UNITED STATES
PARTMENT OF
OMMERCE
UBLICATION

NBS TECHNICAL NOTE 700

COMBO:
A General-Purpose Program for Searching, Annotating, Encoding-Decoding, and Reformatting Data Files

NATIONAL BUREAU OF STANDARDS

The National Bureau of Standards[1] was established by an act of Congress March 3, 1901. The Bureau's overall goal is to strengthen and advance the Nation's science and technology and facilitate their effective application for public benefit. To this end, the Bureau conducts research and provides: (1) a basis for the Nation's physical measurement system, (2) scientific and technological services for industry and government, (3) a technical basis for equity in trade, and (4) technical services to promote public safety. The Bureau consists of the Institute for Basic Standards, the Institute for Materials Research, the Institute for Applied Technology, the Center for Computer Sciences and Technology, and the Office for Information Programs.

THE INSTITUTE FOR BASIC STANDARDS provides the central basis within the United States of a complete and consistent system of physical measurement; coordinates that system with measurement systems of other nations; and furnishes essential services leading to accurate and uniform physical measurements throughout the Nation's scientific community, industry, and commerce. The Institute consists of a Center for Radiation Research, an Office of Measurement Services and the following divisions:

Applied Mathematics—Electricity—Heat—Mechanics—Optical Physics—Linac Radiation[2]—Nuclear Radiation[2]—Applied Radiation[2]—Quantum Electronics[3]—Electromagnetics[3]—Time and Frequency[3]—Laboratory Astrophysics[3]—Cryogenics[3].

THE INSTITUTE FOR MATERIALS RESEARCH conducts materials research leading to improved methods of measurement, standards, and data on the properties of well-characterized materials needed by industry, commerce, educational institutions, and Government; provides advisory and research services to other Government agencies; and develops, produces, and distributes standard reference materials. The Institute consists of the Office of Standard Reference Materials and the following divisions:

Analytical Chemistry—Polymers—Metallurgy—Inorganic Materials—Reactor Radiation—Physical Chemistry.

THE INSTITUTE FOR APPLIED TECHNOLOGY provides technical services to promote the use of available technology and to facilitate technological innovation in industry and Government; cooperates with public and private organizations leading to the development of technological standards (including mandatory safety standards), codes and methods of test; and provides technical advice and services to Government agencies upon request. The Institute also monitors NBS engineering standards activities and provides liaison between NBS and national and international engineering standards bodies. The Institute consists of a Center for Building Technology and the following divisions and offices:

Engineering Standards Services—Weights and Measures—Invention and Innovation—Product Evaluation Technology—Electronic Technology—Technical Analysis—Measurement Engineering—Fire Technology—Housing Technology[4]—Federal Building Technology[4]—Building Standards and Codes Services[4]—Building Environment[4]—Structures, Materials and Life Safety[4]—Technical Evaluation and Application[4].

THE CENTER FOR COMPUTER SCIENCES AND TECHNOLOGY conducts research and provides technical services designed to aid Government agencies in improving cost effectiveness in the conduct of their programs through the selection, acquisition, and effective utilization of automatic data processing equipment; and serves as the principal focus within the executive branch for the development of Federal standards for automatic data processing equipment, techniques, and computer languages. The Center consists of the following offices and divisions:

Information Processing Standards—Computer Information—Computer Services—Systems Development—Information Processing Technology.

THE OFFICE FOR INFORMATION PROGRAMS promotes optimum dissemination and accessibility of scientific information generated within NBS and other agencies of the Federal Government; promotes the development of the National Standard Reference Data System and a system of information analysis centers dealing with the broader aspects of the National Measurement System; provides appropriate services to ensure that the NBS staff has optimum accessibility to the scientific information of the world, and directs the public information activities of the Bureau. The Office consists of the following organizational units:

Office of Standard Reference Data—Office of Technical Information and Publications—Library—Office of International Relations.

---

[1] Headquarters and Laboratories at Gaithersburg, Maryland, unless otherwise noted; mailing address Washington, D.C. 20234.

[2] Part of the Center for Radiation Research.

COMBO: A General-Purpose Program For Searching, Annotating, Encoding-Decoding, and Reformatting Data Files

Robert McClenon and Joseph Hilsenrath

Data Systems Design Group
Office of Standard Reference Data
National Bureau of Standards
Washington, D.C. 20234

Issued September 1972

National Bureau of Standards Technical Note 700

Nat. Bur. Stand. (U.S.), Tech. Note 700, 68 pages (Sept. 1972)

CODEN: NBTNAE

For sale by the Superintendent of Documents, U.S. Government Printing Office
Washington, D.C. 20402 - Price 75 cents

Stock Number 0303–0921

CONTENTS

Appendices

COMBO: A General-Purpose Program for Searching, Annotating, Encoding-Decoding and Reformatting Data Files

by

Robert McClenon and Joseph Hilsenrath

COMBO, a FORTRAN program for searching magnetic tape files, generating reports, and reformatting the file, is described and listed. The program is capable of reading separate card images from a file blocked in physical records and recognizing logical blocks marked by a fixed-field ID. Up to 99 different types of lines, each with its own format, can be recognized by examining a special code or label. The program can be instructed to search for the occurrence of each of certain character strings, using a different list for each type of line and two levels of Boolean logic. Lines can be broken into pieces, using either a fixed-field format or a single separator or flags to define the pieces, and the pieces can be rearranged, with labels or comments optionally inserted between them. Editing, in which specified strings are replaced by other strings, can also be performed. The program can accommodate a variable number of cards of each type per block. It was assembled from general-purpose subroutines of modular design and is substantially machine-independent.

Keywords: Alphanumeric data files; data retrieval; editing program; file editor; FORTRAN program; general-purpose modular programming; reformatting program; report generator; searching program.

1. Introduction

Conventional methods of computer software development involve determining the specific needs which the program being written is to serve, and designing it specially to meet those needs. Usually care is taken to optimize the efficiency of the program. Such custom-made programs are specialized to the particular problem and must be modified if any changes in operation are desired. Such changes generally require familiarity with the program and the same degree of skill as the writing of the original program. This custom approach may be suitable for massive systems programs such as compilers, which are designed for versatility, or for other programs which will be used hundreds or thousands of times. But for more modest applications work it has drawbacks. One is the limited utility of programs written in this manner. While a compiler is intended to process an unlimited variety of source programs, an ordinary applications program can usually solve only one specific problem. If another problem, similar but not identical, must be solved later, extensive changes in the original program are required. Since many applications programs are run only a few times after they are debugged, or even only once, a large amount of programming and debugging time is spent in proportion to the amount of useful output.

Our own involvement in general-purpose programming dates back to 1961 with the design and implementations of OMNIFORM I [1] and OMNITAB [2,3,4]. The latter is one of the more extensive and successful general-purpose

programs developed for numerical and statistical analysis. OMNITAB has found interesting and important applications in fields almost as diverse as FORTRAN itself has. More modest general-purpose programming efforts here have produced SUBSTITUTE [5], which replaces specified character strings from a data file by other strings as indicated on control cards; and REFORM [6] which rearranges fixed-field records and inserts strings according to a user defined plan.

Our preoccupation with general-purpose programs arises from our need to cope with a large number of diverse data files and systems. Only recently, since the advent of the software industry, have there appeared for sale or lease a number of programs of a general-purpose nature. Prominent among these are a variety of data management and management informations systems [7,8,9].

It is our view that many application programmers' and systems analysts' views on program efficiency are unduly colored by early experience with primitive compilers and computer systems. As computers have become faster and as core have grown larger and cheaper, over-all man-machine efficiency considerations often outweigh consideration of optimum machine use. While these classical concerns are still important for long repetitive computer runs or for processing large files, they are really not important for a large fraction of the day to day operations on a modern computer. Moreover, we have found that, for nonnumerical applications, on the whole, a special-purpose program written to solve a specific problem will be no simpler than a general-purpose program capable of solving it and many other related although not identical problems.

General-purpose programming is greatly facilitated by the use of subroutines of modular design. This resembles the construction of a house from prefabricated parts. Just as certain kinds of wall or flooring components are common to buildings of very different design, certain types of subroutines are widely used in a variety of differing applications. Scientific programmers have largely recognized that the same matrix package may be used for both linear programming and quantum mechanics, and that it is not necessary to write two separate specialized program. Persons writing programs for such applications as file management, selective dissemination of information, information storage and retrieval, and text editing, on the other hand, have made minimal use of modular subroutines. Most of them have preferred to write their programs from scratch, much like a builder refusing to use prefab parts on the assumption they will be the wrong size or shape.

We have found that in non-numerical file manipulation and information retrieval, modular subroutines can be as useful and as versatile as in scientific programming. For example, the SUBSTITUTE program, adapted as a subroutine, can be used for such varied jobs as report generation, correction of errors, and computer-assisted typesetting. [5,10]

2. Overall Design of COMBO

This report describes a program, COMBO, developed by the Data Systems Design Group in line with our objectives which include preparing computer programs for handling the varied data file formats used by the Data Centers of the National Standard Reference Data System (NSRDS).

COMBO provides a searching capability and a report generation facility for use on magnetic tape files having a certain structure. A listing of part of such a file, in this case, a file of bibliographic information on molten salts, is shown in Figure 1. The file is blocked in physical records each consisting of a fixed number of lines or card images. Each card image contains an identification number in a fixed position (in Figure 1, columns 1 through 5). All those consecutive records having the same ID number make up one item, or entry, or logical block. One entry may include up to 99 different types of lines, each of them having its own format and referring to different attributes. Each line contains a code in a fixed position (in Figure 1, columns 6 and 7) which tells which type it is. (A 0 in column 6 means the line lists properties covered in a bibliographic reference. A 30 in columns 6 and 7 indicates a journal card; a 32, a card listing authors; and an 11, a compound discussed in the paper cited.)

It should be noted that the number of compound cards for one entry is variable. Many such files, in which several of the same type of line may occur in one entry, are used in NSRDS and elsewhere. COMBO was designed to handle such files. It processes each card image independently and is thus able to accept any number of the same type of lines in one entry. Some of the reformatting programs developed by DSDG [3] interchange pieces between lines: They permit only one line of each type in an entry, and hence are not directly applicable to files structured in the above manner.

The program reads all those records referring to one item prior to searching. Then it searches to see whether certain words, phrases, or combinations of phrases are to be found. Two levels of Boolean logic are provided: One can accept all those items for which one or more lines satisfied the search, or only those for which at least one line of each type satisfied it; similarly, one can accept any record containing any search string, or only a record containing all the search strings provided for the line type. The user has the option of suppressing searching, in which case every record will be accepted.

If an item satisfies the search it may be edited record by record. This may be done in two stages. In the first stage, a record is divided into fields, either by scanning for a specified field delimiter or by using a fixed origin and width as the field definition. The fields can then be rearranged, and insert strings may be placed between them. Separate reformatting instructions may be provided for records referring to each of the different types of lines.

In the second phase of editing, each line is scanned for the occurrence of specified strings. If any of these strings are found, they will be replaced by substitution strings. This feature can be used, for instance, to expand abbreviations or replace numerical codes by alphabetic information prior to report generation. Once again, separate instructions for each line type may be provided as to what string substitutions are to be performed. Both stages of editing are optional. (As a trivial case, searching, reformatting, and string substitution may all be suppressed. In that case COMBO will act as a tape print utility.)

After reformatting and string substitution, the selected items will be printed as a report. Each type of line may also be written onto one or more magnetic tapes. The ability to direct different types of lines to different tape units permits division of a file, or simultaneous abridgment into several differently abridged files. An item that does not satisfy the search is skipped, and the program resumes reading the input tape.

```
010061102IN
010061103LICL-KCL
0100630J. ELECTROCHEM. SOC.                                          107 7051960
0100632LAITINEN H.A.          BHATIA B.B
010070EMF
010071101RACL2
010071102KCL
```

```
010071180WCL2
010071181MOCL5
010071182RHCL3
010071183RHCL2
010071184PTCL
010071185PTCL2
010071186WCL4
010071187RUCL3
010071188WCL5
010071189RHCL
010071190WCL6
010071191PTCL3
010071192PTCL4
010071193AUCL3
010071194AUCL
010071195CCL4
0100730J. ELECTROCHEM. SOC.                                          103 8 1956
0100732MALMBERG M.S.          RUBIN B.                HAMER W.J.
010080CMPF
010081101AGCL-AGNO3
0100830CAN. J. CHEM.                                                 32 8641954
0100832WETMORE F.E.           HILL S.
010090EMF
010091101AGNO3-NACL-NAAO3
0100930J. PHYS. CHEM.                                                6410381960
0100932BLANDER M.             BRAUNSTEIN J.           HILL D.G.
010100EMF
010101101AGNO3(NANO3-KNO3)
010101102NACL(NANO3-KNO3)
0101030J. PHYS. CHEM.                                                6518661961
0101032BLANDER M.             HILL D.G.
010110KA
010111101AGCL(KNO3)
010111102AGCL2(KNO3)
0101130J. PHYS. CHEM.                                                6620691962
0101132MANNING D.L.           BLANDER M.              BRAUNSTEIN J.
010120DECPEMF
```

Figure 1. Part of a listing of a bibliographic file on molten salts, suitable for input to COMBO. The tape is blocked in physical records of 45 80-character images each. The first five characters of each line are a logical block ID; the next two are a line type code. Note that the paper by Laitinen and Bhatia contains data on 95 compounds.

```
(01006)  JRNL:J.ELEC. SOC.  VOL.107 7051960
(01006)  AUTHORS:LAITINEN H.A.    ,BHATIA B.B.      ,

(01013)  DECP
(01013)  SYSTEM #01     NACL
(01013)  SYSTEM #02     NACL-KCL
(01013)  SYSTEM #03     NACL-LICL
(01013)  SYSTEM #04     NABR-KBR
(01013)  SYSTEM #05     NAI-KI
(01013)  SYSTEM #06     KCL-NACL
(01013)  SYSTEM #07     KI-RBI
(01013)  SYSTEM #08     KI-CSI
(01013)  SYSTEM #09     KCL
(01013)  SYSTEM #10     KCL-LICL
(01013)  SYSTEM #11     KI-NAI
(01013)  SYSTEM #12     KBR-NABR
(01013)  SYSTEM #13     RBCL
(01013)  SYSTEM #14     RBBR
(01013)  SYSTEM #15     RBI
(01013)  SYSTEM #16     CSCL
(01013)  SYSTEM #17     CSI
(01013)  JRNL:Z. ELEKTROCHEM.        VOL. 31 2871925
(01013)  AUTHORS:NEUMANN B.       ,RICHTER H.       ,

(01014)  DECP
(01014)  SYSTEM #01     AL2O3
(01014)  JRNL:Z. ELEKTROCHEM.        VOL. 36 1791930
(01014)  AUTHORS:DROSSBACH P.     ,                 ,

(01015)   EMF
(01015)  SYSTEM #01     AGCL
(01015)  SYSTEM #02     PBCL2
(01015)  SYSTEM #03     ZNCL2
(01015)  SYSTEM #04     CDCL2
(01015)  SYSTEM #05     MGCL2
(01015)  JRNL:Z. ELEKTROCHEM.        VOL. 40 3521934
(01015)  AUTHORS:RAU E.A.         ,GRUBE G.         ,

(01023)  DHSL
(01023)  SYSTEM #01     NARE04-NANO3
(01023)  SYSTEM #02     KRE04-KNO3
(01023)  SYSTEM #03     RBRE04-RBNO3
(01023)  SYSTEM #04     CSRE04-CSNO3
(01023)  SYSTEM #05     KCL04-KNO3
(01023)  SYSTEM #06     NAS04-NANO3
(01023)  SYSTEM #07     NACR04-NANO3
(01023)  SYSTEM #08     NAN004-NANO3
(01023)  JRNL:J. PHYS. CHEM.         VOL. 6727501963
(01023)  AUTHORS:KLEPPA O.J.      ,MESCHEL S.V.     ,
```

Figure 2. Part of the report generated by COMBO in response to the control cards shown in Figure 3. A portion of the tape file used as input to the program is shown in Figure 1.

3. Discussion of the Subroutines

COMBO consists of a main program and eleven subroutines. The main program primarily handles communication between subroutines and the production of final output. Nine of the subroutines are called by the main program and can be used separately as part of another system.

The first subroutine called by the main program is called UNBLOK. It is used to read blocked physical records from a tape and decompose them into 80-character card images. The tape, of which a partial listing is shown in Figure 1, is actually formatted in 3600-character (600-word) blocks. UNBLOK is machine-dependent on the 1108 and its FORTRAN compiler. It can be replaced with a FORTRAN READ if the input tape is in records of not more than 132 characters.

UNBLOK returns one card image at a time to the main program. The card images are then transmitted to a subroutine called BSENS. It scans the ID field and collects card images end-to-end in a block buffer until it detects a change of ID, indicating a new item. It then signals the main program and returns the number of lines in the block buffer.

The main program then takes one card image at a time and determines which type of line it is. It does this by calling a subroutine LINTYP, which compares the card type code against each of the entries in a dictionary. A number corresponding to the code is returned. If the code has not been defined, an error message is printed.

The card image is then transmitted with the "type" number to the search subroutine MSERCH, which scans it for the presence of specified strings. MSERCH, which was originally written to search a Chemical Abstracts condensate tape, is adapted from the SEARCH program written by Carla G. Messina [2]; the main difference is that it provides for up to 99 independent search tables (one for each type of line) rather than one table. Each table can be set up either for an and-search, which is satisfied only if every strings in the table is found in the line at least once, or for an or-search, which is satisfied when any one string in the table is found.

There are two levels of Boolean logic for the search. The first level is that of the individual table. The second is the overall search strategy. If the overall search logic is "and," the search ends unsatisfied when one record on an item fails to satisfy the search, or continues until at least one card image of each type is searched successfully. An overall "or" search is satisfied when any record satisfies the criteria, or continues until every record has been unsuccessfully searched. The overall search logic is written into the main program.

If the search is unsuccessful, the item is skipped and COMBO proceeds with the reading of the input tape. If the search is successful, the main program transmits each line to two editing routines, MXREFM and MSUBS. MXREFM is an expanded and modified subroutine version of the authors' REFORM program [3]. It accepts as input a line of text; breaks it into fragments defined either by fixed columns, or by specified delimiters separating the pieces, or by a flag preceding each piece; and rearranges the fragments as directed, placing insert strings between them if desired. A capability is provided to place each of the fragments, which may be of variable width, in a fixed-width field, either flush left, flush right, or

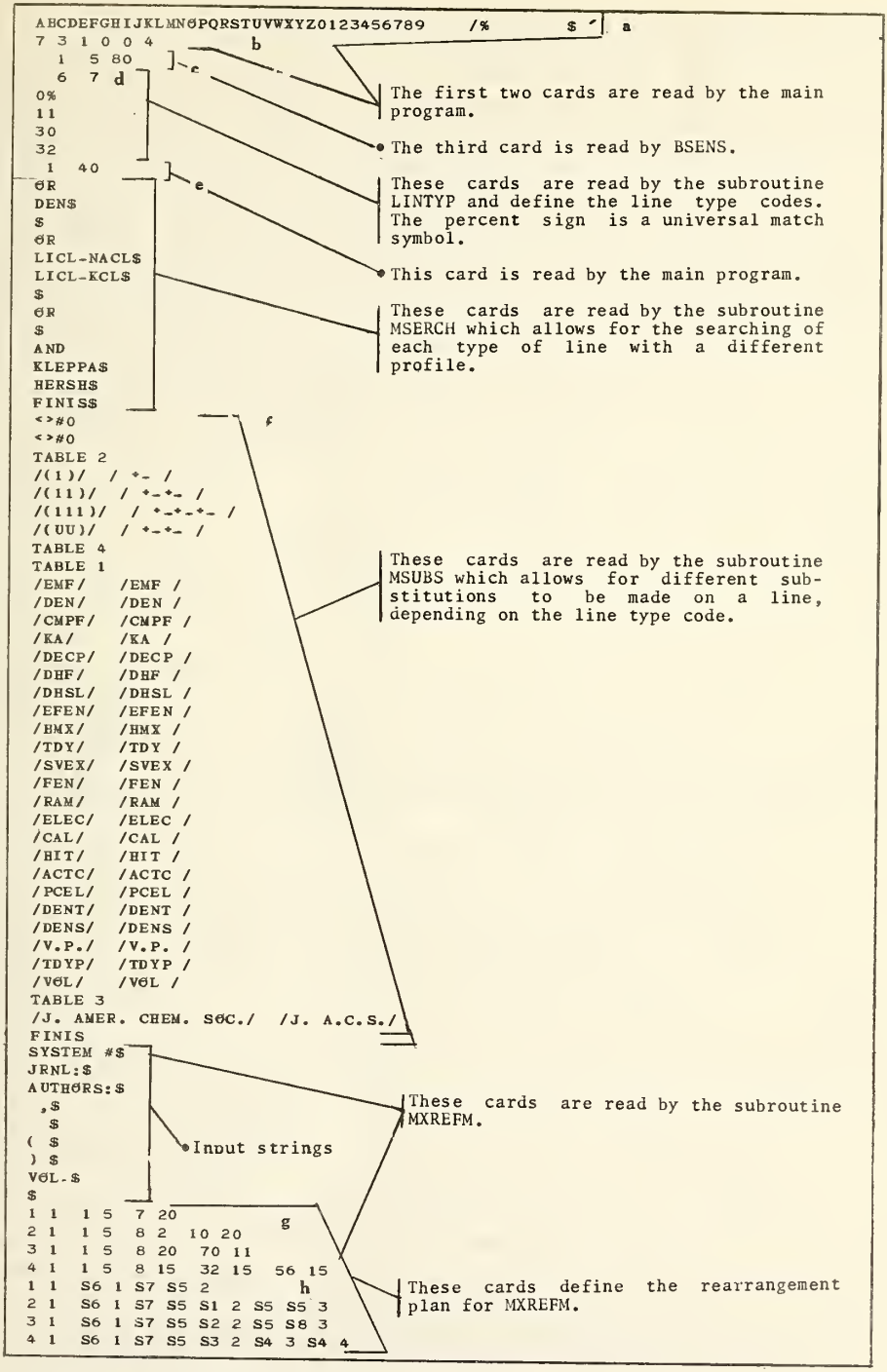


Figure 3. Control cards used to print the report shown in Figure 2 from the molten salts file. A lower case bold face letter refers to a detailed explanation of a control card in Appendix II.

centered. MXREFM, unlike REFORM, reformats only one line at a time, selecting the format for the particular type of line.

