JJJJ=J(IWRT)
       WRITE(6,11)(LOG(I,IWRT),I=1,JJJJ)
    11 FORMAT(' ',2016)
    12 CONTINUE
       IF(NAME(1,KTIC).EQ.KBLNK) GO TO 13
       GO TO 2
    13 STOP
       END
/*
// EXEC LNKEDT
/*
/$ DRC PRINT,16000
/$ DRC DISK(9,190)=MUS001(SDSBUM),16000
// EXEC
  5274
WO2
WO3
WO4
WO5

/*
/&
```

```
C*******************N M BUREAU MUSEUM MULTI SEARCH PROGRAM*************
      DIMENSION LCMT(50,6),MATCH(5)
      INTEGER DRN,HL,JL
      INTEGER CATNO(2),DONOR(2,5),CUNTRA(2,2),STATE(2,2),CONTY(2,2),
     #     TOWN(2,3),SEARCH(16)
      INTEGER DATA(20)
      DIMENSION KODE(5),KDATA(50),KIX(50)
      DIMENSION MINAM(2,5,3),KOMNT(2,6),MINDT(2,4),NAME(16,6),LOOKUP(16
     #     ,LREFX(2)
      INTEGER STLOC(2)
      DIMENSION LCT(50),LMIN(50,5,3),LDON(50,5),LKTA(50,2),LSTT(50,2),
     #     LCTA(50,2),LTWN(50,3),LMDT(50,4),LCM(50,6),LSLC(50)
      DEFINE FILE 9(3155,360,E,KDSK)
      DATA KBLANK/'    '/
      ID=0
      JUST=1
      DRN=0
      READ(5,1)MAXSPL
    1 FORMAT(I6)
      MAXREC=MAXSPL/2
      READ(5,2)NUMSER,(SEARCH(I),I=1,14)
    2 FORMAT(I2,14I1)
      DO 3 K=1,14
      IF(SEARCH(K).NE.1) GO TO 3
      ID=ID+1
      LOOKUP(ID)=K
    3 CONTINUE
      DO 5 J=1,NUMSER
      IF(LOOKUP(J).EQ.7.OR.LOOKUP(J).EQ.14) GO TO 60
      READ(5,4)(NAME(J,I),I=1,6)
    4 FORMAT(6A4)
      GO TO 5
   60 IF(SEARCH(J).EQ.0) GO TO 5
      READ(5,61)(NAME(J,I),I=1,5)
   61 FORMAT(5A3)
    5 CONTINUE
      READ(5,70)(MATCH(I),I=1,5)
   70 FORMAT(5A1)
      DO 30 N=1,MAXREC
      READ(9'N,6)CATNO(1),
     #          ((MINAM(1,I,K),K=1,3),I=1,5),(DONOR(1,I),I=1,5),
     #     (CUNTRA(1,I),I=1,2),(STATE(1,I),I=1,2),(CONTY(1,I),I=1,2),
     #     (TOWN(1,I),I=1,3),(MINDT(1,I),I=1,4),(KOMNT(1,I),I=1,6),
     #     STLOC(1),LREFX(1),
     #          CATNO(2),
     #          ((MINAM(2,I,K),K=1,3),I=1,5),(DONOR(2,I),I=1,5),
     #     (CUNTRA(2,I),I=1,2),(STATE(2,I),I=1,2),(CONTY(2,I),I=1,2),
     #     (TOWN(2,I),I=1,3),(MINDT(2,I),I=1,4),(KOMNT(2,I),I=1,6),
     #     STLOC(2),LREFX(2)
    6 FORMAT(2(I5,5(3A4),5A3,3(2A4),3A4,4A4,6A4,A3,A1))
      DO 30 II=1,2
      DO 29 M=1,NUMSER
      LOTS=LOOKUP(M)
      GO TO (11,12,12,12,12,12,17,18,19,20,21,22,23,24),LOTS
C
CC    CATALOG NUMBER SEARCH
C
   11 IF(NAME(M,1).EQ.CATNO(II)) GO TO 29
      GO TO 30
C
CC    MINERAL SEARCH FOR ONE TO FIVE MINERALS
C
   12 DO 112 JI=1,5
      DO 111 LI=1,3
      IF(MINAM(II,JI,LI).NE.NAME(M,LI)) GO TO 113
  111 CONTINUE
      GO TO 29
  113 IF(JI.EQ.5) GO TO 30
  112 CONTINUE
C
CC    DONOR SEARCH
C
   17 DO 117 JI=1,5
      IF(DONOR(II,JI).NE.NAME(M,JI)) GO TO 30
  117 CONTINUE
      GO TO 29
C
C     COUNTRY SEARCH
```

MULSER (cont.)