MSUBS is called after MXREFM. It is a string substitution routine adapted from a program known as AMSUB, written by Robert C. Thompson for computerized typesetting, which used the structure and logic of Mrs. Messina's SUBSTITUTE. Like SUBSTITUTE, MSUBS scans for specified strings which it replaces with substitution strings. MSUBS differs from the earlier program in that it can accommodate up to 99 different tables and uses the one specified for the particular line type. Some features of AMSUB for typesetting use have been retained in MSUBS but not checked out. Among these is provision for use of a special character to shift to upper case or shift to lower case. Two subroutines, SUNLK and SULOCK, are used to handle the shifting.

```
0000204   -1    -1  -1     -1    -1YESYESREPORT 100FFICE 08REPORT 02NO 00
0000205NO 00YES-1YES-1NO 00NO 00NO 00YES-1NO 00 -5 -5NO              -1
0000206NONE            HOSP    -1-'                -106TRYING TO START AUTO
00002070WN ACTION08NONE          NONENOT ALONE   -3SMOTHERED   RAN(PANIC)
0000208REMOVED     MIN  002   -3 -3MIN  002   -1 -1   -3   -3   -1   -1
0000209   -1   -1   -1   -1   -1   -160CARBURETOR/AUTO     SPARK/BACKFIRE 0101
0000210GASOLINE      EXPLOSION      -3                -3             -30NLY  -3  -3
0000211AIR16SHIRT                        -4-11STNO  -1 -1       -1       -1YES      1A
0000212A1-1 -1-1-1-1-1                              -1NONE          -4 0              1A
0000213A1-1 -1-1-1-1-1                              -1      -4   -4   -4           1A
0000214A1-1 -1-1-1-1-1                              -1      -4   -4   -4           1A
0000215A1-1 -1-1-1-1-1                              -1      -4   -4   -4           1A
0000216A1-1 -1-1-1-1-1                              -1      -4   -4   -4           1A
0000217A1           -1-1             -1             -1      -4   -4   -4           1A
0000218A1           -1-1             -1             -1 0    -4   -4   -4 -4 -4     1A
0000219A1           -1                                                            1A
00003010/18      08COLODENVER             -11N 01HOME            02BASEMENT
000030202BASEMENT       YES1145FRI09176509/20/6509/20/65YESNO NO NO NO
000030329F WHITE SINGLE 06HOMEMAKER      EXACT  1692-1-3              -3-3
0000304EXACT  1692RENTEXACT.R    65YESYESREPORT 08REPORT 08REPORT 00YES-1
*****    280 WORDS IN BLOCK      4 *****


0000305NO 00NO 00YES-1YES-1NO 00NO 00NO 00NO 00 -5 -5NO NONE
0000306NONE            T&R    NO -3              -320LIGHTING COAL FURNACE
00003070WN ACTION08NONE          NONENOT ALONE   -3SMOTHERED             -3
0000308          -3EXACT  0   -3 -3   -3 -3   -1 -1   -3   -3   -1   -1
0000309   -1   -1   -1   -1EXACT    001FURNACE/COAL         EMBER          0101
0000310KEROSENE      EXPLOSION      -3                -3             -30NLY  -3  -3
0000311AIR37SOCKS                        -4-11STNO  -1 -1       -1       -1YES      1A
0000312A101 -1-1-1-1-1COTTON                          NONE          -4 0              1A
0000313A1-1 -1-1-1-1-1                              -1      -4   -4   -4           1A
0000314A1-1 -1-1-1-1-1                              -1      -4   -4   -4           1A
0000315A1-1 -1-1-1-1-1                              -1      -4   -4   -4           1A
0000316A1-1 -1-1-1-1-1                              -1      -4   -4   -4           1A
0000317A1           -1-1             -1             -1      -4   -4   -4           1A
0000318A1           -1-1             -1             -1 0    -4   -4   -4 -4 -4     1A
0000319A1           -1                                                            1A
00004010/19      08COLODENVER             -11N 01HOME            02BASEMENT
```

Figure 4. A listing of a magnetic tape file used as input to COMBO. The information on this tape had been recorded in blocks of 20 84 character card images. The lines starting with "***** 280 WORDS IN BLOCK" were inserted in the printout by the tape listing utility program. The first five columns of each card image contain a case number. The next two contain a line type code.

A general-purpose subroutine, INPUT, is used in COMBO to interpret the control cards. INPUT accepts an 80-character card image in (80A1) format. Any strings of digits are converted to integers. It returns a vector 30 words long containing either alphabetic characters or integers, and a second, parallel vector indicating whether a word is to be read as a character or as an integer. INPUT may be used by any program for reading free-form control cards containing integers, punctuation (which is usually ignored), and alphabetic characters.

After editing, the main program calls the subroutine TAPOUT to write the edited output line on any tape units or other peripheral devices. If the particular type of line is not to be written on tape, or if there is no tape output in this run, TAPOUT returns without action. It writes all the output on a single unit if this was specified. If different types of lines are being written on different units, the line is written on the appropriate devices. (In this case, special instruction cards must have been read providing the list of units to be used.)

```
0000109   -1   -1   -1   -1   -1   -140MATCHES              OPEN FLAME     00-3
0000110            -3              -3-3              -3              -30NLY  -3  -3
0000111A1R51PAJAMA                        -4-11STNO  -1  -11NCREASED       -1NO
**002010/12      08COL00ENVER              -10UT01HOME          17YARD
000020217YARD          YES1700MONO9136509/14/6509/14/65YESNO NO NO NO
000020316M MEX.AMSINGLE 07STUDENT         EXACT   96010-1              -1-1
0000204   -1   -1   -1   -1   -1YESYESREPORT 100FFICE 08REPORT 02NO 00
0000205NO 00YES-1YES-1NO 00NO 00NO 00YES-1NO 00 -5 -5NO              -1
0000206NONE          HOSP    -1-1              -106TRYING TO START AUTO
00002070WN ACTION08NONE            NONENOT ALONE   -3SMOTHERED   RAN(PANIC)
0000208REMOVED     MIN  002   -3 -3MIN  002   -1 -1   -3   -3   -1   -1
0000209   -1   -1   -1   -1   -1   -160CARBURETOR/AUTO     SPARK/BACKFIRE 0101
0000210GASOLINE      EXPLOSION       -3              -3              -30NLY  -3  -3
0000211A1R16SHIRT                        -4-11STNO  -1 -1       -1       -1YES
**003010/18      08COL00ENVER              -11N 01HOME          02BASEMENT
000030202BASEMENT      YES1145FRI09176509/20/6509/20/65YESNO NO NO NO
000030329F WHITE SINGLE 06HOMEMAKER       EXACT  1692-1-3              -3-3
0000304EXACT  1692RENTEXACT.R    65YESYESREPORT 08REPORT 08REPORT 00YES-1
0000305NO 00NO 00YES-1YES-1NO 00NO 00NO 00NO 00 -5 -5NO NONE
0000306NONE          T&R    NO -3              -320LIGHTING COAL FURNACE
00003070WN ACTION08NONE            NONENOT ALONE   -3SMOTHERED             -3
0000308          -3EXACT  0   -3 -3   -3 -3   -1 -1   -3   -3   -1   -1
0000309   -1   -1   -1   -1EXACT    001FURNACE/COAL         EMBER          0101
0000310KEROSENE      EXPLOSION       -3              -3              -30NLY  -3  -3
0000311A1R37SOCKS                        -4-11STNO  -1 -1       -1       -1YES
**004010/19      08COL00ENVER              -11N 01HOME          02BASEMENT
000040202BASEMENT      YES1145FRI09176509/20/6509/20/65NO YESNO NO NO
000040303F WHITE CHILD  10CHILD             -3    -30006HOMEMAKER       -1
0000404EXACT  1692RENTEXACT.R    65YESYESREPORT 08REPORT 06REPORT 02NO 00
0000405NO 00NO 00NO 00YES-1NO 00YES-1NO 00NO 00 -5 -5NO              -1
0000406NONE          HOSP    -1-1              -120WATCHING MOM LIGHT FURN.
00004078YSTANDER 08NONE            NONENOT ALONE  YESSCREAMED             -3
0000408          -3MIN   04   -3 -3   -3 -3   -1 -1   -3   -3   -1   -1
0000409   -1   -1   -1   -1   -1   -101FURNACE/COAL         EMBER          0101
0000410KEROSENE      EXPLOSION       -3              -3              -30NLY  -3  -3
0000411A1R16SHIRT                        -4-10U NO  -1 -1       -1       -1YES
```

Figure 5. The abridged file produced from operation of COMBO on the file shown in Figure 4, as instructed by the cards shown in Figure 6.

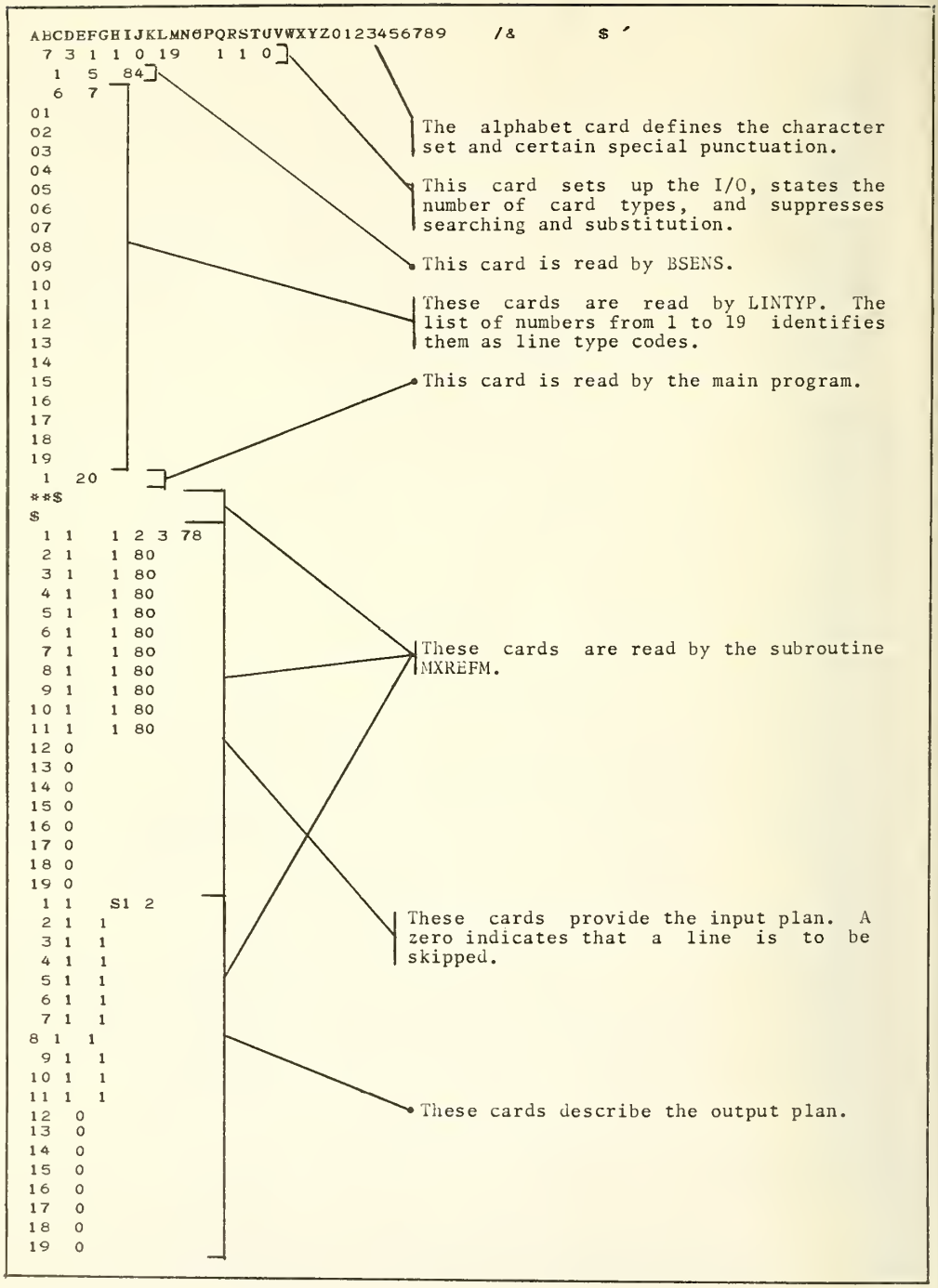


Figure 6. Control cards used to abridge the file shown in Figure 4, by retaining only the type 1 through type 11 records. Figure 5 shows the abridged file.

```
ABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789     /%        $ '
 7 8 1 1 0 19 1 1 0
 1  5  84
 6 7
01
02
16
17
18
19
  1  50
**$
*$
#$
 $
$
1 1  1 5  6 7  8 10  18 2  20 4  24 13  37 7  44 3  47 2  49 15  64 2  66 13
2 0
3 0
4 0
5 0
6 0
7 0
8 0
9 0
10 0
11 0
12 0
13 0
14 0
15 0
16 0
17 0
18 0
19 0
1 1  S3 1 S4 2 S4 3 S1 4 S4 5 S4 6 S2 7 S4 8 S4 9 S2 11 S4 12 <---
2 0
3 0
4 0
5 0
6 0
7 0
8 0
9 0
10 0
11 0
12 0
13 0
14 0
15 0
16 0
17 0
18 0
19 0
```

Figure 7. Control cards used to print the first line (card images) of each case from the file shown in Figure 4, inserting asterisks between items. Output is shown in Figure 8. The card marked with the arrow was changed to produce the alternate output shown in Figure 9.

4. Applications of COMBO

The COMBO program was originally written to search and edit the magnetic tape file of bibliographic information on molten salts, part of which is listed in Figure 1. This file was blocked in physical records consisting of 45 80-character card images each. Control cards shown in Figure 3 were used to specify the strings for which the records will be searched, and the substitution and reformatting instructions. An explanation of the control cards is given below in the section on "Discussion of the Control Cards." The examples cited in that section refer to Figure 3. The report printed by COMBO is shown in Figure 2.

Appendix I shows printout produced by COMBO as it interpreted the control cards shown in Figure 3. This output, printed before it began to read from the input tape, provides much the same information as does Figure 3, but in some cases a less compact format has been used, and sometimes the program has added comments or explanations to the listing.

The application of COMBO to that file utilized all of its capabilities: searching, string substitution, and reformatting. But the program can also be used for applications using only some of its capabilities. Figure 4 shows a partial listing of a magnetic tape file of case reports on ignition of clothing. It was desired to abridge, reformat, and print the file without searching or substituting. Figure 6 shows control cards used to print the first 11 types of card images for each case report. Figure 5 shows part of the abridged listing produced in this manner.

Figure 7 shows another COMBO operation on the same file. Here it was desired to print only the first line of each case from that file, inserting delimiters between different items of information. This was done twice, using the asterisk as the usual separator in one run, and the space in the other. Only one control card, marked with an arrow in Figure 7, had to be changed. The results of these operations are shown in Figure 8 and 9.

The change just described is one of a number of possible changes in the operation of COMBO that can be made by changing only one control card. In this case, calls for one string, consisting of an asterisk, were replaced with calls for a different string, consisting of a space. Here we wished to change only some of the asterisks to spaces. If we had wished to replace all the asterisks with spaces, it would have been even simpler. The card defining String 2 as an asterisk could have been replaced with one providing for a space.

Other changes involving only one control card include changing the order in which fragments of an input line are printed, or the columns which define a fixed input field. One could also change a pair of substitution strings or a search phrase by changing only one card.

In the two data bases just discussed the block or case number and the card type number are contiguous and at the beginning of each image. Figure 10 shows cards making up one logical block of a data base on cases heard by an Italian court. In this data base the case number is in columns 73-79 and the card type code in column 1. COMBO can be used on this file to abridge it for purposes of storing it on disc for on-line query. In addition, the substitution capability of COMBO could be used to replace a limited vocabulary of Italian words with their English equivalents.

```
#00001•010/8  •D/8      ••08•COLD•DENVER      •    -1•IN •01•HOME OF GRANOMA•03•BEDRDDM
#00002•010/12 •D/12     ••08•COLO•DENVER      •    -1•OUT•01•HOME           •17•YARD
#00003•010/18 •D/18     ••08•COLO•DENVER      •    -1•IN •01•HOME           •02•BASEMENT
#00004•010/19 •D/19     ••08•COLO•DENVER      •    -1•IN •01•HOME           •02•BASEMENT
#00005•010/37 •D/37     ••08•CDLO•WALTON      •    -1•OUT•08•SIDE OF ROAD   •19•SIDE OF RDAD
#00006•010/26 •D/26     ••08•COLO•DENVER      •    -1•OUT•01•HOME           •17•BACK YARD
#00007•010/23 •D/23     ••08•COLO•DENVER      •    -1•IN •01•HOME           •05•LIVING ROOM
#00008•01D/43 •D/43     ••08•COLO•DENVER      •    -1•IN •02•MOTEL RESIDENCE•02•BOILER RODM
#00009•010/58 •D/58     ••08•COLD•DENVER      •    -1•DUT•2D•HUNTING COUNTRY•19•CANYDN
#00010•010/62 •D/62     ••08•COLO•DENVER      •    -1•OUT•01•HOME           •17•BACK YARD
#00011•01D/96 •D/96     ••08•COLO•DENVER      •    -1•IN •01•HOME           •D8•KITCHEN
#00012•D10/85 •D/85     ••08•COLD•DENVER      •    -1•DUT•01•HOME           •17•BACK YARD
#00013•01D/81 •D/81     ••08•COLO•DENVER      •    -1•OUT•D8•ALLEY NR. HOME •19•ALLEY
#00014•010/87 •D/87     ••D8•COLO•TRINIDAD    •    -1•OUT•09•WOODED AREA    •19•PATH IN WOODS
#00015•010/102•D/102    ••08•COLO•DENVER      •    -1•OUT•01•HDME           •17•BACKYARD
#00016•010/113•D/113    ••08•COLD•DENVER      •    -1•IN •01•APARTMENT      •08•KITCHEN
#00017•010/109•D/109    ••08•COLO•DENVER      •    -1•IN •01•HOME           •02•BASEMENT
#00018•010/120•D/120    ••08•COLO•DENVER      •    -1•IN •01•HOME           •02•BASEMENT
#00019•01D/127•D/127    ••08•COLO•DENVER      •    -1•IN •01•HOME           •05•LIVING RDOM
#00021•010/116•D/116    ••08•COLO•DENVER      •    -1•OUT•02•NEIGHBDR'S YARD•17•NEIGHBOR YARD
#00022•01D/123•D/123    ••08•COLO•AURORA      •    -1•IN •01•HDME/MDTEL UNIT•03•BEDROOM
#00023•010/131•D/131    ••08•COLO•DENVER      •    -1•IN •01•HOME/APARTMENT •05•LIVING ROOM
#00024•010/150•D/150    ••08•COLO•DENVER      •    -1•IN •01•HOME           •08•KITCHEN
#00025•010/145•D/145    ••08•COLO•FREDERICK   •    -1•IN •01•HOME           •08•KITCHEN
#00026•010/169•D/169    ••08•COLO•DENVER      •    -1•IN •01•HOME/APARTMENT •D3•BEDRDOM
#00027•010/172•D/172    ••08•COLO•DENVER      •    -1•IN •01•HOME           •08•KITCHEN
#00028•010/293•D/293    ••08•COLO•DENVER      •    -1•IN •01•HOME           •03•BEDRODM
#00029•010/196•D/196    ••08•COLO•DENVER      •    -1•IN •01•HOME/APARTMENT •05•LIVING ROOM
```

Figure 8. A printout of the first card image of each case from the file in Figure 4. After seeing this output it was decided to use a blank space rather than an asterisk to separate fields. This was done by changing one control card, as shown in Figure 9.

```
#00001 01D/8   D/8      ••08 COLD DENVER      •   -1 IN  01 HDME OF GRANOMA•D3 BEDROOM
#00002 01D/12  D/12     ••08 COLO DENVER      •   -1 OUT 01 HOME          •17 YARD
#00003 01D/18  D/18     ••08 COLO DENVER      •   -1 IN  01 HDME.         •02 BASEMENT
#00004 01D/19  D/19     ••08 COLO DENVER      •   -1 IN  01 HOME          •02 BASEMENT
#00005 01D/37  D/37     ••08 COLD WALTON      •   -1 OUT 08 SIDE OF RDAD  •19 SIDE OF RDAD
#00006 01D/26  D/26     ••08 CDLD DENVER      •   -1 OUT 01 HDME          •17 BACK YARD
#00007 01D/23  D/23     ••08 COLO DENVER      •   -1 IN  01 HOME          •05 LIVING RDDM
#00008 01D/43  D/43     ••08 COLD DENVER      •   -1 IN  02 MOTEL RESIDENCE•02 BOILER RODM
#00009 01D/58  D/58     ••08 COLO DENVER      •   -1 DUT 2D HUNTING COUNTRY•19 CANYON
#00010 01D/62  D/62     ••08 COLO DENVER      •   -1 OUT 01 HOME          •17 BACK YARD
#00011 01D/96  D/96     ••08 COLD DENVER      •   -1 IN  01 HOME          •D8 KITCHEN
#00012 01D/85  D/85     ••D8 COLD DENVER      •   -1 OUT 01 HDME          •17 BACK YARD
#00013 010/81  D/81     ••08 CDLO DENVER      •   -1 DUT D8 ALLEY NR. HDME •19 ALLEY
#00014 01D/87  D/87     ••08 CDLO TRINIDAD    •   -1 DUT D9 WDDDED AREA    •19 PATH IN WDODS
#00015 01D/102 D/102    ••08 CDLO DENVER      •   -1 DUT 01 HDME          •17 BACKYARD
#00016 010/113 D/113    ••08 COLD DENVER      •   -1 IN  01 APARTMENT      •08 KITCHEN
#00017 01D/109 D/109    ••08 CDLD DENVER      •   -1 IN  D1 HDME          •D2 BASEMENT
#00018 01D/12D D/120    ••08 CDLD DENVER      •   -1 IN  D1 HDME          •02 BASEMENT
#00019 01D/127 D/127    ••08 CDLD DENVER      •   -1 IN  D1 HDME          •D5 LIVING RDDM
#00021 010/116 D/116    ••08 CDLO DENVER      •   -1 OUT 02 NEIGHBOR'S YARD•17 NEIGHBOR YARD
#00022 01D/123 D/123    ••08 CDLD AURORA      •   -1 IN  01 HDME/MDTEL UNIT•03 BEDRODM
#00023 D10/131 D/131    ••08 COLO DENVER      •   -1 IN  01 HOME/APARTMENT •05 LIVING RODM
#00024 01D/150 D/150    ••08 CDLO DENVER      •   -1 IN  D1 HOME          •08 KITCHEN
#00025 01D/145 D/145    ••08 COLO FREDERICK   •   -1 IN  01 HOME          •08 KITCHEN
#00026 01D/169 D/169    ••08 CDLO DENVER      •   -1 IN  01 HDME/APARTMENT •D3 BEDROOM
#00027 01D/172 D/172    ••08 COLD DENVER      •   -1 IN  01 HOME          •08 KITCHEN
#00028 01D/293 D/293    ••08 CDLD DENVER      •   -1 IN  01 HDME          •03 BEDROOM
```

Figure 9. Printout of the first card image of each case, using a space as the separator. The control cards were the same as in Figure 7, except that the control card marked with the arrow was changed to
1 1 S3 1 S4 2 S4 3 S1 4 S4 5 S4 6 S2 7 S4 8 S4 9 S2 11 S4 12

Figures 11-13 illustrate the use of multiple tape units for output. The control cards shown in Figure 11 caused the tape file listed in Figure 1 to be divided into two abridged files, which were written onto units 8 and 9. Figure 12 lists part of the output written to tape 8. Figure 13 lists the corresponding output written to tape 9.

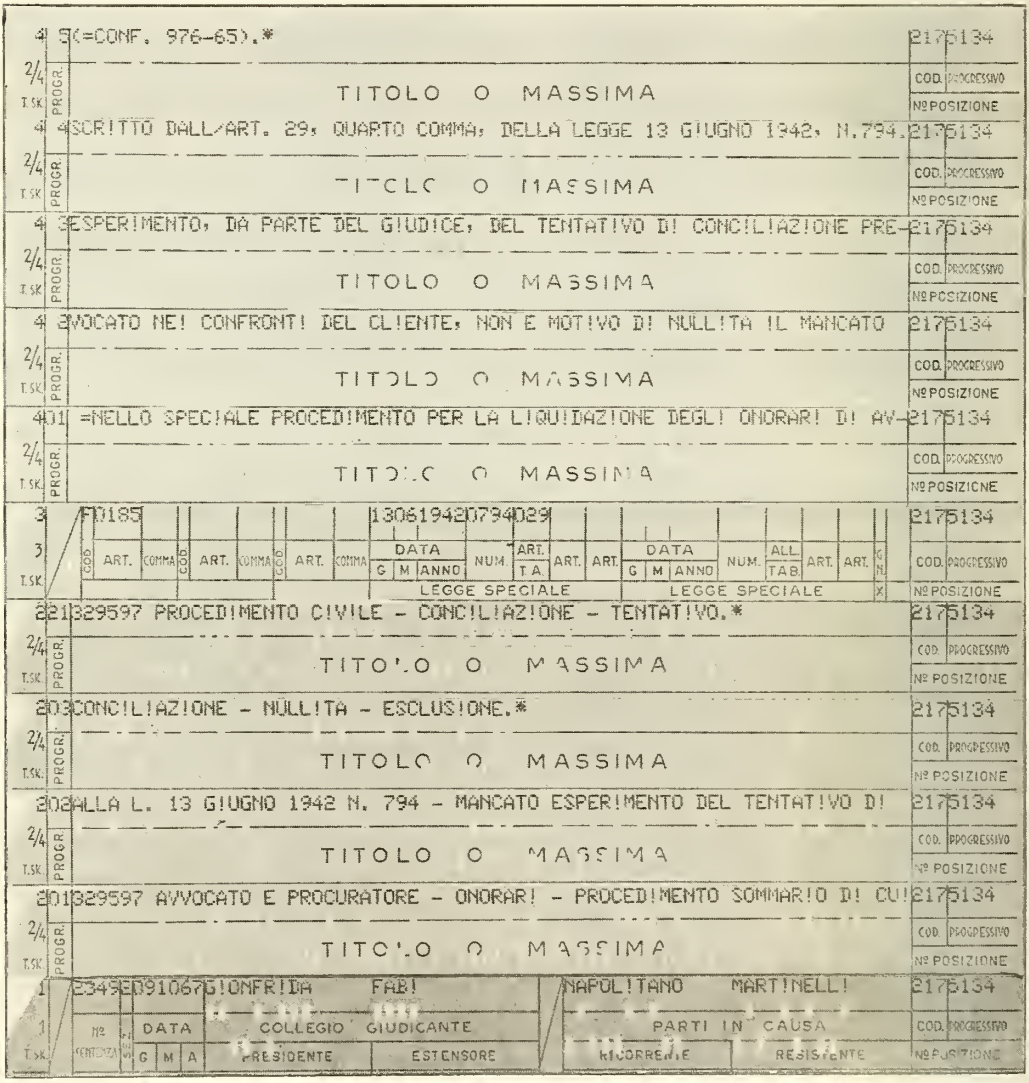


Figure 10. These cards are part of a data base on cases heard by an Italian superior court. This data base could be handled by COMBO, since card columns 73 through 79 contain a case number and the number in column 1 identifies the card type.

The number 2 in the fourth position on the second control card (shown in Figure 11) indicates that multiple tape units are being used for output. The number 1 in the seventh position of the cards suppresses searching. The tape unit table follows the input to LINTYP. It directs the system cards and the property code card for each entry to tape 8 to form a substance-property index, and the author and journal cards to tape 9 as a bibliography.

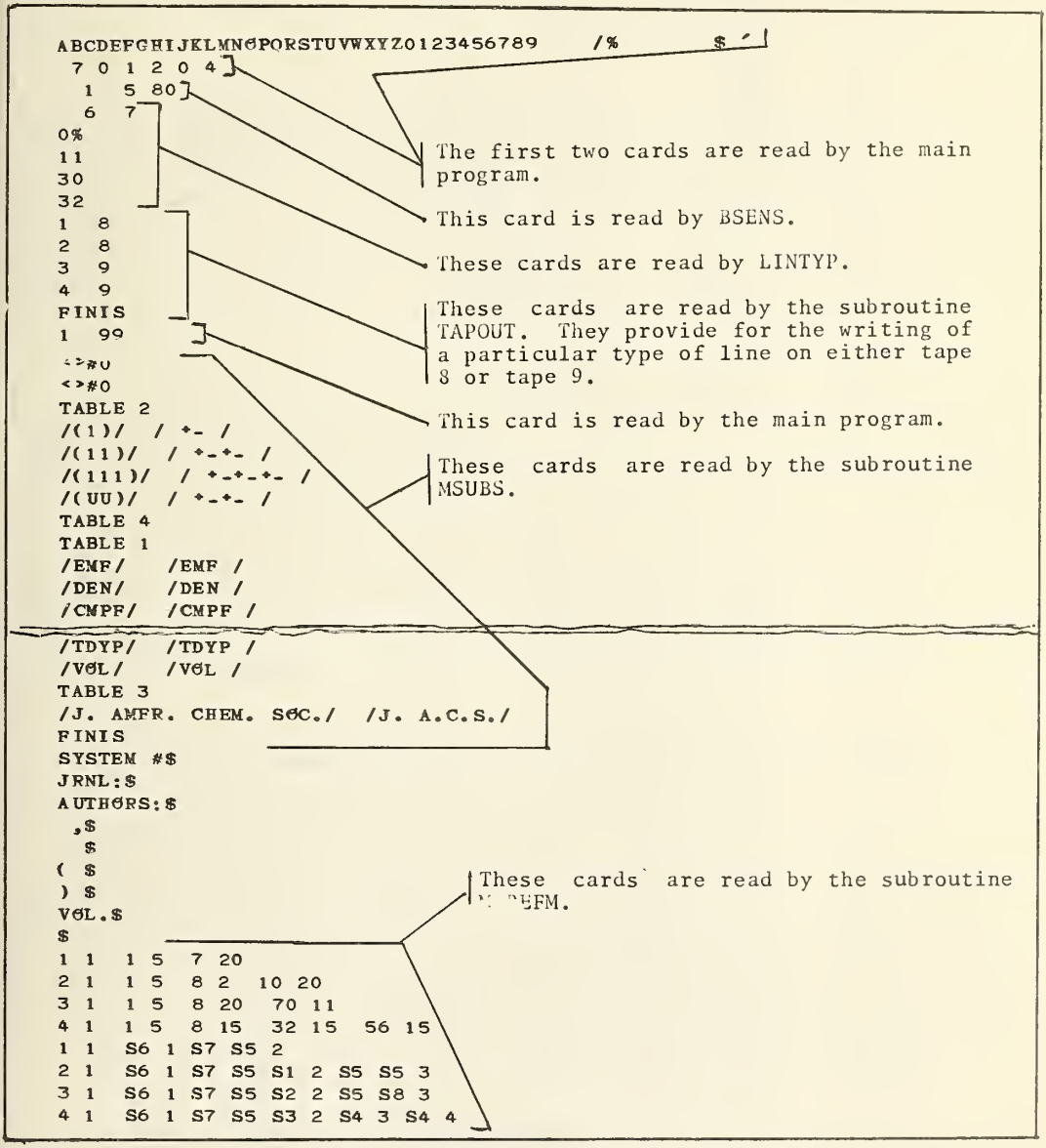


Figure 11. Control cards used to divide the file shown in Figure 1 into two abridged files. The two files produced are shown as Figure 12 and Figure 13.

```
( 01006)   SYSTEM #02     IN
( 01006)   SYSTEM #03     LICL-KCL
( 01007)   EMF
( 01007)   SYSTEM #01     RACL2
( 01007)   SYSTEM #02     KCL
( 01007)   SYSTEM #91     PTCL3
( 01007)   SYSTEM #92     PTCL4
( 01007)   SYSTEM #93     AUCL3
( 01007)   SYSTEM #94     AUCL
( 01007)   SYSTEM #95     CCL4
( 01008)   CMPF
( 01008)   SYSTEM #01     AGCL-AGNØ3
( 01009)   EMF
( 01009)   SYSTEM #01     AGNØ3-NACL-NANØ3
( 01010)   EMF
( 01010)   SYSTEM #01     AGNØ3(NANØ3-KNØ3)
( 01010)   SYSTEM #02     NACL(NANØ3-KNØ3)
( 01011)   KA
( 01011)   SYSTEM #01     AGCL(KNØ3)
( 01011)   SYSTEM #02     AGCL2(KNØ3)
( 01012)   DECP EMF
( 01012)   SYSTEM #01     KCL
( 01013)   DECP
( 01013)   SYSTEM #01     NACL
( 01013)   SYSTEM #02     NACL-KCL
```

Figure 12. The substance-property index produced from the molten salts data base by selecting only the system cards and the property codes card. The control cards which produced this file are shown in Figure 11.

```
( 01006)   JRNL:J. ELECTRØCHEM. SØC.  VØL.107 7051960
( 01006)   AUTHØRS:LAITINEN H.A.    ,BHATIA B.B.      ,
( 01007)   JRNL:J. ELECTRØCHEM. SØC.  VØL.103   81956
( 01007)   AUTHØRS:MALMBERG M.S.    ,PUBIN B.         ,HAMER W.J.
( 01008)   JRNL:CAN. J. CHEM.         VØL. 32 8641954
( 01008)   AUTHØRS:WETMØRE F.E.     ,HILL S.          ,
( 01009)   JRNL:J. PHYS. CHEM.        VØL. 6410381960
( 01009)   AUTHØRS:BLANDER M.       ,BRAUNSTEIN J.    ,HILL D.G.
( 01010)   JRNL:J. PHYS. CHEM.        VØL. 6518661961
( 01010)   AUTHØRS:BLANDER M.       ,HILL D.G.        ,
( 01011)   JRNL:J. PHYS. CHEM.        VØL. 6620691962
( 01011)   AUTHØRS:MANNING D.L.     ,BLANDER M.       ,BRAUNSTEIN J.
( 01012)   JRNL:TRANS. ELECTRØCHEM.   VØL. 69 6611936
( 01012)   AUTHØRS:KIRK R.C.        ,BRADT W.E.       ,
( 01013)   JRNL:Z. ELEKTRØCHEM.       VØL. 31 2871925
( 01013)   AUTHØRS:NEUMANN B.       ,RICHTER H.       ,
( 01014)   JRNL:Z. ELEKTRØCHEM.       VØL. 36 1791930
( 01014)   AUTHØRS:DRØSSBACH P.     ,                 ,
( 01015)   JRNL:Z. ELEKTRØCHEM.       VØL. 40 3521934
( 01015)   AUTHØRS:RAU F.A.          ,GRUBE G.         ,
( 01016)   JRNL:Z. ANØRG. CHEM.       VØL.185 3241930
( 01016)   AUTHØRS:ISBEKØW W.        ,                 ,
( 01017)   JRNL:Z. PHYSIK. CHEM.      VØL.130  391927
( 01017)   AUTHØRS:LØRENZ R.         ,                 ,
```

Figure 13. The bibliography produced from the molten salts file (shown in Figure 1) by selecting only the author and journal lines. The control cards which produced this file are shown in Figure 11.

5. Control Cards

The main program and six of the subroutines -- BSENS, LINTYP, MSERCH, MSUBS, MXREFM, and TAPOUT -- each require one or more control cards. Each of those subroutines has a variable in its argument list which serves as a switch. When set to zero, it directs the subroutine to read control cards and initialize itself. A nonzero value indicates a production call. Sample control cards for the main program and subroutines are shown in Figure 2. A detailed analysis of the same control cards is shown in Appendix I.

The first two control cards are read by the main program. The first card, in (80A1) format, is used to make the program independent of the internal representation of Hollerith characters. It has the letters of the alphabet in columns 1 through 26 in order, and digits from 0 to 9 in columns 27 through 36. Column 41 contains a symbol used in the print output by MSUBS to bracket the strings. Column 42 contains a special character, the universal match symbol, the function of which will be explained shortly, used in LINTYP and MSERCH. Column 50 contains a string terminator used to delimit the strings on control cards read by MSERCH and MXREFM. Column 52 contains a character which MXREFM recognizes as a continuation symbol; it means that the next control card will be treated as a continuation. The 47th column of the alphabet card must be blank.

The next card contains not more than nine numbers, in free field. These are the logical unit number for the input tape, the logical unit number for an output tape or card punch if such output is desired, a switch indicating whether a printed report is wanted (0 for no, 1 for yes), another switch indicating whether tape or card output is wanted, the overall search strategy (0 for "or", 1 for "and"), and the number of possible line types for an item in the file. In the example shown, the input tape is on unit 7 and, if tape output were desired, it would be on unit 8. However, although printed output is desired, tape output is not. An overall "or" search strategy will be used. There are to be four line types and therefore four tables each for searching, reformatting, and substitution. The last three numbers are subroutine switches. If nonzero, they will suppress the use of MSERCH, MSUBS, or MXREFM respectively. Those subroutines will be used if the switches are set to zero, or if they are omitted.

The third control card is read by BSENS and is in (I3) format. It gives the starting and final columns of the ID field and the width of a record on tape, in characters. For instance, in Figure 2 the ID field is positions 1 through 5 in an 80-character record.

Special cards are required by TAPOUT at this point if the option to direct different types of lines to different output units is in effect. Each of these cards has a line type number followed by one or more logical units numbers. For instance, "2 3 8" means that each line type 2 should be output on units 3 and 8. Neither the printer nor any read-only device (e.g., card reader) should be specified; no check is made for this error. Line types must be listed in order -- 3 may not be followed by 2 -- but may be omitted from the list. If a type is omitted, no tape output for lines of that type will be generated. A card with FINIS flush left terminates the control cards for TAPOUT. These cards are not used if a single tape unit for all output is specified, or if no tape output has been requested.

The next control card is read by the main program. It contains two numbers in free format. The first is the number of the first logical block to be processed by COMBO. If this number is greater than 1, blocks preceding the first one to process will be read and ignored. The second

number is the last logical block to be processed. After processing it COMBO stops. The second number may be omitted, in which case COMBO will continue to the end-of-file.

The next group of control cards is read by LINTYP. The first one gives the starting and ending columns for the field identifying the line type. In this case it consists of columns 6 and 7. Then as many cards as there are record types follow. Each card has, beginning in column 1 (in this case in columns 1 and 2) the characters which serve as a code for an line type. For instance, the second type of record, for an item of the file shown in Figure 1, enumerating the compounds described, is indicated by "11" in columns 6 and 7. If a universal match character appears, any character may be in its place in the file. For instance, the code for the first record type is "0%", where the percent sign is the universal match character. A record of this type is indicated by a zero in column 6; anything may appear in column 7.

In Figure 2, codes of "30" and "32" indicate records giving the journal in which an article appeared and its authors, respectively. LINTYP expects one card for each type of record.

Cards are then read by MSERCH. One search table is read for each card type. The search table begins with the word AND or OR, stating the search logic for the table. Then come the search strings for the table, each ended by a period. A card with a period in column 1 marks the end of a table. Table 4 has no search strings, so there is to be no search for authors. The word FINIS left-adjusted signals the end of card input to MSERCH. If a string including a universal match character were in the input, it would indicate that any character would satisfy the search.

The next subroutine to read cards is MSUBS. The first two cards, in (3A1,I1) format, give certain information about the input and output, respectively. The first character is a "shift up" symbol and the second a "shift down" symbol. If the text in the file does not have upper and lower case with shift symbols, characters which will not be found in the file, such as the lesser and greater signs, should be used. The third character is a universal match symbol for substitution. The number is a binary switch which, if 1, means the shift and lock convention is in use, or if 0, the shift and unlock convention. This number is unimportant if no shift convention is in force, but in that case should be zero for efficiency. Each substitution table is introduced by a card with the word TABLE in columns 1 through 5, and the table number in columns 7 and 8. Each table entry consists of two strings, a search string to be located in the text and its replacement or substitution string. Each of the strings is delimited on each side by the character in column 1 of the card. Figure 2 shows that each card may have its own delimiter. The word FINIS flush left on a card terminates the reading of substitution tables. Note that the tables need not be in order.

Control cards for MXREFM are read last. The first cards read are the insertion strings, each ended by a terminator (here, the dollar sign.) MXREFM assigns each string a number in order of appearance starting with 1. A card with a string terminator in column 1 ends the reading of strings.

The remaining MXREFM control cards are in free format. These are two groups of them, one defining the input format and one the output format, each consisting of one card (or one card and continuations) for each type of line. The input format card has the number of the table to which it refers as the first number on the card. As a check the cards must be arranged with these numbers in ascending order. The second number tells

which of three types of formats the record will be in. A 1 indicates fixed-field format, a 2 indicates a free-field structured format with a specified delimiter, while 3 indicates an unstructured format with a flag preceding each record. If the input is fixed-field, as shown in Figure 2, pairs of numbers follow, giving the starting column and width of each field. If the input is free-field, a delimiter, consisting of one to three characters, comes next, and then a number which is the maximum number of fields in the record. If no delimiter appears, one or more blanks separate the fields. If the input is unstructured but flagged, the list of flags follows, with each flag bracketed in the same manner as strings in MSUBS.

The fields of the input line are numbered by MXREFM. If the input is free-field but structured, the fields are numbered sequentially; otherwise they are numbered in the order in which they are defined. (A field is defined by stating its fixed columns or the flags associated with it.) These numbers are used to specify how to reassemble the pieces into an edited output line. Each card specifying the output plan has as its first number the number of the table, as a check; these must be in ascending order. The next number is either 1 or 2. If 1, it indicates that the width of a field in the output is to be the same as it was in the input. This should always be provided for fixed-field records. A 2 means that each variable-width field of the input record is to be inserted into a fixed-width output field. Then comes the output specifications. A field of the input is designated by its number. An insert string is indicated by the letter S and its number, e.g., S2 names the second string. If option 2 (the inclusion of variable-width fragments into fixed fields) is in effect, the number specifying an input piece is followed by two more numbers. The first one is a switch, 0, 1, or 2, telling how the input piece is to be adjusted or located in the fixed-width field. A 0 means the piece is to be centered, a 1 indicates flush left adjustment, and a 2 flush right adjustment. If the input piece is wider than the output field part of it will be lost. The second extra number is the width of the fixed field. For instance, "3 1 12" would mean that the third piece was to be left-adjusted in a 12-character field.

It is the policy of the Data Systems Design Group to program in standard FORTRAN wherever possible, to reduce machine- or system-dependence. Since it is not always possible to make a program completely machine-independent, we have segregated non-standard features in special subroutines appropriate to the 1108, which can be replaced by ones required for a different machine.

The principal non-standard aspect of COMBO is the subroutine UNBLOK. This subroutine is used to read records longer than 132 characters, unpack from A6 to A1 format internally, and return 80-character card images. It makes use of NTRAN, a utility tape handler for the 1108, and the FLD statement, used to manipulate bits.

Two other features may require modification for other machines (or other installations using the same hardware). The units for the card reader and printer are equated to variables IN and IOUT in the main program. These values are currently 5 and 6, respectively, but should be set to the proper values for a user's system.

Calls are included to a subroutine CLOCK which is in the 1108 library and prints the time in hours, minutes, seconds, and milliseconds. On other machines it may be necessary to change the name of the routine in the calls or delete the calls.

The common block STR is used by MSERCH, MXREFM, and MSUBS for storage of character string data. Dynamic allocation of string storage is effected by an interesting technique. A pointer, ICNOW, indicates the next unused cell in the vector IC. A subroutine requiring string storage may use those cells in IC immediately beyond ICNOW, recording their location. ICNOW must be reset by the subroutine after using IC.

The present version of COMBO operates in about 50K words of core on the 1108 at NBS. This will, of course, vary not only from machine to machine but from system to system, since it includes the input-output package and depends on the machine language. If it is necessary to decrease the amount of storage required, the size of the common blocks IB and STR can be cut. The amount of storage used by certain arrays in the control sections MSUBS and MXREFM can also be reduced.

Certain limitations, which we hope will not be considered severe, have been imposed in COMBO. Three of these are worth mentioning. One is that only 250 characters have been provided for the dictionary of record type codes: The product of the number of record types and the width of the record type field may not exceed 250. (If the record type field is columns 69 through 72, for instance, there may be no more than 62 different record types.) Another is that COMBO cannot accept a logical block longer than 20000 characters. If standard 80-character card images make up the input, up to 250 of them will be allowed as one entry. (This rather lavish use of core will need to be reduced on a smaller machine. It was intended to take full advantage of the core capacity and charging algorithm of the NBS computer.) Third, no more than 8000 characters may be stored in the block STR. If any of these restrictions is violated, the program stops.