```
C
   18 DO 118 LI=1,2
      IF(CUNTRA(II,LI).NE.NAME(M,LI)) GO TO 30
  118 CONTINUE
      GO TO 29
CC
CC     STATE SEARCH
C
   19 DO 119 LI=1,2
      IF(STATE(II,LI).NE.NAME(M,LI)) GO TO 30
  119 CONTINUE
      GO TO 29
CC
CC     COUNTY SEARCH
C
   20 DO 200 LI=1,2
      IF(CONTY(II,LI).NE.NAME(M,LI)) GO TO 30
  200 CONTINUE
      GO TO 29
CC
CC     TOWN SEARCH
C
   21 DO 210 LI=1,3
      IF(TOWN(II,LI).NE.NAME(M,LI)) GO TO 30
  210 CONTINUE
      GO TO 29
CC
CC    MINING DISTRICT SEARCH
C
   22 DO 220 LI=1,4
      IF(MINDT(II,LI).NE.NAME(M,LI)) GO TO 30
  220 CONTINUE
      GO TO 29
CC
CC     COMMENT SEARCH
C
   23 DO 230 LI=1,6
      IF(KOMNT(II,LI).NE.NAME(M,LI)) GO TO 30
  230 CONTINUE
      GO TO 29
C
CC     STORAGE LOCATION SEARCH
C
   24 IF(STLOC(II).NE.NAME(M,1)) GO TO 30
   29 CONTINUE
      LCT(JUST)=CATNO(II)
      DO 26 HL=1,5
      DO 26 JL=1,3
      LMIN(JUST,HL,JL)=MINAM(II,HL,JL)
   26 CONTINUE
      DO 27 HL=1,5
      LDON(JUST,HL)=DONOR(II,HL)
   27 CONTINUE
      DO 28 HL=1,2
      LKTA(JUST,HL)=CUNTRA(II,HL)
      LSTT(JUST,HL)=STATE(II,HL)
      LCTA(JUST,HL)=CONTY(II,HL)
   28 CONTINUE
      DO 329 HL=1,3
      LTWN(JUST,HL)=TOWN(II,HL)
  329 CONTINUE
      DO 31 HL=1,4
      LMDT(JUST,HL)=MINDT(II,HL)
   31 CONTINUE
      DO 32 HL=1,6
      LCMT(JUST,HL)=KOMNT(II,HL)
   32 CONTINUE
      LSLC(JUST)=STLOC(II)
      JUST=JUST+1
      JMIN=JUST-1
      IF(LREFX(II).EQ.KBLANK) GO TO 30
      IF(II.EQ.2) GO TO 72
      READ(9'N,71)(KODE(K),K=1,5)
   71 FORMAT(80X,71X,5A1)
      GO TO 74
   72 READ(9'N,73)(KODE(K),K=1,5)
   73 FORMAT(160X,80X,71X,5A1)
   74 DO 75 K=1,5
```

MULSER (cont.)

```
      DO 75 KK=1,5
      IF(KODE(K).NE.MATCH(KK)) GO TO 75
      DRN=DRN+1
      KIX(DRN)=KODE(K)
      KDATA(DRN)=LCT(JMIN)
   75 CONTINUE
   30 CONTINUE
      WRITE(6,40)
   40 FORMAT('1','SEARCH OF CATALOG FOR')
      DO 45 JX=1,NUMSER
      WRITE(6,41)(NAME(JX,I),I=1,6)
   41 FORMAT('0',20X,6A4)
   45 CONTINUE
      WRITE(6,800)
  800 FORMAT('0','**************************',/)
      DO 52 I=1,JMIN
      WRITE(6,50)LCT(I),((LMIN(I,M,N),N=1,3),M=1,5),(LDON(I,M),M=1,5)
      WRITE(6,51)(LKTA(I,M),M=1,2),(LSTT(I,M),M=1,2),(LCTA(I,M),M=1,2),
     #     (LTWN(I,M),M=1,3),(LMDT(I,M),M=1,4),(LCMT(I,M),M=1,6),
     #     LSLC(I)
   50 FORMAT('0',I5,2X,5(3A4,2X),5A3)
   51 FORMAT(' ',5X,3(2A4,2X),3A4,2X,4A4,2X,6A4,2X,A4)
   52 CONTINUE
      WRITE(6,801)
  801 FORMAT('0','EXTRA DATA',/)
      KRN=DRN
      READ(5,80)KENT
   80 FORMAT(I5)
      DO 91 KNX=1,KENT
       READ(1,81)KIX1,KDATA1
   81 FORMAT(A1,I5)
      DO 90 KLN=1,DRN
      IF(KRN.LE.0) GO TO 99
      IF(KIX1.NE.KIX(KLN)) GO TO 96
      IF(KDATA1.NE.KDATA(KLN)) GO TO 96
      KRN=KRN-1
      READ(1,82)NUM
   82 FORMAT(I2)
      WRITE(6,802)KIX1,KDATA1
  802 FORMAT('0','SEARCH TYPE IS ',A1,'   CATALOG NO. ',I5)
      DO 85 KNUM=1,NUM
      READ(1,83)(DATA(I),I=1,20)
   83 FORMAT(20A4)
      WRITE(6,84)(DATA(I),I=1,20)
   84 FORMAT(' ',20A4)
   85 CONTINUE
      GO TO 91
   96 READ(1,82)NUM
      DO 98 KDX=1,NUM
      READ(1,97)
   97 FORMAT(80X)
   98 CONTINUE
   90 CONTINUE
   91 CONTINUE
   99 STOP
      END
```

Example of MULSER Output