The COMBO program and subroutines, with the exception of the machine-dependent input routine UNBLOK, handle characters exclusively in A1 format. The opinion held by some programmers that FORTRAN is not a suitable language for character and string manipulation seems to result largely from

the habit of storing characters in A6 format. While the "discipline" of conserving memory in this fashion was often necessary ten years ago, when core was often limited to a few thousand words, it is seldom warranted any longer, at least not on large computers. Use of A1 format for reading, storing (one character per word), and outputting frees one from reliance on assembly language and permits character manipulation in standard FORTRAN.

7. Afterword

As indicated earlier, COMBO was designed with the view of handling data files having the general structure of the file on molten salts. The degree to which we succeeded in separating the general from the particular details of that file, can be judged by the ability of this program to cope with other files of similar structure but vastly different content and format. It seems pertinent therefore to point out that neither of the files shown in Figure 6 or Figure 10 came to our attention until a year after the completion of COMBO.

COMBO was designed for applications in which its searching, string substitution, and reformatting capabilities are all used on (nearly) every card type, such as the application shown in Figures 2 and 3. We recognize that for applications such as those illustrated in Figures 5-9, in which some or all of the pieces are to be accepted unchanged or ignored, the use of COMBO may seem cumbersome, because of the elaborate setup of control cards. The number of control cards required in such applications can be profitably reduced without major reprogramming. In particular, the requirement that instructions be given to MSERCH and MXREFM for every line type, even if these routines are not being used on some of the types, can be eliminated. Such modifications in COMBO are planned for the near future. We have however chosen to document COMBO in its present preliminary form rather than further delay publication, since it and its subroutines are useful in their present form.

The authors wish to express their appreciation for the assistance of several persons in the development of the program. Professor George H. Janz of Rensselaer Polytechnic Institute provided a data tape which presented the challenge which we trust has been met by COMBO. Mrs. Carla G. Messina contributed the UNBLOK subroutine, and MSERCH is based on her SEARCH program. Special thanks are due Mr. Robert C. Thompson, both for his assistance in modifying his multiple table substitution program, and for assistance in the debugging of MXREFM, and for originally suggesting that COMBO be written.

We also wish to thank Mrs. D. W. Jones and Mrs. S. T. Moore for typing this report into a time-shared computer system from which camera-ready copy was produced.

1. Hilsenrath, J., and Galler, G., OMNIFORM I: A General-Purpose Machine Program for the Calculation of Tables of Functions Given Explicitly in Terms of One Variable. Nat. Bur. Stand. (U.S.), Tech. Note 125, 35 pp. (May 1962)

2. Cameron, J. M., and Hilsenrath, J. "Use of General-Purpose Coding Systems for Satistical Calculations," Proceedings of the IBM Scientific Computting Symposium on Statistics, (White Plains, New York), October 21-23, 1963, 323 pp.

3. Hilsenrath, J., Ziegler, G.G., Messina, C.G., Walsh, P.J., and Herbold, R.J., OMNITAB: A Computer Program for Statistical and Numerical Analysis. Nat. Bur. Stand. (U.S.), Handbook 101, 256 pp. (March 4, 1966); 2d ed., 275 pp. (Jan. 1968)

4. Rosenblatt, J., Joiner, B. L., and Hogben, D., "OMNITAB-Rapid Statistical Manipulation," in Final 1970 Census Plans, and Four Programming Systems For Computerized Data Retrieval and Manipulation, Census Tract Papers Series GE40, No. 6., pp. 46-59, U.S. Bureau of the Census, 61 pp. (January 1970)

5. Messina, C.G., and Hilsenrath, J., EDPAC: Utility Programs for Computer-Assisted Editing, Copy-Production, and Data Retrieval. Nat. Bur. Stand. (U.S.), Tech. Note 470, 76 pp., (Jan. 1969)

6. McClenon, R., and Hilsenrath, J., REFORM: A Computer Program for Manipulating Formatted Data Files. Nat. Bur. Stand. (U.S.), Tech. Note 444, 25 pp. (August 1968)

7. Maguire, J., and Jakes, M., "Quick Qwery," in Final 1970 Census Plans, pp. 29-33

8. Shapiro, K., "Tablemaker 2," in Final 1970 Census Plans, pp. 34-39

9. Wall, H., "GIS - Generalized Information System," in Final 1970 Census Plans

10. Messina, C.G., and Hilsenrath, J., Edit-Insertion Programs for Automatic Typesetting of Computer Printout. Nat. Bur. Stand. (U.S.), Tech. Note 500, 47 pp., (April 1970)

Appendix I

Control Card Printout

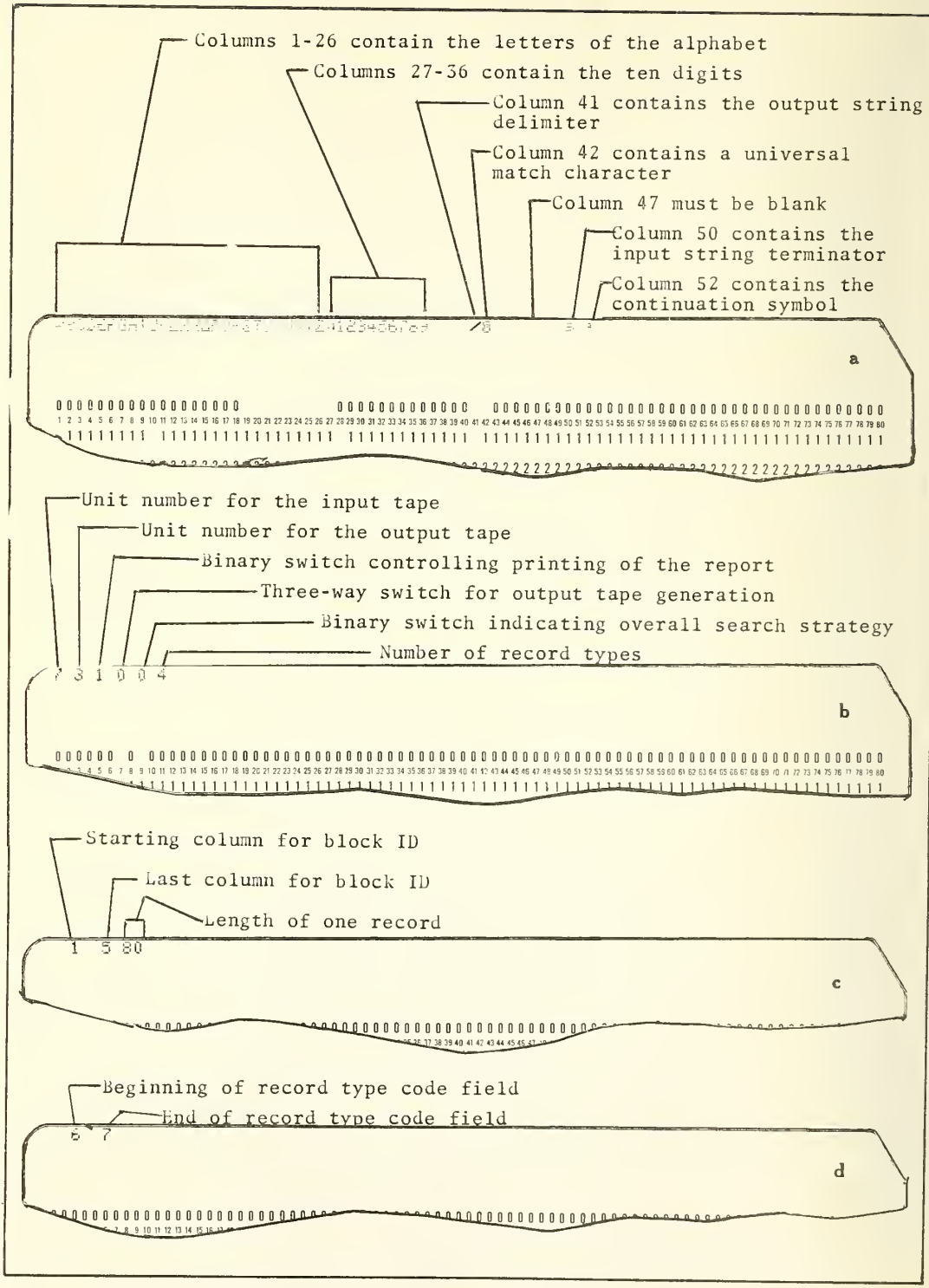
The following pages show printout produced by COMBO on reading the control cards shown in Figure 3. This printout was produced during initialization, before COMBO began reading from tape and generating the report shown in Figure 2. It generally parallels the listing of control cards (Figure 3), but is often in a less compact format, and sometimes includes comments or interpretive information.

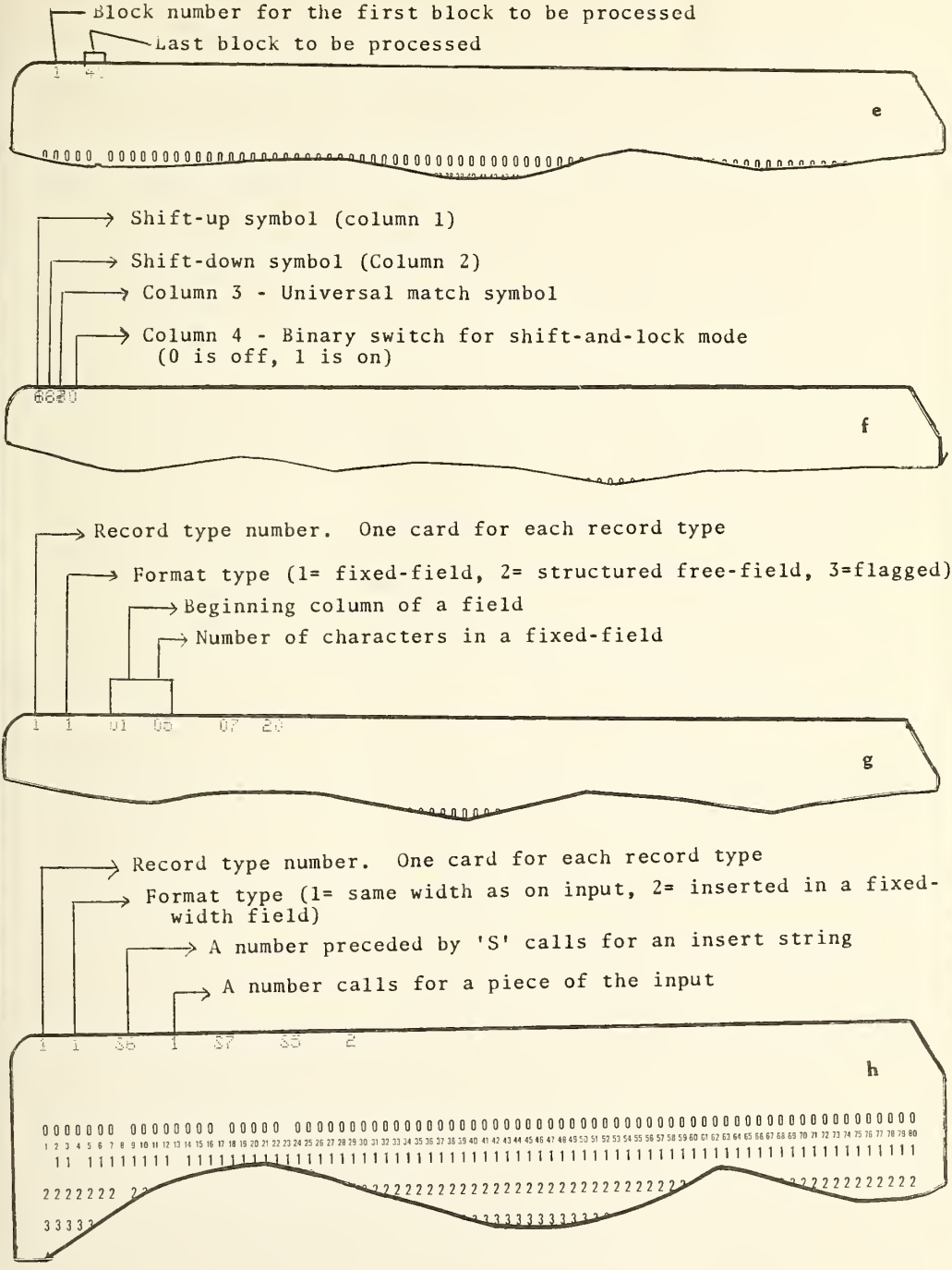
```
REAL TIME CLOCK INTERRUGATED AT 18:08:53.0759
 ABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789     /&        $ '

 INPUT INTERPRETER INITIALIZED
 ABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789     /&        $ '
 7 3 1 0 0 4
   1   5 80
   6   7
  1 08
  2 11
  3 30
  4 32
  1   40

••• SEARCH CONTROL CARDS FOLLOW •••

SEARCH TABLE  1
OR

THE PROGRAM IS SEARCHING FOR LINES CONTAINING ANY OF THE WORDS GIVEN BELOW.


DEN

SEARCH TABLE  2
OR

THE PRUGRAM IS SEARCHING FOR LINES CONTAINING ANY OF THE WORDS GIVEN BELOW.


LICL-NACL
LICL-KCL

SEARCH TABLE  3
OR

THE PROGRAM IS SEARCHING FOR LINES CONTAINING ANY OF THE WORDS GIVEN BELOW.


SEARCH TABLE  4
AND

THE PROGRAM IS SEARCHING FOR LINES CONTAINING ALL OF THE WORDS GIVEN BELOW.


KLEPPA
HERSH

THERE ARE  4 TABLES

••• SUBSTITUTE CONTROL CARDS FOLLOW •••
<>#0
<>#0
TABLE 2                                                          0 0
/(1)/                          / += /                            0 0
/(11)/                         / +=+= /                          0 0
/(111)/                        / +=+=+= /                        0 0
/DHF/                          /OHF /                            0 0
/DHSL/                         /OHSL /                           0 0
/EFEN/                         /EFEN /                           0 0
/HMX/                          /HMX /                            0 0
/TOY/                          /TDY /                            0 0
/SVEX/                         /SVEX /                           0 0
/FEN/                          /FEN /                            0 0
/RAM/                          /RAM /                            0 0
/ELEC/                         /ELEC /                           0 0
/CAL/                          /CAL /                            0 0
/HIT/                          /HIT /                            0 0
/ACTC/                         /ACTC /                           0 0
/PCEL/                         /PCEL /                           0 0
/DENT/                         /DENT /                           0 0
/DENS/                         /DENS /                           0 0
/V.P./                         /V.P. /                           0 0
/TDYP/                         /TDYP /                           0 0
/VOL/                          /VOL /                            0 0
/J. AMER. CHEM. SOC./          /J. A.C.S./                       0 0
FINIS
```

```
TABLE  1 BEGINS AT     9 ENDS AT    56 FIRST CHAR  IS    72
     19/J. AMER. CHEM. SOC./  /J. A.C.S./                                  0     0
      4/CMPF/  /CMPF /                                                     0     0
      4/DECP/  /DECP /                                                     0     0
      3/HIT/  /HIT /                                                       0     0
      3/VOL/  /VOL /                                                       0     0
      2/KA/  /KA /                                                         0     0
TABLE  2 BEGINS AT     1 ENDS AT     8 FIRST CHAR  IS    32
      5/(III)/  / +-+++- /                                                 0     0
      4/(II)/  / +-+- /                                                    0     0
      4/(UU)/  / +-+- /                                                    0     0
      3/(I)/  / +- /                                                       0     0
TABLE  4 BEGINS AT     9 ENDS AT     8 FIRST CHAR  IS    72
     19/J. AMER. CHEM. SOC./  /J. A.C.S./
REAL TIME CLOCK INTERROGATED AT 18:08:54.2089

••• REFORM CONTROL CARDS FOLLOW •••

THERE ARE  8 STRINGS.  THEY ARE --

 S  1      8 CHARACTERS SYSTEM #
 S  2      5 CHARACTERS JRNL:
 S  3      8 CHARACTERS AUTHORS:
 S  4      2 CHARACTERS  ,
 S  5      2 CHARACTERS
 S  6      2 CHARACTERS (
 S  7      2 CHARACTERS )
 S  8      4 CHARACTERS VOL.
RECORD IS FIXED-FIELD.  STARTING COLUMNS AND WIDTHS ARE --
    1  5      7 20
RECORD IS FIXED-FIELD.  STARTING COLUMNS AND WIDTHS ARE --
    1  5      8  2    10 20
RECORD IS FIXED-FIELD.  STARTING COLUMNS AND WIDTHS ARE --
    1  5      8 20    70 11
RECORD IS FIXED-FIELD.  STARTING COLUMNS AND WIDTHS ARE --
    1  5      8 15    56 15

OUTPUT INSTRUCTIONS FOR RECORD 1
STRING  6
PIECE  1 OF RECORD  1
STRING  7
STRING  5
PIECE  2 OF RECORD  1

OUTPUT INSTRUCTIONS FOR RECORD 2
STRING  6
PIECE  1 OF RECORD  2
STRING  7
STRING  5
STRING  1
PIECE  2 OF RECORD  2
STRING  5
STRING  5
PIECE  3 OF RECORD  2

OUTPUT INSTRUCTIONS FOR RECORD 3
STRING  6
PIECE  1 OF RECORD  3
STRING  7
STRING  5
STRING  2
PIECE  2 OF RECORD  3
STRING  5
STRING  8
PIECE  3 OF RECORD  3

OUTPUT INSTRUCTIONS FOR RECORD 4
STRING  6
PIECE  1 OF RECORD  4
STRING  7
STRING  5
STRING  3
PIECE  2 OF RECORD  4
STRING  4
PIECE  3 OF RECORD  4
STRING  4
PIECE  4 OF RECORD  4
REAL TIME CLOCK INTERROGATED AT 18:08:54.4579

••• BEGIN RETRIEVAL
```

Appendix II

Annotated Control Cards


Columns 1-26 contain the letters of the alphabet
Columns 27-36 contain the ten digits
Column 41 contains the output string delimiter
Column 42 contains a universal match character
Column 47 must be blank
Column 50 contains the input string terminator
Column 52 contains the continuation symbol
a
Unit number for the input tape
Unit number for the output tape
Binary switch controlling printing of the report
Three-way switch for output tape generation
Binary switch indicating overall search strategy
Number of record types
7 3 1 0 0 4
b
Starting column for block ID
Last column for block ID
Length of one record
1 5 80
c
Beginning of record type code field
End of record type code field
6 7
d


Block number for the first block to be processed
Last block to be processed
e
Shift-up symbol (column 1)
Shift-down symbol (Column 2)
Column 3 - Universal match symbol
Column 4 - Binary switch for shift-and-lock mode
(0 is off, 1 is on)
f
Record type number. One card for each record type
Format type (1= fixed-field, 2= structured free-field, 3=flagged)
Beginning column of a field
Number of characters in a fixed-field
g
Record type number. One card for each record type
Format type (1= same width as on input, 2= inserted in a fixed-width field)
A number preceded by 'S' calls for an insert string
A number calls for a piece of the input
h

Appendix III

Program Listings

```
C     MAIN PROGRAM FOR READING THE RPI DATA TAPE AND PRINTING REPORTS   COM   5
C     FROM IT                                                           COM  10
C                                                                       COM  15
C     THIS PROGRAM USES FOR INPUT A TAPE CONTAINING BLOCKED CARD IMAGES COM  20
C     CONTAINING A BLOCK ID IN A FIXED POSITION AND A CODE IDENTIFYING  COM  25
C     THE TYPE OF INFORMATION OF EACH CARD IMAGE IN ANOTHER FIXED       COM  30
C     POSITION.                                                         COM  35
C     THIS PROGRAM UNBLOCKS THE TAPE INTO CARD IMAGES AND COLLECTS THEM COM  40
C     IN A BUFFER UNTIL THE END OF A UNIT OF INFORMATION IS SENSED.     COM  45
C     NOTE -- THE INFORMATION IS UNBLOCKED FROM THE FIXED-LENGTH        COM  50
C     PHYSICAL BLOCKS IN WHICH IT IS WRITTEN ON THE TAPE, BUT IS THEN   COM  55
C     REBLOCKED INTO LOGICAL BLOCKS, IDENTIFIED BY THE ID NUMBER IN A   COM  60
C     FIXED POSITION ON THE CARD.                                       COM  65
C     THEN IT SEARCHES THE BLOCK LINE-FOR-LINE FOR THE OCCURRENCE OF    COM  70
C     SPECIFIED STRINGS (A SEPARATE SEARCH TABLE BEING USED FOR EACH    COM  75
C     TYPE OF LINE).  IF THE SEARCH CRITERIA ARE MET, IT REFORMATS EACH COM  80
C     LINE AND PERFORMS INDICATED SUBSTITUTIONS ON IT.  THE EDITED LINE COM  85
C     IS PRINTED, THIS PROCEDURE IS FOLLOWED FOR THE NEXT LINE IN THE   COM  90
C     BLOCK, AND THE PROGRAM PROCEEDS TO UNBLOCKING AND REBLOCKING      COM  95
C     ANOTHER UNIT OF INFORMATION.                                      COM 100
C                                                                       COM 105
C     CONTROL CARDS READ BY THE MAIN PROGRAM ARE AS FOLLOWS --          COM 110
C     THE FIRST CARD CONTAINS THE ALPHABET IN COLUMNS 1 THRU 26.  A     COM 115
C     CHARACTER TO BE READ AS THE UNIVERSAL MATCH SYMBOL BY LINTYP IS INCOM 120
C     COLUMN 42.                                                        COM 125
C     THE SECOND CONTROL CARD READ BY THE MAIN PROGRAM IS IN I3 FORMAT. COM 130
C     IT CONTAINS THE NUMBER FOR THE INPUT TAPE, THE NUMBER FOR THE     COM 135
C     OUTPUT TAPE OR CARD PUNCH, IF OUTPUT IN ADDITION TO THE PRINTOUT  COM 140
C     IS DESIRED, AND TWO SWITCHES INDICATING WHETHER THE EDITED OUTPUT COM 145
C     FROM THE PROGRAM IS TO BE PRINTED AND WHETHER IT IS TO BE WRITTEN COM 150
C     ON AN OUTPUT UNIT.  THESE NUMBERS ARE 0 FOR NO AND 1 FOR YES.  THECOM 155
C     FIFTH NUMBER IS 0 TO INDICATE AN 'OR' SEARCH IN WHICH THE PROGRAM COM 160
C     WILL SEARCH FOR BLOCKS IN WHICH AT LEAST ONE RECORD SATISFIES THE COM 165
C     SEARCH CRITERIA.  IT IS 1 FOR AN 'AND' SEARCH IN WHICH THE PROGRAMCOM 170
C     SEARCHES FOR BLOCKS IN WHICH AT LEAST ONE RECORD OF EACH TYPE     COM 175
C     MEETS THE CRITERIA.                                               COM 180
C     THE NEXT CONTROL CARD IS READ BY BSENS.  IT GIVES THE FIRST AND   COM 185
C     LAST CARD COLUMNS IN WHICH TO LOOK FOR THE BLOCK ID NUMBER.       COM 190
C     THE CONTROL CARDS FOR LINTYP COME NEXT.                           COM 195
C     THEN FOLLOW CONTROL CARDS FOR THE SEARCH, SUBSTITUTION, AND       COM 200
C     REFORMATTING ROUTINES, IN THAT ORDER.  WHEN ALL CONTROL CARDS HAVECOM 205
C     BEEN READ, THE PROGRAM BEGINS READING FROM THE INPUT TAPE AND     COM 210
C     PROCESSING THE INPUT RECORDS.                                     COM 215
C                                                                       COM 220
C     THIS PROGRAM WRITTEN BY R. MCCLENON, NSRDS-NBS, FEB. 1970.        COM 225
C                                                                       COM 230
      DIMENSION LINE(400),IBLOK(8000),KARD(250),IABC(120),KBUF(250)     COM 235
      DIMENSION KSRCH(99),JDATA(60),KEY(60)                             COM 240
C     IN AND IOUT ARE THE LOGICAL UNIT NUMBERS FOR THE CARD READER AND  COM 245
C     PRINTER, RESPECTIVELY.  THEY MUST BE SET IN THE MAIN PROGRAM TO   COM 250
C     THE APPROPRIATE VALUES FOR THE INSTALLATION.                      COM 255
      COMMON /IO/ IN,IOUT                                               COM 260
      COMMON /IA/ IABC                                                  COM 265
      COMMON /IB/ MAX,IBLOK                                             COM 270
C     STR IS A COMMON BLOCK CONTAINING STRING BUFFER IC                 COM 275
C     IC MAY BE USED FOR THE STORAGE OF STRING DATA BY MORE THAN ONE    COM 280
C     SUBROUTINE SIMULTANEOUSLY.  ICLEN IS THE LENGTH OF IC.  ICNOW IS ACOM 285
C     POINTER INDICATING HOW MUCH OF IC HAS BEEN USED.  IT POINTS TO THECOM 290
C     NEXT AVAILABLE CELL IN IC.  ANY SUBROUTINE USING IC MUST RESET    COM 295
C     ICNOW TO INDICATE ITS USE OF IC, AND MUST SAVE A POINTER TELLING  COM 300
C     WHERE IN IC IT BEGAN STORAGE OF STRINGS.                          COM 305
      COMMON /STR/ ICLEN,ICNOW,IC(8000)                                 COM 310
      COMMON /W/ IW,NTABL                                               COM 315
      COMMON /COUNT/ NPHYS,NCARD,NBLOK                                  COM 320
C     ** THE NEXT TWO STATEMENTS ARE INSTALLATION-DEPENDENT **          COM 325
      IN=5                                                              COM 330
      IOUT=6                                                            COM 335
```

```
C     **                                                              COM 340
      ICLEN=8000                                                      COM 345
      ICNOW=1                                                         COM 350
      CALL CLOCK                                                      COM 355
      READ (IN,380) (IABC(J),J=1,80)                                  COM 360
      WRITE (IOUT,370) (IABC(J),J=1,80)                               COM 365
C      KRDLEN IS THE LENGTH OF KARD OR KBUF, WHICHEVER IS SHORTER     COM 370
      KRDLEN=200                                                      COM 375
      CALL INPUT (0,IABC,JDATA,KEY,MX)                                COM 380
      IF (MX.LT.-1) GO TO 360                                         COM 385
      READ (IN,380) (KARD(J),J=1,80)                                  COM 390
      WRITE (IOUT,370) (KARD(J),J=1,80)                               COM 395
      CALL INPUT (4,KARD,JDATA,KEY,80)                                COM 400
      ITAPE=JDATA(1)                                                  COM 405
      JTAPE=JDATA(2)                                                  COM 410
      ITST1=JDATA(3)                                                  COM 415
      ITST2=JDATA(4)                                                  COM 420
      ISRCH=JDATA(5)                                                  COM 425
      NTABL=JDATA(6)                                                  COM 430
      ISER=JDATA(7)                                                   COM 435
      ISUB=JDATA(8)                                                   COM 440
      IREF=JDATA(9)                                                   COM 445
      IF (NTABL.LT.1) GO TO 360                                       COM 450
      NPHYS=0                                                         COM 455
      NCARD=0                                                         COM 460
      NBLOK=0                                                         COM 465
C     INITIALIZATION OF SUBROUTINES                                   COM 470
      CALL BSENS (IABC,0)                                             COM 475
      CALL LINTYP (IBLOK,I1,0,M)                                      COM 480
      CALL TAPOUT (0,JDATA,LL)                                        COM 485
      READ (IN,380) (KARD(J),J=1,80)                                  COM 490
      WRITE (IOUT,370) (KARD(J),J=1,80)                               COM 495
      CALL INPUT (4,KARD,JDATA,KEY,80)                                COM 500
      ISTART=JDATA(1)                                                 COM 505
      IFIN=JDATA(2)                                                   COM 510
      IF (IFIN.EQ.0) IFIN=99999                                       COM 515
      IF (IFIN.LT.ISTART) GO TO 360                                   COM 520
      IF (ISER.GT.0) GO TO 10                                         COM 525
      WRITE (IOUT,400)                                                COM 530
      CALL MSERCH (IBLOK,80,0)                                        COM 535
      GO TO 20                                                        COM 540
10    WRITE (IOUT,470)                                                COM 545
20    IF (ISUB.GT.0) GO TO 30                                         COM 550
      WRITE (IOUT,410)                                                COM 555
      CALL MSUBS (LINE,300,0)                                         COM 560
      GO TO 40                                                        COM 565
30    WRITE (IOUT,480)                                                COM 570
40    IF (IREF.GT.0) GO TO 50                                         COM 575
      WRITE (IOUT,420)                                                COM 580
      CALL MXREFM (0,KBUF,KARD,KRDLEN)                                COM 585
      GO TO 60                                                        COM 590
50    WRITE (IOUT,490)                                                COM 595
60    CALL CLOCK                                                      COM 600
      WRITE (IOUT,390)                                                COM 605
      L=0                                                             COM 610
C     READ A CARD IMAGE FROM THE TAPE                                 COM 615
70    CALL UNBLOK (ITAPE,LINE,IW,1R ,L,IE)                            COM 620
      NCARD=NCARD+1                                                   COM 625
      DO 80 J=1,NTABL                                                 COM 630
80    KSRCH(J)=0                                                      COM 635
C     CHECK TAPE STATUS - IF ABNORMAL, GO TO END-ACTION               COM 640
      IF (IE) 300,90,90                                               COM 645
90    K=1                                                             COM 650
C     SEND LINE TO BSENS TO SEE IF BLOCK IS FINISHED                  COM 655
      CALL BSENS (LINE,K)                                             COM 660
      IF (K) 70,70,100                                                COM 665
C     NEW BLOCK - K IS NUMBER OF CHARACTERS - NL IS NUMBER OF LINES   COM 670
100   NL=K/IW                                                         COM 675
```

```
      NBLØK=NBLØK+1                                                    CØM 680
      IF (NBLØK.LT.ISTART) GØ TØ 70                                    CØM 685
      IF (NBLØK.GT.IFIN) GØ TØ 320                                     CØM 690
      IF (ISER.GT.0) GØ TØ 180                                         CØM 695
C     EXAMINE ØNE LINE AT A TIME                                       CØM 700
      DØ 150 J=1,NL                                                    CØM 705
      I2=IW*J                                                          CØM 710
      I1=I2+1-IW                                                       CØM 715
C     SEE WHICH TYPE LINE IS                                           CØM 720
      CALL LINTYP (IBLØK,I1,IW,M)                                      CØM 725
      MM=M                                                             CØM 730
      IF (M) 150,150,110                                               CØM 735
C     VALID TYPE                                                       CØM 740
110   DØ 120 I=I1,I2                                                   CØM 745
      JJ=I-I1+1                                                        CØM 750
C     TRANSFER LINE TØ SEARCH AND SUBSTITUTIØN BUFFER                  CØM 755
120   KBUF(JJ)=IBLØK(I)                                                CØM 760
      CALL MSERCH (KBUF,IW,M)                                          CØM 765
      IF (M) 150,150,130                                               CØM 770
C     SEARCB SUCCESSFUL                                                CØM 775
130   IF (ISRCH) 180,180,140                                           CØM 780
C     IN AND-MØDE - CØNTINUE SEARCHING AFTER MARKING SUCCESS           CØM 785
140   KSRCH(MM)=1                                                      CØM 790
150   CØNTINUE                                                         CØM 795
      IF (ISRCH) 70,70,160                                             CØM 800
C     SEE IF AND-SEARCH SATISFIED FØR EACH TABLE                       CØM 805
160   DØ 170 J=1,NTABL                                                 CØM 810
      IF (KSRCH(J)) 70,70,170                                          CØM 815
170   CØNTINUE                                                         CØM 820
C     SEARCB SATISFIED                                                 CØM 825
180   DØ 290 J=1,NL                                                    CØM 830
      I2=IW*J                                                          CØM 835
      I1=I2+1-IW                                                       CØM 840
      CALL LINTYP (IBLØK,I1,IW,M)                                      CØM 845
      IF (M) 190,190,200                                               CØM 850
190   WRITE (IØUT,430) (IBLØK(I),I=I1,I2)                              CØM 855
      GØ TØ 290                                                        CØM 860
200   DØ 210 I=I1,I2                                                   CØM 865
      JJ=I-I1+1                                                        CØM 870
210   KBUF(JJ)=IBLØK(I)                                                CØM 875
C     SEND EACH LINE TØ SUBSTITUTIØN AND REFØRMATTING                  CØM 880
      LL=IW                                                            CØM 885
      IF (IREF.GT.0) GØ TØ 220                                         CØM 890
      CALL MXREFM (M,KBUF,KARD,LL)                                     CØM 895
      IF (LL.LE.0) GØ TØ 290                                           CØM 900
      GØ TØ 240                                                        CØM 905
220   DØ 230 I=1,IW                                                    CØM 910
230   KARD(I)=KBUF(I)                                                  CØM 915
240   IF (ISUB.GT.0) GØ TØ 250                                         CØM 920
      CALL MSUBS (KARD,LL,M)                                           CØM 925
250   CØNTINUE                                                         CØM 930
      IF (ITST1) 260,270,260                                           CØM 935
260   WRITE (IØUT,370) (KARD(I),I=1,LL)                                CØM 940
270   IF (ITST2) 280,290,280                                           CØM 945
280   CALL TAPØUT (M,KARD,LL)                                          CØM 950
290   CØNTINUE                                                         CØM 955
      WRITE (IØUT,440)                                                 CØM 960
C     GET NEXT BLØCK                                                   CØM 965
      GØ TØ 70                                                         CØM 970
C     TBAT WRAPS IT UP                                                 CØM 975
300   IF (IE+3) 340,340,310                                            CØM 980
310   WRITE (IØUT,450)                                                 CØM 985
320   IF (ITST2) 330,350,330                                           CØM 990
330   CALL TAPØUT (-1,KARD,LL)                                         CØM 995
      GØ TØ 350                                                        CØM1000
340   WRITE (IØUT,460) IE                                              CØM1005
350   STØP                                                             CØM1010
360   WRITE (IØUT,500)                                                 CØM1015
```

```
      GO TO 350                                                          COM1020
                                                                         COM1025
C                                                                        COM1030
370   FORMAT (1X,120A1)                                                  COM1035
380   FORMAT (120A1)                                                     COM1040
390   FORMAT (21HO*** BEGIN RETRIEVAL /1H1)                              COM1045
400   FORMAT (37HO*** SEARCH CONTROL CARDS FOLLOW *** )                  COM1050
410   FORMAT (41HO*** SUBSTITUTE CONTROL CARDS FOLLOW *** )              COM1055
420   FORMAT (37HO*** REFORM CONTROL CARDS FOLLOW *** )                  COM1060
430   FORMAT (26HOBAD LINE TYPE CODE BELOW /1X,80A1)                     COM1065
440   FORMAT (1X)                                                        COM1070
450   FORMAT (20HOEND-OF-FILE.  STOP.)                                   COM1075
460   FORMAT (22HONTRAN ERROR , STATUS ,I2,8H. STOP. )                   COM1080
470   FORMAT (45BO *** SEARCHING OMITTED - NO SEARCB CARDS ** )          COM1085
480   FORMAT (52HO *** SUBSTITUTION OMITTED - NO SUBSTITUTE CARDS ** )   COM1090
490   FORMAT (48BO *** REFORMATTING OMITTED - NO REFORM CARDS ** )       COM1095
500   FORMAT (51BO *** BAD PARAMETER OR MISFORMATTED CARD. STOP. *  )    COM1100-
      END

      SUBROUTINE BSENS (LINE,L)                                          BSN   5
C           BSENS                                                        BSN  10
C                                                                        BSN  15
C     BSENS DETECTS THE END OF A BLOCK OF INFORMATION IN A FILE WBERE    BSN  20
C     SEVERAL CARD IMAGES BAVING TBE SAME BLOCK IDENTIFICATION IN A      BSN  25
C     FIXED POSITION MAKE UP ONE BLOCK.                                  BSN  30
C     BSENS IS CALLED WITB ONE CARD IMAGE AT A TIME IN LINE.  IT         BSN  35
C     COLLECTS THEM IN IBLOK UNTIL THE BEGINNING OF A NEW BLOCK IS       BSN  40
C     DETECTED, WHEN IT SIGNALS THIS TO THE MAIN PROGRAM AND RETURNS     BSN  45
C     BLOCK.                                                             BSN  50
C                                                                        BSN  55
C     BSENS MUST BE INITIALIZED BY CALLING IT WITH L=0.  THIS WILL CAUSEBSN  60
C     IT TO READ A CONTROL CARD FROM UNIT IN AND TO SET CERTAIN SWITCBESBSN  65
C     THE CONTROL CARD CONTAINS IN I3 FORMAT THE STARTING AND ENDING     BSN  70
C     COLUMNS FOR THE BLOCK ID IN A RECORD AND TBE WIDTB OF A RECORD.    BSN  75
C     IF THE RECORDS ARE CARD IMAGES THE LAST NUMBER WILL BE 80.         BSN  80
C                                                                        BSN  85
C     LINE - THE VECTOR IN WHICH ONE CARD IMAGE OR RECORD AT A TIME IS   BSN  90
C     TRANSMITTED TO BSENS                                               BSN  95
C     L - IN THE CALL SET TO 0 TO INITIALIZE BSENS, OTBERWISE TO A       BSN 100
C     POSITIVE NUMBER.  ON RETURN SET TO ZERO FOR END OF BLOCK NOT FOUNDBSN 105
C     OR TO TBE LENGTB OF TBE BLOCK ( IN CBARACTERS) IF TBE END IS       BSN 110
C     SENSED                                                             BSN 115
C     IBLOK - TBE ARRAY IN WHICB A BLOCK IS BUILT UP FROM THE LINES      BSN 120
C     ALL TBE RECORDS IN IBLOK WITB THE SAME ID WILL BE RETURNED TO TBE  BSN 125
C     CALLING PROGRAM WHEN THE END IS SENSED                             BSN 130
C     IABC - CONTAINS TBE STANDARD ALPBABETIC DICTIONARY CARD            BSN 135
C     IN AND IOUT - THE LOGICAL UNIT NUMBERS FOR CARD READER AND PRINTERBSN 140
C                                                                        BSN 145
C     TBIS ROUTINE WRITTEN BY R. MCCLENON , NSRDS-NBS, FEB. 1970         BSN 150
C                                                                        BSN 155
      DIMENSION LINE(250),IBLOK(8000),IBUF(250),N(250)                   BSN 160
      DIMENSION IABC(120),JDATA(60),KEY(60)                              BSN 165
      COMMON /IO/ IN,IOUT                                                BSN 170
      COMMON /IB/ MAX,IBLOK                                              BSN 175
      COMMON /IA/ IABC                                                   BSN 180
      COMMON /W/ KL,NTABL                                                BSN 185
      COMMON /COUNT/ NPBYS,NCARD,NBLOK                                   BSN 190
10    FORMAT (80A1)                                                      BSN 195
20    FORMAT (1X,80A1)                                                   BSN 200
      IF (L) 30,30,70                                                    BSN 205
C     INITIALIZATION                                                     BSN 210
30    READ (IN,10) (IBUF(J),J=1,80)                                      BSN 215
      WRITE (IOUT,20) (IBUF(J),J=1,80)                                   BSN 220
      KLMAX=250                                                          BSN 225
      IDLEN=10                                                           BSN 230
      CALL INPUT (4,IBUF,JDATA,KEY,80)                                   BSN 235
      K1=JDATA(1)                                                        BSN 240
```

```
      K2=JDATA(2)                                                    BSN 245
      KL=JDATA(3)                                                    BSN 250
      KW=K2-K1+1                                                     BSN 255
      IF (KW-IDLEN) 40,40,240                                        BSN 260
40    DO 50 J=1,KW                                                   BSN 265
50    N(J)=IABC(47)                                                  BSN 270
      IF (KL-KLMAX) 60,60,250                                        BSN 275
60    IX=-1                                                          BSN 280
      I=0                                                            BSN 285
      MAX=8000                                                       BSN 290
      RETURN                                                         BSN 295
C     CBECK NEXT RECORD                                              BSN 300
70    I1=I+1                                                         BSN 305
      I2=I+KL                                                        BSN 310
      IF (I2-MAX) 80,80,200                                          BSN 315
80    IF (IX) 170,110,90                                             BSN 320
90    DO 100 J=1,KL                                                  BSN 325
C     RECOVER A RECORD FROM TEMPORARY STORAGE IN IBUF                BSN 330
100   IBLOK(J)=IBUF(J)                                               BSN 335
      IX=0                                                           BSN 340
110   DO 120 J=I1,I2                                                 BSN 345
      II=J-I1+1                                                      BSN 350
C     TRANSFER TBE RECORD TO IBLOK                                   BSN 355
120   IBLOK(J)=LINE(II)                                              BSN 360
      I=I+KL                                                         BSN 365
      I3=I1+K1-1                                                     BSN 370
      I4=I1+K2-1                                                     BSN 375
      DO 130 J=I3,I4                                                 BSN 380
      JJ=J-I3+1                                                      BSN 385
C     COMPARE THE ID FIELD AGAINST THAT OF THE LAST RECORD           BSN 390
      IF (IBLOK(J)-N(JJ)) 140,130,140                                BSN 395
130   CONTINUE                                                       BSN 400
      GO TO 190                                                      BSN 405
C     FOUND START OF NEW RECORD                                      BSN 410
140   DO 150 J=1,KW                                                  BSN 415
      JJ=J+I3-1                                                      BSN 420
C     SAVE ID AND SET SIGNAL TO MAIN PROGRAM                         BSN 425
150   N(J)=IBLOK(JJ)                                                 BSN 430
      L=I1-1                                                         BSN 435
      DO 160 J=I1,I2                                                 BSN 440
      JJ=J-I1+1                                                      BSN 445
C     STORE TBE BEGINNING OF TBE NEXT RECORD IN IBUF                 BSN 450
160   IBUF(JJ)=IBLOK(J)                                              BSN 455
      I=KL                                                           BSN 460
      IX=1                                                           BSN 465
      RETURN                                                         BSN 470
170   I3 = I1+K1-1                                                   BSN 475
      DO 180 J=1,KW                                                  BSN 477
      JJ=J+I3-1                                                      BSN 480
180   N(J) = LINE(JJ)                                                BSN 482
      DO 185 J=1,KL                                                  BSN 484
185   IBLOK(J) = LINE(J)                                             BSN 486
      I = KL                                                         BSN 488
      IX=0                                                           BSN 490
190   L=0                                                            BSN 495
      RETURN                                                         BSN 500
200   WRITE (IOUT,210) MAX                                           BSN 505
210   FORMAT (37HOLOGICAL BLOCK IS TOO LONG.  LIMIT IS,I6,9H.  STOP. /52BSN 510
     1H THE FIRST AND LAST CARD IMAGES IN TBE BUFFER ARE --)         BSN 515
220   FORMAT (1X,126A1)                                              BSN 520
      WRITE (IOUT,220) (IBLOK(J),J=1,KL)                             BSN 525
      WRITE (IOUT,220) (LINE(J),J=1,KL)                              BSN 530
      STOP                                                           BSN 535
230   FORMAT (31HOID FIELD WIDTB MAY NOT EXCEED ,I3)                 BSN 540
240   WRITE (IOUT,230) IDLEN                                         BSN 545
      KW=IDLEN                                                       BSN 550
      GO TO 40                                                       BSN 555
```

```