```
SEARCH OF CATALOG FOR
                    CERUSSITE
                    GALENA
                    USA
                    NW MXICO

  98  GALENA        ANGLESITE     CERUSSITE
      USA           NW MXICO  SOCORRO   KELLY          MAGDALENA        KELLY MINE BROWN C T                    B21

  98  GALENA        ANGLESITE     CERUSSITE
      USA           NW MXICO  SOCORRO   KELLY          MAGDALENA        KELLY MINE BROWN C T                    SD6

 100  GALENA        CERUSSITE
      USA           NW MXICO  SOCORRO   KELLY          MAGDALENA        KELLY MINE BROWN C T                    B08

 103  GALENA        ANGLESITE     CERUSSITE
      USA           NW MXICO  SOCORRO   KELLY          MAGDALENA        KELLY MINE BROWN C T                    SD4

 103  GALENA        ANGLESITE     CERUSSITE
      USA           NW MXICO  SOCORRO   KELLY          MAGDALENA        KELLY MINE BROWN C T                    SD5

3563  GALENA        CERUSSITE
      USA           NW MXICO  SOCORRO   KELLY          MAGDALENA        ALTERING TO CERUSSITE                   B09

3581  GALENA        CERUSSITE
      USA           NW MXICO  SOCORRO                  HANSONBURG       OSCURA MTNS/ALT CERRUSIT NEEDHAM C E    B09

3581  GALENA        CERUSSITE
      USA           NW MXICO  SOCORRO                  HANSONBURG       OSCURA MTNS/ALT CERUSSIT NEEDHAM C E    SD4

3670  PYROMORPHITE  CERUSSITE      GALENA              CRYSOCOLLA
      USA           NW MXICO  GRANT                    HANOVER-FIERRO                                           AS4

4494  CERARGYRITE   PYRARGYRITE   CERUSSITE     GALENA
      USA           NW MXICO  SIERRA   LAKE VALLEY    SALVAGE WORKINGS MILLER D M                             HA2

5759  CERUSSITE     GALENA
      USA           NW MXICO  DONA ANA                ORGAN            XLS                                     AS4

5772  PYROMORPHITE  MALACHITE     BARITE        CERUSSITE     GALENA
      USA           NW MXICO  SIERRA                                   ARGENTITE, CHLORITE                     A03
```

Example of EXTRA DATA Output

```
SEARCH OF CATALOG FOR
                    CELESTITE
                    USA
                    OHIO
**************************

  820  CELESTITE                                                    BROWN C T
     USA          OHIO                  PUT-IN-BAY                                    AS3
  827  CELESTITE                                                    BROWN C T
     USA          OHIO                  PUT-IN BAY                                    A14
 6566  CELESTITE
     USA          OHIO                  PUT-IN-BAY                                    A14
 6567  CELESTITE                                                    GUNNELL E M
     USA          OHIO                  CLAY CENTER                        A          DB
 7836  FLUORITE       CELESTITE
     USA          OHIO                  CLAY CENTER                                   AW1
EXTRA DATA

SEARCH TYPE IS A   CATALOG NO.  6567
MINERAL NAME- CELESTITE
COMMENTS- CLAY CENTER, OHIO
X-RAY CONDITIONS
KV. 40          TARGET- CU          MA. 20          FILTER NI        MONOCHROM. N7A
DETECTOR- CAMERA #2, EIGHT HOURS
DATE  5-13-67
HKL       D          D(ASTM)              HKL       D         D(ASTM)
        OBSERVED     #5-0593
011       4.20       4.23                 230       1.639     1.640
111       3.76       3.77                 501                 1.625
201                  3.57                 223,114             1.604
002       3.44       3.433                522       1.600     1.601
210       3.296      3.295                231                 1.569
102       3.178      3.177                132,511   1.554     1.555
211       2.969      2.972                214       1.522     1.521
112       2.732      2.731                323       1.475     1.475
020       2.669      2.674                421       1.447
301       2.582      2.582                024       1.445     1.444
121                  2.388                124       1.424     1.424
212       2.378      2.377
220       2.250      2.253
103       2.206      2.208
302       2.163      2.164
221       2.138      2.141
122       2.042      2.045
113                  2.041
203                  2.006
401       1.9997     1.999
410       1.945      1.947
321       1.855      1.857
303       1.768      1.769
031                  1.728
004       1.719      1.715
412,131              1.691
313       1.680      1.679
```