250   WRITE (IOUT,260) KLMAX                                          HSN 560
260   FORMAT (41HOHSENS RESTRICTS THE CARD IMAGE WIDTH TO ,I4)        HSN 565
      KL=KLMAX                                                        BSN 570
      GO TO 60                                                        HSN 575
      END                                                             HSN 580-

      SUBROUTINE UNHLOK (IRTAPE,ID,LENGTH,IBLANK,K,IEND)              UNB   5
C                        UNHLOK                                       UNB  10
C                                                                     UNB  15
C     UNBLOK UNBLOCKS HCD RECORDS AND RETURNS THE ORIGINAL RECORD IN  UNB  20
C     PIECES (LINES) OF SPECIFIED LENGTH IN A1 FORMAT.  IT SERVES THE UNB  25
C     SAME PURPOSE THAT AN A1 READ STATEMENT DOES FOR UNBLOCKED RECORDS.UNB 30
C                                                                     UNB  35
C IRTAPE   IS THE UNIT ON WHICH TO EXPECT THE BLOCKED INFORMATION.    UNB  40
C ID()     IS THE VARIABLE IN WHICH UNBLKS PLACES THE CURRENT LINE.   UNB  45
C LENGTH   IS THE NUMBER OF CHARACTERS TO BE RETURNED PER LINE.  IF THE UNB 50
C          LAST PIECE OF THE ORIGINAL RECORD IS SMALLER THAN 'LENGTH'  UNB  55
C          THE REST OF ID() IS FILLED OUT WITH 'IBLANK'.               UNB  60
C          USUALLY A BLANK.  IF COMPARISONS ARE TO BE MADE WITH        UNB  65
C          CHARACTERS READ IN UNDER A1 FORMAT, IBLANK MUST ALSO HAVE   UNB  70
C          BEEN READ IN UNDER A1 FORMAT.                               UNB  75
C     IEND IS SET TO A NON-ZERO INTEGER WHEN THE TAPE CANNOT BE READ FORUNB 80
C          ANY REASON.  IT IS SET TO -3 OR -4 ON A TAPE READ ERROR AND UNB  85
C          TO -2 ON REACHING AN END OF FILE.                           UNB  90
C     K IS A COUNTER WHICH MUST BE A VARIABLE IN THE CALL STATEMENT.   UNB  95
C          IT IS THE CURRENT LINE BEING REQUESTED MINUS ONE.  IN THE   UNB 100
C          FIRST CALL TO UNBLKS, 'K' MUST BE SET TO ZERO.  THE         UNB 105
C          SUBROUTINE WILL ITSELF UP-DATE  'K'  AND RESET IT TO ZERO   UNB 110
C          AT THE END OF EACH BLOCKED RECORD.                          UNB 115
      DIMENSION ID(136),ISTRIN(4500),IB(750)                          UNB 120
      IEND=0                                                          UNB 125
C     IF K IS ZERO, A NEW RECORD MUST BE READ                         UNB 130
      IF (K) 10,10,80                                                 UNB 135
10    K=0                                                             UNB 140
      CALL NTRAN (IRTAPE,2,750,IB,L)                                  UNB 145
20    IF (L+1) 30,20,50                                               UNB 150
C     ABNORMAL TERMINATION                                            UNB 155
30    IEND=L                                                          UNB 160
40    RETURN                                                          UNB 165
50    ICBAR=6*L                                                       UNB 170
C     UNPACK CBARACTERS                                               UNB 175
      DO 60 I=1,ICHAR                                                 UNB 180
60    ISTRIN(I)=IBLANK                                                UNB 185
      DO 70 I=1,ICBAR                                                 UNB 190
      J=I-((I-1)/6)*6                                                 UNB 195
      IZ=(I-1)/6+1                                                    UNB 200
      FLD(0,6,ISTRIN(I))=FLD(6*(J-1),6,IB(IZ))                        UNB 205
70    CONTINUE                                                        UNB 210
C     SET POINTERS                                                    UNB 215
80    L1=K*LENGTH+1                                                   UNB 220
      L2=K*LENGTH+LENGTH                                              UNB 225
      I=0                                                             UNB 230
      IF (L1-ICBAR) 90,90,10                                          UNB 235
90    IF (L2-ICHAR) 130,120,100                                       UNB 240
100   IZ=ICHAR+1                                                      UNB 245
      DO 110 J=IZ,L2                                                  UNB 250
110   ISTRIN(J)=IBLANK                                                UNB 255
120   K=-1                                                            UNB 260
C     TRANSFER CBARACTERS                                             UNB 265
130   DO 140 J=L1,L2                                                  UNB 270
      I=I+1                                                           UNB 275
140   ID(I)=ISTRIN(J)                                                 UNB 280
      K=K+1                                                           UNB 285
      GO TO 40                                                        UNB 290
      END                                                             UNB 295-
```

```
C          FREE-FIELD INPUT ROUTINE                                      IN    5
C                                                                        IN   10
      SUBROUTINE INPUT (M,LINE,N,IT,MM)                                  IN   15
C                                                                        IN   20
C     THIS IS A FREE-FIELD INPUT ROUTINE FOR USE WITH REFORM AND RELATEDIN   25
C     PROGRAMS.  IT IS CALLED WITH A LINE OF 80 CHARACTERS IN A1 FORMAT.IN   30
C     IT ANALYZES THE LINE, DETERMINES WHETHER EACH NON-BLANK ITEM IS A IN   35
C     CHARACTER STRING OR AN INTEGER, AND RETURNS THE INFORMATION IN     IN   40
C     ARRAY N.  THE TYPE OF EACH ITEM IS INDICATED BY THE SETTING OF IT,IN   45
C     WHICH IS 1 FOR AN INTEGER AND 2 FOR A STRING CHARACTER.  A ZERO    IN   50
C     VALUE IN IT INDICATES THAT THE LINE HAS BEEN EXHAUSTED, A -1 THAT IN   55
C     A CONTINUATION SYMBOL HAS BEEN READ.                               IN   60
C     THE SWITCH M INDICATES WHICH OF SEVERAL MODES ARE TO BE USED TO    IN   65
C     CONVERT THE LINE.  THERE ARE AT PRESENT FOUR POSSIBLE MODES.       IN   70
C     M = 1 ALL CHARACTERS ARE TO BE RETURNED TO THE CALLING PROGRAM     IN   75
C           REGARDLESS OF WHETHER THEY ARE LETTERS OR PUNCTUATION.       IN   80
C           CONTIGUOUS DIGITS ARE TO BE TREATED AS DISTINCT ONE-DIGIT    IN   85
C           INTEGERS.                                                    IN   90
C     M = 2 PUNCTUATION MARKS ARE TO BE RETURNED ON THE SAME BASIS AS    IN   95
C           ALPHABETIC CHARACTERS.  CONTIGUOUS DIGITS ARE TO BE TREATED IN  100
C           AS PART OF A SINGLE INTEGER.                                 IN  105
C     M = 3 NON-NUMERIC CHARACTERS OTHER THAN LETTERS OF THE ALPHABET    IN  110
C           (I.E., PUNCTUATION) ARE TO BE IGNORED.  ADJOINING DIGITS     IN  115
C           WILL BE TREATED AS PART OF A SINGLE INTEGER.                 IN  120
C     M = 4 ALL NON-NUMERIC CHARACTERS WILL BE TREATED AS NOISE AND      IN  125
C           IGNORED, INCLUDING ALPHABETIC LETTERS.  CONTIGUOUS DIGITS    IN  130
C           WILL BE TREATED AS PART OF A SINGLE INTEGER.                 IN  135
C     OTHER MODES CAN BE PROVIDED BY MINOR REPROGRAMMING.                IN  140
C                                                                        IN  145
C     IF INPUTS IS CALLED WITH M=-MX, MM WILL BE SET TO THE POSITION OR IN  150
C     CARD COLUMN IN WHICH N(M) STARTED ON THE PREVIOUS CARD READ.       IN  155
C                                                                        IN  160
C     THIS SUBROUTINE MUST BE INITIALIZED BY CALLING IT WITH M=0 AND THEIN  165
C     EDPAC DICTIONARY CARD IN THE ARRAY LINE.  THIS CARD CONTAINS THE   IN  170
C     NUMBERS IN COLUMNS 27-36.  COLUMN 47 MUST BE BLANK.  THE CHARACTERIN  175
C     IN COLUMN 50 WILL BE TREATED AS A SCAN TERMINATOR -- ANY           IN  180
C     INFORMATION FOLLOWING IT WILL BE IGNORED ON A CARD.  THE           IN  185
C     CONTINUATION SYMBOL WILL BE FOUND IN COLUMN 52.                    IN  190
C     THIS CARD IS STORED DURING THE INITIALIZATION AND DEFINES THE BCD  IN  195
C     REPRESENTATIONS OF EACH OF THE CHARACTERS.                         IN  200
C     THIS SUBROUTINE WRITTEN BY R. MCCLENON, NSRDS-NBS, SEPTEMBER 1968,IN  205
C     AND REWRITTEN AUGUST 1969.                                         IN  210
C                                                                        IN  215
C     * * * * * * *  FREE-FIELD INPUT ROUTINE  * * * * * * * * * * * *  IN  220
C                                                                        IN  225
C                                                                        IN  230
      DIMENSION NUM(10),KA(26),LOC(61)                                   IN  235
      DIMENSION LINE(80),N(60),IT(60)                                    IN  240
10    FORMAT (31HOINPUT INTERPRETER INITIALIZED )                        IN  245
20    FORMAT (1X,80A1)                                                   IN  250
30    FORMAT (25HOSCAN TERMINATOR OMITTED )                              IN  255
40    FORMAT (29HOCONTINUATION SYMBOL OMITTED )                          IN  260
50    FORMAT (6HOMODE ,I4,32H REQUESTED IN INPUTS IS UNKNOWN )           IN  265
60    FORMAT (35HO INPUTS CONFUSED, CANNOT CONTINUE )                    IN  270
70    FORMAT (33HOINPUTS NOT PROPERLY INITIALIZED )                      IN  275
      IOUT=6                                                             IN  280
C     CHECK THE MODE                                                     IN  285
      IF (M) 160,80,230                                                  IN  290
C     INITIALIZATION                                                     IN  295
80    WRITE (IOUT,10)                                                    IN  300
      WRITE (IOUT,20) (LINE(JX),JX=1,80)                                 IN  305
      KBL=LINE(47)                                                       IN  310
      MM=0                                                               IN  315
C     STORE THE NUMBERS AND LETTERS APPROPRIATELY                        IN  320
      DO 90 J=1,26                                                       IN  325
90    KA(J)=LINE(J)                                                      IN  330
      DO 100 J=1,10                                                      IN  335
```

```
100   NUM(J)=LINE(26+J)                                          IN  340
      KST=LINE(50)                                               IN  345
      IF (KST-KBL) 120,110,120                                   IN  350
110   KST=0                                                      IN  355
C     ERROR, NO TERMINATOR PROVIDED                              IN  360
      MM=MM-2                                                    IN  365
      WRITE (IOUT,30)                                            IN  370
120   KCN=LINE(52)                                               IN  375
      IF (KCN-KBL) 140,130,140                                   IN  380
130   KCN=0                                                      IN  385
C     ERROR, NO CONTINUATION SYMBOL PROVIDED                     IN  390
      MM=MM-1                                                    IN  395
      WRITE (IOUT,40)                                            IN  400
140   KZ=LINE(45)                                                IN  405
      DO 150 J=1,31                                              IN  410
150   LOC(J)=0                                                   IN  415
C     SET STATUS INDICATOR TO READY                              IN  420
      MSTAT=1                                                    IN  425
C     INITIALIZATION FINISHED                                    IN  430
      RETURN                                                     IN  435
C     LOCATE POSITION OF FIELD                                   IN  440
160   IF (M+61) 190,190,170                                      IN  445
170   IF (MSTAT) 210,200,180                                     IN  450
180   MX=-M                                                      IN  455
      MM=LOC(MX)                                                 IN  460
      RETURN                                                     IN  465
C     ERROR TERMINATION SECTION                                  IN  470
190   WRITE (IOUT,50) M                                          IN  475
      GO TO 210                                                  IN  480
200   WRITE (IOUT,70)                                            IN  485
210   WRITE (IOUT,60)                                            IN  490
C     THE PROGRAM IS DISABLED TO PREVENT FURTHER OPERATION UNTIL  IN  495
C     RECOVERY IS MADE                                           IN  500
      MSTAT=-1                                                   IN  505
      DO 220 J=1,30                                              IN  510
      IT(J)=-2                                                   IN  515
220   N(J)=0                                                     IN  520
      RETURN                                                     IN  525
C                                                                IN  530
C     THIS IS THE OPERATIONAL PORTION OF INPUTS                  IN  535
230   NMODES=6                                                   IN  540
      IF (M-NMODES) 240,240,190                                  IN  545
240   IF (MSTAT) 210,200,250                                     IN  550
C     THE SWITCHES I1 AND I2 ARE SET DEPENDING ON THE VALUE OF M IN  555
250   GO TO (260,270,280,290,300,190), M                         IN  560
260   I1=0                                                       IN  565
      I2=0                                                       IN  570
      GO TO 310                                                  IN  575
270   I1=1                                                       IN  580
      I2=0                                                       IN  585
      GO TO 310                                                  IN  590
280   I1=1                                                       IN  595
      I2=1                                                       IN  600
      GO TO 310                                                  IN  605
290   I1=1                                                       IN  610
      I2=2                                                       IN  615
      GO TO 310                                                  IN  620
300   GO TO 190                                                  IN  625
C     PREPARE TO CONVERT LINE                                    IN  630
C     K IS THE CURRENT POSITION OF THE SCAN                      IN  635
C     NC IS THE CURRENT POSITION IN N                            IN  640
310   K=1                                                        IN  645
      NC=1                                                       IN  650
      DO 320 J=1,60                                              IN  655
C     ERASE THE ARRAYS                                           IN  660
      N(J)=0                                                     IN  665
      IT(J)=0                                                    IN  670
```

```
320   LOC(J)=0                                                          IN  675
      L=0                                                               IN  680
      KNT=0                                                             IN  685
      MK=80                                                             IN  690
C     MM, IF NON-ZERO, IS THE LINE WIDTH                                IN  695
C     MK, THE LINE WIDTH, IS SET TO 80 IN DEFAULT                       IN  700
      IF (MM) 210,340,330                                               IN  705
330   MK=MM                                                             IN  710
C     TEST FOR LINE TERMINATOR                                          IN  715
340   IF (LINE(K)-KST) 350,530,350                                      IN  720
C     TEST FOR CONTINUATION SYMBOL                                      IN  725
350   IF (LINE(K)-KCN) 360,520,360                                      IN  730
360   IF (LINE(K)-KBL) 390,370,390                                      IN  735
C     IF THIS IS A BLANK, ADVANCE THE POINTER AND CONTINUE THE SCAN     IN  740
370   K=K+1                                                             IN  745
      IF (L) 210,380,450                                                IN  750
380   IF (K-MK) 340,340,530                                             IN  755
390   CONTINUE                                                          IN  760
      DO 400 J=1,10                                                     IN  765
C     CHECK FOR A NUMBER                                                IN  770
      IF (LINE(K)-NUM(J)) 400,410,400                                   IN  775
400   CONTINUE                                                          IN  780
      GO TO 440                                                         IN  785
C     NUMBER FOUND                                                      IN  790
410   NN=J-1                                                            IN  795
C     ADD NUMERICAL VALUE TO COUNT REGISTER                             IN  800
      KNT=10*KNT+NN                                                     IN  805
      L=1                                                               IN  810
      IF (I1) 210,450,420                                               IN  815
420   IF (K-MK) 430,450,450                                             IN  820
430   K=K+1                                                             IN  825
      GO TO 340                                                         IN  830
440   IF (L) 210,460,450                                                IN  835
C     STORE THE NUMBER IN N                                             IN  840
450   N(NC)=KNT                                                         IN  845
      IT(NC)=1                                                          IN  850
      LOC(NC)=K-1                                                       IN  855
      KNT=0                                                             IN  860
      L=0                                                               IN  865
      GO TO 510                                                         IN  870
C     MUST DETERMINE WHETHER TO STORE OR REJECT CHARACTER               IN  875
460   IF (I2-1) 500,470,490                                             IN  880
470   DO 480 J=1,26                                                     IN  885
C     SEE IF CHARACTER IS A LETTER                                      IN  890
      IF (KA(J)-LINE(K)) 480,500,480                                    IN  895
480   CONTINUE                                                          IN  900
      IF (KZ-LINE(K)) 490,500,490                                       IN  905
490   K=K+1                                                             IN  910
C     CHARACTER IGNORED, POINTER ADVANCED, SCAN CONTINUES               IN  915
      GO TO 340                                                         IN  920
C     STORE A NON-NUMERIC CHARACTER IN N                                IN  925
500   N(NC)=LINE(K)                                                     IN  930
      IT(NC)=2                                                          IN  935
      LOC(NC)=K                                                         IN  940
      K=K+1                                                             IN  945
C     INCREASE COUNT OF CHARACTERS OR NUMBERS STORED                    IN  950
510   NC=NC+1                                                           IN  955
      IF (NC-60) 340,340,530                                            IN  960
C     CONTINUATION SYMBOL READ, STORE -1 IN N AND RETURN                IN  965
520   LOC(NC)=K                                                         IN  970
      IT(NC)=-1                                                         IN  975
      N(NC)=0                                                           IN  980
      NC=NC+1                                                           IN  985
C     INPUT RETURNS WHEN THE LINE IS EXHAUSTED OR WHEN A STOP OR        IN  990
C     CONTINUE SYMBOL IS READ                                           IN  995
530   RETURN                                                            IN 1000
      END                                                               IN 1005-
```

```
      SUBROUTINE LINTYP (IBUF,K,L,N)                                    LNT   5
C           LINTYP                                                      LNT  10
C                                                                       LNT  15
C     LINTYP DETERMINES THE FORMAT OR KIND OF INFORMATION OF A RECORD INLNT  20
C     A FILE CONTAINING SEVERAL DIFFERENT KINDS OF RECORDS, WHICH ARE   LNT  25
C     DISTINGUISHED FROM EACH OTHER BY A LABEL OR IDENTIFIER IN A FIXED LNT  30
C     FIELD.                                                            LNT  35
C     A NUMBER INDICATING THE TYPE OF RECORD IS RETURNED IN N.          LNT  40
C                                                                       LNT  45
C     WHEN L IS SET TO ZERO TO SIGNAL INITIALIZATION LINTYP READS       LNT  50
C     CONTROL CARDS FROM THE CARD READER                                LNT  55
C     THE FIRST CONTROL CARD IS IN I3 FORMAT AND TELLS THE STARTING AND LNT  60
C     ENDING POSITIONS OF THE IDENTIFER FIELD WITHIN A RECORD           LNT  65
C     EACH SUBSEQUENT CONTROL CARD IS IN 10A1 FORMAT AND CONTAINS THE   LNT  70
C     CHARACTERS WHICH WILL BE USED AS AN IDENTIFIER SEQUENCE FOR A     LNT  75
C     RECORD TYPE.  ANY RECORD HAVING THE IDENTIFIER SEQUENCE CONTAINED LNT  80
C     ON THE FIRST OF THESE CARDS WILL BE CONSIDERED A TYPE 1 RECORD,   LNT  85
C     ANY CARD HAVING THE SAME IDENTIFIER AS THE SECOND SUCH CARD IS A  LNT  90
C     TYPE 2 RECORD, ETC.  THE MATCH CHARACTER, IAHC(42), INDICATES THATLNT  95
C     ANY CHARACTER IS TO BE ACCEPTED IN THE LOCATION WHERE IT APPEARS  LNT 100
C     A BLANK CARD SIGNALS THE END OF THE CARD INPUT TO LINTYP          LNT 105
C                                                                       LNT 110
C     IHUF - A BLOCK WHICH INCLUDES THE LINE IN QUESTION.  IHUF MAY BE  LNT 115
C     THE LINE ITSELF OR IT MAY BE AN ARRAY CONTAINING BLOCKED DATA.    LNT 120
C     K - THE POINTER TELLING LINTYP THE POSITION IN IHUF WHERE THE LINELNT 125
C     BEGINS.  IF THE DATA HAS BEEN UNBLOCKED BY THE MAIN PROGRAM BEFORELNT 130
C     CALLING LINTYP K SHOULD BE SET TO 1.                              LNT 135
C     L - IS SET TO ZERO TO INITIALIZE LINTYP OR TO A POSITIVE NUMBER TOLNT 140
C     IDENTIFY A RECORD                                                 LNT 145
C     N - ON RETURN CONTAINS THE NUMERIC TYPE OF THE RECORD IF IT HAD A LNT 150
C     A RECOGNIZABLE IDENTIFIER OR ZERO IF IT DID NOT                   LNT 155
C     IAHC - THE STANDARD ALPHABETIC DICTIONARY CARD.  IAHC(42) SHOULD  LNT 160
C     CONTAIN A MATCH CHARACTER WHICH IN A CONTROL CARD INDICATES THAT  LNT 165
C     ANY CHARACTER MAY APPEAR IN THIS POSITION FOR THIS IDENTIFIER     LNT 170
C     SEQUENCE                                                          LNT 175
C     IN AND IOUT - LOGICAL UNIT NUMBERS FOR CARD READER AND PRINTER    LNT 180
C                                                                       LNT 185
C     THIS ROUTINE WRITTEN BY R. MCCLENON , NSRDS-NBS, FEB. 1970        LNT 190
C                                                                       LNT 195
      COMMON /IO/ IN,IOUT                                               LNT 200
      DIMENSION IBUF(4000),IAHC(80),KODE(250)                           LNT 205
      COMMON /IA/ IAHC                                                  LNT 210
      COMMON /W/ KL,NTAHL                                               LNT 215
      DIMENSION JDATA(60),KEY(60),KARD(80)                              LNT 220
10    FORMAT (1X,80A1)                                                  LNT 225
20    FORMAT (80A1)                                                     LNT 230
30    FORMAT (1X,I2,1X,10A1)                                            LNT 235
      IF (L) 40,40,90                                                   LNT 240
C     INITIALIZATION - READ CONTROL CARDS                               LNT 245
40    READ (IN,20) (KARD(J),J=1,80)                                     LNT 250
      WRITE (IOUT,10) (KARD(J),J=1,80)                                  LNT 255
      CALL INPUT (4,KARD,JDATA,KEY,80)                                  LNT 260
      K1=JDATA(1)                                                       LNT 265
      K2=JDATA(2)                                                       LNT 270
      IW=K2-K1+1                                                        LNT 275
      LIM=250                                                           LNT 280
      IF (IW-10) 50,50,70                                               LNT 285
50    I1=1                                                              LNT 290
      I2=IW                                                             LNT 295
C     READ AN IDENTIFIER INTO KODE                                      LNT 300
      DO 60 M=1,NTAHL                                                   LNT 305
      READ (IN,20) (KODE(J),J=I1,I2)                                    LNT 310
      WRITE (IOUT,30) M,(KODE(J),J=I1,I2)                               LNT 315
```

```
      I1=I1+IW                                                          LNT 320
      I2=I2+IW                                                          LNT 325
      IF (I2-LIM) 60,60,70                                              LNT 330
60    CONTINUE                                                          LNT 335
      RETURN                                                            LNT 340
70    WRITE (IOUT,80)                                                   LNT 345
80    FORMAT (42HOSTORAGE LIMIT OF LINTYP EXCEEDED.  STOP. )            LNT 350
      STOP                                                              LNT 355
C     COMPARE RECORD AGAINST EACH STORED IDENTIFIER                     LNT 360
90    DO 120 J=1,NTABL                                                  LNT 365
      KK=K1+K-1                                                         LNT 370
      DO 110 I=1,IW                                                     LNT 375
      II=I+(J-1)*IW                                                     LNT 380
      IF (IABC(42)-KODE(II)) 100,110,100                                LNT 385
100   IF (IBUF(KK)-KODE(II)) 120,110,120                                LNT 390
110   KK=KK+1                                                           LNT 395
C     IDENTIFIER RECOGNIZED                                             LNT 400
      N=J                                                               LNT 405
      RETURN                                                            LNT 410
120   CONTINUE                                                          LNT 415
C     RECORD DID NOT CONTAIN A RECOGNIZABLE IDENTIFIER                  LNT 420
      N=0                                                               LNT 425
      RETURN                                                            LNT 430
      END                                                               LNT 435-
```

```
      SUBROUTINE TAPOUT (M,LINE,LEN)                                    TAP   5
C     TAPOUT                                                            TAP  10
C                                                                       TAP  15
C     THIS SUBROUTINE WRITES THE OUTPUT FROM COMBO ONTO ONE OR MORE     TAP  20
C     TAPES.  (CARD OUTPUT MAY ALSO BE OBTAINED FROM THIS SUBROUTINE.   TAP  25
C     NO CHECK IS MADE TO ASSURE THAT LOGICAL UNITS SPECIFIED ARE LEGAL TAP  30
C     FOR THE INSTALLATION.)                                            TAP  35
C     THE FOURTH NUMBER ON THE SECOND CONTROL CARD TO COMBO (THE ONE     TAP  40
C     AFTER THE ALPHABET CARD) INDICATES THE TYPE OF OUTPUT DESIRED FROMTAP  45
C     THE ROUTINE TAPOUT                                                TAP  50
C     A 0 INDICATES NO OUTPUT OTHER THAN A PRINTED REPORT.              TAP  55
C     A 1 INDICATES A SINGLE OUTPUT UNIT, ON WHICH EVERY LINE WILL BE   TAP  60
C     WRITTEN.  THIS IS ESSENTIALLY A TAPE COPY OF THE PRINTED REPORT.  TAP  65
C     A 2 INDICATES MULTIPLE OUTPUT UNITS, OR SELECTIVE OUTPUT TO TAPE. TAP  70
C     ONLY IN THE EVENT THAT 2 IS SPECIFIED WILL SPECIAL CONTROL CARDS  TAP  75
C     BE READ BY TAPOUT.                                                TAP  80
C     THE CONTROL CARDS TO TAPOUT ARE FREE-FORM, VIA THE ROUTINE INPUT. TAP  85
C     THERE IS ONE FOR EACH TYPE OF LINE WHICH IT IS DESIRED TO WRITE TOTAP  90
C     TAPE (OR OTHER DEVICE BESIDES THE PRINTER).  THE CARD FOR A GIVEN TAP  95
C     LINE TYPE HAS THE TYPE NUMBER, FOLLOWED BY ONE OR MORE LOGICAL    TAP 100
C     UNITS.  FOR INSTANCE, IF A CARD CONTAINS                          TAP 105
C     3    7 9                                                          TAP 110
C     EVERY TYPE 3 LINE THAT SATISFIED THE SEARCH WILL, AFTER EDITING,  TAP 115
C     BE WRITTEN ON UNITS 7 AND 9.                                      TAP 120
C     ONE RESTRICTION IS THAT THE CONTROL CARDS MUST BE IN ORDER BY LINETAP 125
C     TYPE, THAT IS, LINE 4 MUST FOLLOW LINE 2 OR LINE 3.               TAP 130
C     LINES MAY BE OMITTED FROM THE LIST.  A LINE WHICH IS OMITTED WILL TAP 135
C     NOT BE WRITTEN IO TAPE.                                           TAP 140
C     THE READING OF A FINIS CARD TERMINATES THE LIST                   TAP 145
C                                                                       TAP 150
```

```
C     ARGUMENTS --                                                      TAP 155
C     M - IF ZERO, THIS IS INITIALIZATION.  CONTROL CARDS WILL BE READ. TAP 160
C       FOR INITIALIZATION, LINE MUST CONTAIN THE NUMBERS FROM THE COMBOTAP 165
C     PARAMETER CARD (JDATA).  SEE THE LISTING OF COMBO.                TAP 170
C     IF M IS POSITIVE, IT IS THE LINE TYPE.                            TAP 175
C     IF M IS NEGATIVE, THIS IS  PROGRAM TERMINATION.  A FILE MARK IS   TAP 180
C     WRITTEN ON EACH TAPE.                                             TAP 185
C     LINE - THE BUFFER TO BE WRITTEN ONTO THE OUTPUT TAPE(S)           TAP 190
C     LEN - THE NUMBER OF CHARACTERS IN LINE                            TAP 195
C     JUPOIN CONTAINS POINTERS TO THE UNIT DESIGNATIONS                 TAP 200
C     JUNITS CONTAINS THE UNIT DESIGNATIONS                             TAP 205
C                                                                       TAP 210
      DIMENSION LINE(200),JUNITS(200),JUPOIN(100),KARD(80)              TAP 215
      DIMENSION JDATA(60),KEY(60)                                       TAP 220
      COMMON /W/ IW,NTABL                                               TAP 225
      COMMON /IA/ IABC(120)                                             TAP 230
      COMMON /IO/ IN,IOUT                                               TAP 235
C     THIS IS THE CARD PUNCH - IT IS NOT ENDFILED                       TAP 236
      DATA KPUNCH / 3 /                                                 TAP 238
C     WHICH MODE                                                        TAP 240
      IF (M) 400,20,320                                                 TAP 245
C     INITIALIZATION                                                    TAP 250
20    MODE=LINE(4)                                                      TAP 255
      IF (LINE(4)-1) 240,40,60                                          TAP 260
C     SINGLE OUTPUT UNIT                                                TAP 265
40    JTAPE=LINE(2)                                                     TAP 270
      GO TO 240                                                         TAP 275
C     STORE MULTIPLE OUTPUT UNIT LIST                                   TAP 280
60    WRITE (IOUT,480)                                                  TAP 285
      K=1                                                               TAP 290
      II=0                                                              TAP 295
80    READ (IN,500) (KARD(J),J=1,80)                                    TAP 300
      WRITE (IOUT,520) (KARD(J),J=1,80)                                 TAP 305
C     CHECK FOR FINIS CARD                                              TAP 310
      IF (KARD(1).NE.IABC(6)) GO TO 100                                 TAP 315
      IF (KARD(2).NE.IABC(9)) GO TO 100                                 TAP 320
      IF (KARD(3).NE.IABC(14)) GO TO 100                                TAP 325
      IF (KARD(4).NE.IABC(9)) GO TO 100                                 TAP 330
      IF (KARD(5).NE.IABC(19)) GO TO 100                                TAP 335
      GO TO 200                                                         TAP 340
100   CALL INPUT (4,KARD,JDATA,KEY,80)                                  TAP 345
      I=JDATA(1)                                                        TAP 350
      IF (I.EQ.0) GO TO 200                                             TAP 355
      IF (I.LE.II) GO TO 260                                            TAP 360
      I1=I-1                                                            TAP 365
      II1=II+1                                                          TAP 370
      IF (II1.GT.I1) GO TO 140                                          TAP 375
      DO 120 J=II1,I1                                                   TAP 380
      IF (J.GT.NTABL) GO TO 260                                         TAP 385
120   JUPOIN(J)=K                                                       TAP 390
140   L=2                                                               TAP 395
      JUPOIN(I)=K                                                       TAP 400
160   CONTINUE                                                          TAP 405
      IF (KEY(L).LT.1) GO TO 180                                        TAP 410
      JUNITS(K)=JDATA(L)                                                TAP 415
      K=K+1                                                             TAP 420
      L=L+1                                                             TAP 425
      IF (K-200) 160,280,280                                            TAP 430
180   II=I                                                              TAP 435
      GO TO 80                                                          TAP 440
200   JUPOIN(NTABL+1)=K                                                 TAP 445
      IF (II.GE.NTABL) GO TO 240                                        TAP 450
      II1=II+1                                                          TAP 455
```

```
      DO 220 J=II1,NTABL                                                TAP 460
220   JUPOIN(J)=K                                                       TAP 465
240   RETURN                                                            TAP 470
260   WRITE (IOUT,540)                                                  TAP 475
      GO TO 300                                                         TAP 480
C     ERROR                                                             TAP 485
280   WRITE (IOu1,560)                                                  TAP 490
300   STOP                                                              TAP 495
C     OPERATION - WRITE BUFFER LINE ONTO TAPE                           TAP 500
C     CHECK FOR SINGLE OUTPUT UNIT OR MULTIPLE OUTPUT UNITS             TAP 505
320   IF (MODE-1) 240,340,360                                           TAP 510
C     WRITE ON SINGLE OUTPUT UNIT                                       TAP 515
340   WRITE (JTAPE,500) (LINE(J),J=1,LEN)                               TAP 520
      GO TO 240                                                         TAP 525
C     WRITE ON MULTIPLE OUTPUT UNITS                                    TAP 530
360   K1=JUPOIN(M)                                                      TAP 535
C     SET POINTERS                                                      TAP 540
      K2=JUPOIN(M+1)-1                                                  TAP 545
      IF (K2.LT.K1) GO TO 240                                           TAP 550
      DO 380 J=K1,K2                                                    TAP 555
      JTAPE=JUNITS(J)                                                   TAP 560
380   WRITE (JTAPE,500) (LINE(I),I=1,LEN)                               TAP 565
      GO TO 240                                                         TAP 570
C     TERMINATION - WRITE FILE MARKS                                    TAP 575
400   IF (MODE-1) 240,420,440                                           TAP 580
420   IF (JTAPE.EQ.KPUNCH) GO TO 240                                    TAP 585
      END FILE JTAPE                                                    TAP 588
      GO TO 240                                                         TAP 590
440   KK=K-1                                                            TAP 595
      DO 460 J=1,KK                                                     TAP 600
      JTAPE=JUNITS(J)                                                   TAP 605
      IF (JTAPE.EQ.KPUNCH) GO TO 460                                    TAP 608
      END FILE JTAPE                                                    TAP 610
460   CONTINUE                                                          TAP 612
      GO TO 240                                                         TAP 615
C                                                                       TAP 620
480   FORMAT (49H0 * MULTIPLE OUTPUT UNITS -- UNIT LIST FOLLOWS * )     TAP 625
500   FORMAT (80A1)                                                     TAP 630
520   FORMAT (1X,80A1)                                                  TAP 635
540   FORMAT (49HOTAPE UNIT CARDS OUT OF ORDER, OR NO FINIS CARD. )     TAP 640
560   FORMAT (37HOTOO MANY TAPE SPECIFICATIONS. STOP. )                 TAP 645
      END                                                               TAP 650-
```

```
C     MULTIPLE TABLE SEARCH SUHROUTINE                                  MSR   5
C                                                                       MSR  10
C     MSERCH PERFORMS ANY OF UP TO 99 STRING SEARCHES, AND-MODE OR OR-  MSR  15
C     MODE, ON A LINE OF A1 CHARACTERS AND SIGNALS SUCCESS OR FAILURE.  MSR  20
C                                                                       MSR  25
      SUBROUTINE MSERCH (ICOL,LEN,M)                                    MSR  30
C                                                                       MSR  35
C     THE ARGUMENTS FOR THIS ROUTINE ARE --                             MSR  40
C     ICOL - A VECTOR OF A1 CHARACTERS TO BE SEARCHED                   MSR  45
C     LEN - THE NUMBER OF CHARACTERS IN ICOL                            MSR  50
C     M - ON ENTRY, THE NUMBER OF THE SEARCH TABLE TO BE USED           MSR  55
C       IF ZERO, INSTRUCTS MSERCH TO READ CONTROL CARDS                 MSR  60
C       ON RETURN, 0 INDICATES AN UNSUCCESSFUL SEARCH, 1 SIGNALS SUCCESSMSR  65
C                                                                       MSR  70
C     CONTROL CARDS ARE --                                              MSR  75
C     THE EDPAC DICTIONARY IS ALREADY IN COMMON BLOCK IA.  IT HAS THE 26MSR  80
C     LETTERS IN ORDER IN COLUMNS 1-26, AND THE DIGITS FROM 0 TO 9 IN   MSR  85
C     COLUMNS 27-36.  COLUMN 47 SHOULD BE BLANK.                        MSR  90
C     COLUMN 42 CONTAINS A CHARACTER, THE UNIVERSAL MATCH SYMBOL, WHOSE MSR  95
C     APPEARANCE IN A SEARCH STRING INDICATES THAT ANY CHARACTER FOUND  MSR 100
C     IN THAT POSITION IN THE STRING WILL SATISFY THE SEARCH.           MSR 105
C     COLUMN 50 CONTAINS THE SEARCH STRING TERMINATOR WHICH DELIMITS THEMSR 110
C     SEARCH STRINGS.                                                   MSR 115
C                                                                       MSR 120
C     THE SEARCH TABLES ARE READ IN.  EACH BEGINS WITH A CARD ON WHICH  MSR 125
C     EITHER THE WORD AND OR THE WORD OR IS PUNCHED STARTING IN COLUMN  MSR 130
C     1.                                                                MSR 135
C             THE 'AND' DEMANDS THAT ALL SEARCH WORDS OR PHRASES MUST   MSR 140
C             BE FOUND IN A CARD IMAGE IN ORDER TO BE CHOSEN.           MSR 145
C             THE 'OR' REQUIRES ONLY THAT ONE OF THE SEARCH WORDS OR    MSR 150
C             PHRASES BE FOUND TO SATISFY THE SEARCH.                   MSR 155
C     THE SEARCH STRINGS ARE EACH ON A SEPARATE CARD, STARTING IN COLUMNMSR 160
C     1 AND ENDING WITH THE TERMINATOR.                                 MSR 165
C                                                                       MSR 170
C                                                                       MSR 175
C     EACH SEARCH TABLE HAS ITS END MARKED BY A CARD WITH THE TERMINATORMSR 180
C     IN COLUMN 1.  THE PROGRAM THEN READS ANOTHER TABLE.               MSR 185
C     TABLES ARE NUMBERED IN ORDER OF APPEARANCE.                       MSR 190
C                                                                       MSR 195
C     THE END OF SEARCH TABLES, AND OF CARD INPUT, IS MARKED BY A CARD  MSR 200
C     WITH THE WORD FINIS IN COLUMNS 1-5 AND A TERMINATOR IN COLUMN 6.  MSR 205
C                                                                       MSR 210
C     VARIABLES --                                                      MSR 215
C     IC - A VECTOR SHARED WITH OTHER ROUTINES USED FOR STRING STORAGE  MSR 220
C     ICLEN - THE SIZE OF IC                                            MSR 225
C     ICNOW - THE FIRST AVAILABLE (UNUSED) POSITION IN IC.  IT MUST BE  MSR 230
C     RESET AFTER STORING IN IC.                                        MSR 235
C     ITAPE - UNIT NUMBER FOR THE CARD READER                           MSR 240
C     IOTAPE - UNIT NUMBER FOR THE PRINTER                              MSR 245
C     IA - THE EDPAC DICTIONARY                                         MSR 250
C     IB - TEMPORARY STORAGE FOR INPUT FROM CARDS                       MSR 255
C     N - THE LENGTH OF EACH SEARCH STRING                              MSR 260
C     KS - POINTERS TO THE BEGINNING OF EACH TABLE IN N                 MSR 265
C     KC - POINTERS TO THE BEGINNING OF EACH TABLE IN IC                MSR 270
C     IAND - THE SEARCH MODE (AND/OR) FOR EACH TABLE                    MSR 275
C                                                                       MSR 280
C     CODE WRITTEN BY MRS CARLA G. MESSINA  NSRDS NBS 1966              MSR 285
C                                                                       MSR 290
C     ADAPTED AS SUBROUTINE BY R. MCCLENON NSRDS-NBS  FEB. 1970         MSR 295
C                                                                       MSR 300
      COMMON /STR/ ICLEN,ICNOW,IC                                       MSR 305
      COMMON /IO/ ITAPE,IOTAPE                                          MSR 310
      COMMON /IA/ IA(120)                                               MSR 315
      DIMENSION IB(81),IC(8000),N(200),ICOL(4000)                       MSR 320
      DIMENSION KC(100),IAND(99),KS(100)                                MSR 325
C     INITIALIZATION OR SEARCH                                          MSR 330
```

```
      IF (M) 240,10,240                                          MSR 335
C     READ DICTIONARY                                            MSR 340
10    CONTINUE                                                   MSR 345
      N1=0                                                       MSR 350
      N3=ICNOW                                                   MSR 355
      NSTRIN=200                                                 MSR 360
      L=1                                                        MSR 365
      KS(1)=1                                                    MSR 370
      KC(1)=1                                                    MSR 375
20    READ (ITAPE,390) (IB(J),J=1,80)                            MSR 380
      WRITE (IOTAPE,420) L                                       MSR 385
      WRITE (IOTAPE,400) (IB(J),J=1,80)                          MSR 390
C     WHICH AND/OR MODE                                          MSR 395
      IF (IB(1)-IA(1)) 60,30,60                                  MSR 400
30    IF (IB(2)-IA(14)) 60,40,60                                 MSR 405
40    IF (IB(3)-IA(4)) 60,50,60                                  MSR 410
50    IAND(L)=1                                                  MSR 415
      WRITE (IOTAPE,430) IA(1),IA(12),IA(12)                     MSR 420
      GO TO 70                                                   MSR 425
60    IAND(L)=0                                                  MSR 430
      WRITE (IOTAPE,430) IA(1),IA(14),IA(25)                     MSR 435
C     READ A STRING                                              MSR 440
70    READ (ITAPE,390) (IB(J),J=1,80)                            MSR 445
      N2=0                                                       MSR 450
      J=1                                                        MSR 455
      IF (IB(1)-IA(6)) 130,80,130                                MSR 460
80    IF (IB(2)-IA(9)) 130,90,130                                MSR 465
90    IF (IB(3)-IA(14)) 130,100,130                              MSR 470
100   IF (IB(4)-IA(9)) 130,110,130                               MSR 475
110   IF (IB(5)-IA(19)) 130,120,130                              MSR 480
120   IF (IB(6)-IA(50)) 130,230,130                              MSR 485
130   DO 140 I=1,80                                              MSR 490
      IF (IB(I)-IA(50)) 140,150,140                              MSR 495
140   N2=I                                                       MSR 500
150   IF (N2) 220,220,160                                        MSR 505
160   N1=N1+1                                                    MSR 510
      N(N1)=N2                                                   MSR 515
      N4=N3+N2-1                                                 MSR 520
      IF (N4-ICLEN) 190,190,170                                  MSR 525
C     OVERFLOW                                                   MSR 530
170   WRITE (IOTAPE,440) N4,N1                                   MSR 535
180   STOP                                                       MSR 540
190   IF (N1-200) 200,200,170                                    MSR 545
200   J=1                                                        MSR 550
      DO 210 I=N3,N4                                             MSR 555
      IC(I)=IB(J)                                                MSR 560
210   J=J+1                                                      MSR 565
      N3=N3+N2                                                   MSR 570
      WRITE (IOTAPE,400) (IB(J),J=1,N2)                          MSR 575
      GO TO 70                                                   MSR 580
C     END OF TABLE - START NEW TABLE                             MSR 585
220   KS(L+1)=N1+1                                               MSR 590
      KC(L+1)=N3                                                 MSR 595
      L=L+1                                                      MSR 600
      GO TO 20                                                   MSR 605
C     END OF STRINGS                                             MSR 610
230   KS(L+1)=N1+1                                               MSR 615
      ICNOW=N3                                                   MSR 620
      WRITE (IOTAPE,410) L                                       MSR 625
      RETURN                                                     MSR 630
C                                                                MSR 635
C     SEARCH                                                     MSR 640
240   IF (L-M) 260,250,250                                       MSR 645
250   IF (M) 260,260,270                                         MSR 650
260   WRITE (IOTAPE,450) M                                       MSR 655
      GO TO 180                                                  MSR 660
```

```
270   IF (IAND(M)) 340,280,340                                          MSR 665
C     OR-SEARCH                                                         MSR 670
280   N3=KC(M)                                                          MSR 675
      NN=KS(M)                                                          MSR 680
      N1=KS(M+1)                                                        MSR 685
      DO 320 K1=NN,N1                                                   MSR 690
      N2=N(K1)                                                          MSR 695
      N4=N3+N2-1                                                        MSR 700
      I2=LEN+1-N2                                                       MSR 705
      DO 310 I=2,I2                                                     MSR 710
      J1=1                                                              MSR 715
C     DOES IT MATCH                                                     MSR 720
      DO 300 J=N3,N4                                                    MSR 725
      K=I+J1-1                                                          MSR 730
      IF (ICOL(K)-IA(42)) 290,300,290                                   MSR 735
290   IF (ICOL(K)-IC(J)) 310,300,310                                    MSR 740
300   J1=J1+1                                                           MSR 745
C     SUCCESS                                                           MSR 750
      GO TO 330                                                         MSR 755
C     NO MATCH, KEEP TRYING                                             MSR 760
310   CONTINUE                                                          MSR 765
320   N3=N3+N2                                                          MSR 770
      M=0                                                               MSR 775
      RETURN                                                            MSR 780
330   M=1                                                               MSR 785
      RETURN                                                            MSR 790
C     AND-SEARCH                                                        MSR 795
340   N3=KC(M)                                                          MSR 800
      NN=KS(M)                                                          MSR 805
      N1=KS(M+1)                                                        MSR 810
      DO 380 K1=NN,N1                                                   MSR 815
      N2=N(K1)                                                          MSR 820
      N4=N3+N2-1                                                        MSR 825
      I2=LEN+1-N2                                                       MSR 830
      DO 370 I=2,I2                                                     MSR 835
      J1=1                                                              MSR 840
C     DOES IT MATCH                                                     MSR 845
      DO 360 J=N3,N4                                                    MSR 850
      K=I+J1-1                                                          MSR 855
      IF (ICOL(K)-IA(42)) 350,360,350                                   MSR 860
350   IF (ICOL(K)-IC(J)) 370,360,370                                    MSR 865
360   J1=J1+1                                                           MSR 870
C     FAILURE                                                           MSR 875
      GO TO 380                                                         MSR 880
C     MATCH, KEEP TRYING                                                MSR 885
370   CONTINUE                                                          MSR 890
      M=0                                                               MSR 895
      RETURN                                                            MSR 900
380   N3=N3+N2                                                          MSR 905
      M=1                                                               MSR 910
      RETURN                                                            MSR 915
C                                                                       MSR 920
390   FORMAT (80A1)                                                     MSR 925
400   FORMAT (1X,80A1)                                                  MSR 930
410   FORMAT (11HOTHERE ARE ,I2,8H TABLES )                             MSR 935
420   FORMAT (13HOSEARCH TABLE,I3)                                      MSR 940
430   FORMAT (47HOTHE PROGRAM IS SEARCHING FOR LINES CONTAINING ,3A1,26HMSR 945
     1 OF THE WORDS GIVEN BELOW.//)                                     MSR 950
440   FORMAT (47HOLIST OF SEARCH WORDS TOO LONG. PLEASE SHORTEN.,/68HOMAMSR 955
     1XIMUM CHARACTER LENGTH IS 8000. MAXIMUM NUMBER OF PHRASES IS 200 /MSR 960
     221H  CURRENT VALUES ARE ,2I6)                                     MSR 965
450   FORMAT (28HOMSERCH HAS NO SEARCH TABLE ,I5)                       MSR 970
      END                                                               MSR 975-
```

```
      SUHROUTINE MSUBS (IE,IW,MODE)                                    MSB   5
C           MSUBS                                                      MSB  10
C     MULTIPLE TAHLE SUBSTITUTION SUBROUTINE                           MSH  15
C                                                                      MSH  20
C     THIS PROGRAM IS A MODIFICATION OF AMSUB WRITTEN HY R.C. THOMPSON MSB  25
C     AMSUH USES TBE LOGIC OF SUHSTITUTE WRITTEN HY MRS. C. MESSINA    MSH  30
C     MODIFICATIONS MADE BY R. MCCLENON, NSRDS-NHS, MARCH 1970.        MSB  35
C                                                                      MSH  40
C     THIS IS A MULTIPLE TABLE SUBSTITUTION SUHROUTINE.  IT ACCEPTS AS MSH  45
C     INPUT A LINE OF TEXT AND SEARCHES AND SUHSTITUTES ACCORDING TO ANYMSH  50
C     OF 99 DIFFERENT TAHLES                                           MSB  55
C     TBE ARGUMENTS ARE --                                             MSB  60
C     IB - THE LINE TO BE PROCESSED ON INPUT.  ON RETURN FROM MSUBS IB MSH  65
C     CONTAINS TBE EDITED LINE.                                        MSH  70
C     IW - TBE LENGTB OF THE INPUT LINE IN IB (IN CHARACTERS).  ON     MSB  75
C     RETURN IW IS TBE NEW LENGTB OF IB.                               MSB  80
C     MODE - ZERO INDICATES TBAT CONTROL CARDS ARE TO HE READ TO       MSB  85
C     INITIALIZE MSUBS.  A POSITIVE NUMBER INDICATES THE NUMBER OF THE MSB  90
C     SUBSTITUTION TABLE WHICB IS TO BE USED TO EDIT IB.               MSB  95
C                                                                      MSB 100
C     COMMON BLOCK IO CONTAINS ITAPE AND IOTAPE, THE LOGICAL UNITS FOR MSB 105
C     THE CARD READER AND PRINTER, RESPECTIVELY.                       MSB 110
C                                                                      MSB 115
C     COMMON HLOCK STR CONTAINS IC IN WHICH THE SUBSTITUTION TABLE IS  MSH 120
C     STORED.  ICLEN IS THE LENGTB OF IC.  ICNOW IS A POINTER TO TBE   MSB 125
C     NEXT AVAILABLE CELL IN IC.  IT IS RESET BY EACH PROGRAM WHICB    MSB 130
C     STORES STRINGS IN IC.                                            MSB 135
C                                                                      MSB 140
C     THE CONTROL CARDS FOR MSUHS ARE AS FOLLOWS --                    MSB 145
C     THE EDPAC DICTIONARY IS ALREADY IN COMMON IA.  THE ALPHABET STARTSMSB 150
C     WITH THE LETTER A IN COLUMN 1 THRU Z IN 26.  TBE NUMBERS 0 TO 9 GOMSB 155
C     IN COLUMNS 27 TO 36.  COLUMN 41 CONTAINS THE PRINTOUT STRING     MSB 160
C     DELIMITER.  COLUMN 47 MUST HE HLANK.                             MSB 165
C     THE NEXT TWO CARDS ARE IN (3A1,I1) FORMAT.  THEY GIVE INFORMATION MSB 170
C     FOR INPUT AND OUTPUT, RESPECTIVELY.  TBE FIRST TWO CHARACTERS ON MSH 175
C     EACB CARD ARE TBE SBIFT UP AND SBIFT DOWN SYMBOLS, RESPECTIVELY. MSB 180
C     TBE THIRD CHARACTER IS TBE UNIVERSAL MATCH CHARACTER IN THE      MSB 185
C     SUBSTITUTION TABLES.  THE NUMBER IS 1 FOR SBIFT AND LOCK MODE OR MSB 190
C     ZERO FOR SBIFT AND UNLOCK MODE.                                  MSB 195
C     THE SUHSTITUTION TABLES FOLLOW.  EACH TABLE IS PRECEDED BY A CARD MSB 200
C     WITB THE WORD 'TABLE' IN COLUMNS 1 TO 5 AND THE TABLE NUMBER IN  MSB 205
C     COLUMNS 6 AND 7.  IF THE FIRST SUBSTITUTION CARDS ARE NOT PRECEDEDMSB 210
C     BY A TABLE NUMBER TABLE 99 WILL BE ASSUMED.                      MSB 215
C     A SUBSTITUTION CARD CONTAINS TBE STRING TO BE SEARCHED FOR       MSB 220
C     FOLLOWED BY THE REPLACEMENT STRING, WITH BOTH STRINGS BRACKETED HYMSB 225
C     ANY CBARACTER NOT APPEARING IN EITHER STRING.  FOR INSTANCE,     MSB 230
C     /YES/  /NO/  WILL CAUSE ALL OCCURRENCES OF 'YES' TO BE CHANGED TO MSB 235
C     'NO' WHEN TBAT TABLE IS USED.                                    MSB 240
C     THE READING OF SUBSTITUTION TABLES IS ENDED BY READING A CARD WITBMSB 245
C     THE WORD 'FINIS' IN COLUMNS 1 TBRU 5 AND COLUMN 6 BLANK          MSH 250
C                                                                      MSB 255
C     THIS PROGRAM SBOULD BE USED WITH SUBROUTINES SULOCK AND SUNLK    MSB 260
C                                                                      MSB 265
      COMMON /IO/ ITAPE,IOTAPE                                         MSB 270
      COMMON /STR/ ICLEN,ICNOW,IC                                      MSB 275
      DIMENSION IC(8000),NS(1500),NFG1(100),NFG2(100),NT(750),NSFG(100) MSB 280
      DIMENSION NB(750),IX(20)                                         MSB 285
```

```
      DIMENSIØN IB(3000)                                              MSB 290
      CØMMØN /IA/ IA(120)                                             MSB 295
      IF (MØDE) 660,10,660                                            MSB 300
10    IBLEN=2995                                                      MSB 305
      IWRT=5                                                          MSB 310
      ICHK=-1                                                         MSB 315
      IEND=0                                                          MSB 320
      ISHFT=0                                                         MSB 325
      NSMAX=1496                                                      MSB 330
      DØ 20 J=1,99                                                    MSB 335
      NFG1(J)=0                                                       MSB 340
20    NFG2(J)=0                                                       MSB 345
      J1=0                                                            MSB 350
      READ (ITAPE,1000) IA(83),IA(85),IA(89),LØCK1                    MSB 355
      WRITE (IØTAPE,1010) IA(83),IA(85),IA(89),LØCK1                  MSB 360
      READ (ITAPE,1000) IA(84),IA(86),IA(90),LØCK2                    MSB 365
      WRITE (IØTAPE,1010) IA(84),IA(86),IA(90),LØCK2                  MSB 370
      N1=0                                                            MSB 375
      LG=9                                                            MSB 380
      LF=9                                                            MSB 385
      N3=ICNØW                                                        MSB 390
C     START READING SUBSTITUTIØN TABLES                               MSB 395
30    READ (ITAPE,1020) (IB(J),J=1,77),KB,KT                          MSB 400
      N2=0                                                            MSB 405
      N22=0                                                           MSB 410
C CHECK FØR FINISH CARD                                               MSB 415
      IF (IB(1)-IA(6)) 80,40,80                                       MSB 420
40    IF (IB(2)-IA(9)) 220,50,220                                     MSB 425
50    IF (IB(3)-IA(14)) 220,60,220                                    MSB 430
60    IF (IB(4)-IA(9)) 220,70,220                                     MSB 435
70    IF (IB(5)-IA(19)) 220,450,220                                   MSB 440
C     CHECK FØR TABLE CARD                                            MSB 445
80    IF (IB(1)-IA(20)) 220,90,220                                    MSB 450
90    IF (IB(2)-IA(1)) 220,100,220                                    MSB 455
100   IF (IB(3)-IA(2)) 220,110,220                                    MSB 460
110   IF (IB(4)-IA(12)) 220,120,220                                   MSB 465
120   IF (IB(5)-IA(5)) 220,130,220                                    MSB 470
C     DETERMINE TABLE NUMBER                                          MSB 475
130   NFG2(LT)=N1                                                     MSB 480
      LT=0                                                            MSB 485
      LTT=0                                                           MSB 490
      DØ 140 J=27,36                                                  MSB 495
      IF (IB(6)-IA(J)) 140,150,140                                    MSB 500
140   LT=LT+1                                                         MSB 505
      LT=0                                                            MSB 510
150   DØ 160 J=27,36                                                  MSB 515
      IF (IB(7)-IA(J)) 160,170,160                                    MSB 520
160   LTT=LTT+1                                                       MSB 525
      LTT=0                                                           MSB 530
170   IF (LTT) 190,190,180                                            MSB 535
180   LT=10*LT+LTT                                                    MSB 540
190   IF (LT) 200,200,210                                             MSB 545
200   WRITE (IØTAPE,1040) IA(6),IA(7)                                 MSB 550
      IEND=1                                                          MSB 555
      GØ TØ 30                                                        MSB 560
210   NFG1(LT)=N1+1                                                   MSB 565
      NSFG(LT)=N3                                                     MSB 570
      GØ TØ 440                                                       MSB 575
```

```
C  FIND LENGTH OF SEARCH STRING N2                                    MSB 580
220    DO 240 I=2,75                                                 MSB 585
       IF (IB(I)-IB(1)) 240,230,240                                  MSB 590
230    IF (N2) 30,30,250                                             MSB 595
240    N2=I-1                                                        MSB 600
250    J=N2+3                                                        MSB 605
       IF (J-76) 270,260,260                                         MSB 610
260    WRITE (IOTAPE,1050) IB(1),(IB(I),I=1,77)                      MSB 615
       IEND=1                                                        MSB 620
       GO TO 30                                                      MSB 625
270    K=J+1                                                         MSB 630
C      FIND BEGINNING OF REPLACEMENT STRING                          MSB 635
       DO 280 I=J,76                                                 MSB 640
       IF (IB(I)-IB(1)) 280,290,280                                  MSB 645
280    K=I+2                                                         MSB 650
       GO TO 260                                                     MSB 655
C      FIND LENGTH OF REPLACEMENT STRING N22                         MSB 660
290    DO 300 I=K,77                                                 MSB 665
       IF (IB(I)-IB(1)) 300,310,300                                  MSB 670
300    N22=I-K+1                                                     MSB 675
       GO TO 260                                                     MSB 680
C      PLACE STRINGS IN BUFFER IC                                    MSB 685
310    N1=N1+2                                                       MSB 690
       N4=N3+N2-1                                                    MSB 695
       IF (N4-ICLEN+80) 330,330,320                                  MSB 700
320    WRITE (IOTAPE,1060) ICLEN,NSMAX,N4,N1                         MSB 705
       IEND=1                                                        MSB 710
       STOP                                                          MSB 715
330    IF (N1-NSMAX) 340,340,320                                     MSB 720
340    J=2                                                           MSB 725
       NS(N1-1)=N2                                                   MSB 730
       NS(N1)=N22                                                    MSB 735
       IN2=N1/2                                                      MSB 740
       NT(IN2)=KT                                                    MSB 745
       NB(IN2)=KB                                                    MSB 750
       DO 350 I=N3,N4                                                MSB 755
       IC(I)=IB(J)                                                   MSB 760
350    J=J+1                                                         MSB 765
       N3=N3+N2                                                      MSB 770
       IF (N22) 380,380,360                                          MSB 775
360    N4=N3+N22-1                                                   MSB 780
       J=K                                                           MSB 785
       DO 370 I=N3,N4                                                MSB 790
       IC(I)=IB(J)                                                   MSB 795
370    J=J+1                                                         MSB 800
C  MOVE REPLACEMENT STRING TO CENTER OF PAGE FOR OUTPUT              MSB 805
380    IF (N2-36) 390,390,400                                        MSB 810
390    K=40                                                          MSB 815
400    K1=K+N22-1                                                    MSB 820
       J=N2+3                                                        MSB 825
       DO 410 L=J,78                                                 MSB 830
410    IB(L)=IA(47)                                                  MSB 835
       IB(1)=IA(41)                                                  MSB 840
       IB(N2+2)=IA(41)                                               MSB 845
       IB(K-1)=IA(41)                                                MSB 850
       IB(K1+1)=IA(41)                                               MSB 855
       IF (N22) 440,440,420                                          MSB 860
420    DO 430 I=N3,N4                                                MSB 865
```

```
      IB(K)=IC(I)                                                    MSB 870
430   K=K+1                                                          MSB 875
      N3=N3+N22                                                      MSB 880
440   WRITE (IOTAPE,1030) (IB(J),J=1,77),KB,KT                       MSB 885
      GO TO 30                                                       MSB 890
450   IF (IEND) 460,470,460                                          MSB 895
460   STOP                                                           MSB 900
C  SORT SUBSTITUTE TABLES TO PUT LONGEST STRING AT BEGINNING OF TABLE MSB 905
470   N44=N4                                                         MSB 910
      ICNOW=N4+1                                                     MSB 915
      NFG2(LT)=N1                                                    MSB 920
      DO 590 LT=1,99                                                 MSB 925
      IF (NFG2(LT)-NFG1(LT)-4) 590,480,480                           MSB 930
480   N1=NFG2(LT)                                                    MSB 935
      NO=NFG1(LT)                                                    MSB 940
      NO=NO+3                                                        MSB 945
      IF (N1-4) 600,490,490                                          MSB 950
490   N7=N1+2                                                        MSB 955
500   N3=NSFG(LT)                                                    MSB 960
      K1=0                                                           MSB 965
      N7=N7-2                                                        MSB 970
      IF (N7-4) 510,520,520                                          MSB 975
510   N7=N7+2                                                        MSB 980
520   DO 580 I=NO,N7,2                                               MSB 985
      N2=NS(I-3)+NS(I-2)                                             MSB 990
      N22=NS(I-1)+NS(I)                                              MSB 995
      IF (NS(I-3)-NS(I-1)) 540,530,530                               MSB1000
530   N3=N3+N2                                                       MSB1005
      GO TO 580                                                      MSB1010
540   N4=NS(I-3)                                                     MSB1015
      NS(I-3)=NS(I-1)                                                MSB1020
      NS(I-1)=N4                                                     MSB1025
      N4=NS(I)                                                       MSB1030
      NS(I)=NS(I-2)                                                  MSB1035
      NS(I-2)=N4                                                     MSB1040
      IN2=I/2                                                        MSB1045
      N4=NT(IN2)                                                     MSB1050
      NT(IN2)=NT(IN2-1)                                              MSB1055
      NT(IN2-1)=N4                                                   MSB1060
      N4=NB(IN2)                                                     MSB1065
      K1=K1+1                                                        MSB1080
      N4=N3+N2-1                                                     MSB1085
      K=0                                                            MSB1090
      DO 550 J=N3,N4                                                 MSB1095
      K=K+1                                                          MSB1100
550   IB(K)=IC(J)                                                    MSB1105
      DO 560 J=1,N22                                                 MSB1110
      K=N3+J-1                                                       MSB1115
      N6=N4+J                                                        MSB1120
560   IC(K)=IC(N6)                                                   MSB1125
      N3=N3+N22                                                      MSB1130
      DO 570 J=1,N2                                                  MSB1135
      K=N3+J-1                                                       MSB1140
570   IC(K)=IB(J)                                                    MSB1145
580   CONTINUE                                                       MSB1150
      IF (K1) 590,590,500                                            MSB1155
590   CONTINUE                                                       MSB1160
600   WRITE (IOTAPE,990) ,IA(6),IA(9),IA(14),IA(9),IA(19)            MSB1165
```

```
      DO 640 J=1,99                                                    MSB1170
      IF (NFG1(J)) 610,640,610                                         MSB1175
610   WRITE (IOTAPE,1070) J,NFG1(J),NFG2(J),NSFG(J)                    MSB1180
C   IF ICHK = -1  WRITE OUT SORTED SUBSTITUTE LIST                     MSB1185
      IF (ICBK) 620,640,640                                            MSB1190
620   NGF1=NFG1(J)+1                                                   MSB1195
      NGF2=NFG2(J)                                                     MSB1200
      N3=NSFG(J)                                                       MSB1205
      DO 630 I=NGF1,NGF2,2                                             MSB1210
      N2=NS(I-1)                                                       MSB1215
      N22=NS(I)                                                        MSB1220
      N4=N3+N2-1                                                       MSB1225
      N5=N4+1                                                          MSB1230
      N6=N4+N22                                                        MSB1235
      NI1=N2+N22+7                                                     MSB1240
      I2=I/2                                                           MSB1245
      WRITE (IOTAPE,1080) N2,IA(41),(IC(N),N=N3,N4),IA(41),IA(47),IA(47)MSB1250
     1,IA(41),(IC(M),M=N5,N6),IA(41),(IA(47),L=NI1,80),NB(I2),NT(I2)   MSB1255
630   N3=N3+N2+N22                                                     MSB1260
640   CONTINUE                                                         MSB1265
      CALL CLOCK                                                       MSB1270
      DO 650 I=1,500                                                   MSB1275
650   IB(I)=IA(47)                                                     MSB1280
C     END OF INITIALIZATION, RETURN TO MAIN PROGRAM                    MSB1285
      RETURN                                                           MSB1290
C     CALLED WITH LINE IN IB TO BE EDITED                              MSB1295
660   IB(IW+1)=IA(47)                                                  MSB1300
      LT=MODE                                                          MSB1305
      IF (LOCK1) 670,680,670                                           MSB1310
670   CALL SUNLK (IA,IB,IW,IOTAPE)                                     MSB1315
680   K2=1                                                             MSB1320
      ILK=0                                                            MSB1325
690   N6=0                                                             MSB1330
      N7=0                                                             MSB1335
      DO 760 K=K2,IW                                                   MSB1340
C     CHECK FOR EMPTY TABLE                                            MSB1345
      IF (NFG2(LT)-NFG1(LT)) 760,760,700                               MSB1350
700   NGF1=NFG1(LT)+1                                                  MSB1355
      NGF2=NFG2(LT)                                                    MSB1360
      N3=NSFG(LT)                                                      MSB1365
C     SEARCH FOR OCCURRENCE OF SEARCH STRINGS                          MSB1370
      DO 750 I=NGF1,NGF2,2                                             MSB1375
      IF (IB(K)-IC(N3)) 750,710,750                                    MSB1380
710   N2=NS(I-1)                                                       MSB1385
      N22=NS(I)                                                        MSB1390
      K1=K                                                             MSB1395
      N4=N3+N2-1                                                       MSB1400
      IZ=0                                                             MSB1405
      DO 740 J=N3,N4                                                   MSB1410
      IF (IC(J)-IA(89)) 730,720,730                                    MSB1415
720   IZ=IZ+1                                                          MSB1420
      IX(IZ)=IB(K1)                                                    MSB1425
      GO TO 740                                                        MSB1430
730   IF (IB(K1)-IC(J)) 750,740,750                                    MSB1435
740   K1=K1+1                                                          MSB1440
      N7=I                                                             MSB1445
      N6=N3                                                            MSB1450
      GO TO 770                                                        MSB1455
750   N3=N3+NS(I)+NS(I-1)                                              MSB1460
760   CONTINUE                                                         MSB1465
      GO TO 960                                                        MSB1470
770   IF (K1-IW) 790,790,780                                           MSB1475
780   IW=K1                                                            MSB1480
      ILK=1                                                            MSB1485
790   K1=K                                                             MSB1490
      IN2=I/2                                                          MSB1495
      KB=NB(IN2)                                                       MSB1500
```

```
      N2=NS(N7-1)                                                       MSB1505
      N22=NS(N7)                                                        MSB1510
      N3=N6+N2                                                          MSB1515
      N4=N3+N22-1                                                       MSB1520
      N5=N22-N2                                                         MSB1525
      IF (N5) 930,800,890                                               MSB1530
C  INSERT REPLACEMENT STRING                                            MSB1535
800   DO 850 J=N3,N4                                                    MSB1540
      IF (IC(J)-IA(90)) 840,810,840                                     MSB1545
810   IY=IY+1                                                           MSB1550
      IF (IY-IZ) 830,830,820                                            MSB1555
820   WRITE (IOTAPE,1090)                                               MSB1560
      IY=IZ                                                             MSB1565
830   IB(K1)=IX(IY)                                                     MSB1570
      GO TO 850                                                         MSB1575
840   IB(K1)=IC(J)                                                      MSB1580
850   K1=K1+1                                                           MSB1585
860   IF (ILK) 870,870,960                                              MSB1590
870   K2=K1-KB                                                          MSB1595
      IF (NT(I)) 880,690,880                                            MSB1600
880   LT=NT(I)                                                          MSB1605
      GO TO 690                                                         MSB1610
890   IF (IW+N5-2999) 910,910,900                                       MSB1615
900   WRITE (IOTAPE,1100)                                               MSB1620
      WRITE (IOTAPE,990) ,(IB(J),J=1,IW)                                MSB1625
      STOP                                                              MSB1630
C  MAKE ROOM FOR LARGER STRING                                          MSB1635
910   IW=IW+N5                                                          MSB1640
      K2=IW                                                             MSB1645
      DO 920 J=K1,IW                                                    MSB1650
      K9=K2-N5                                                          MSB1655
      IB(K2)=IB(K9)                                                     MSB1660
920   K2=K2-1                                                           MSB1665
      GO TO 800                                                         MSB1670
C  SHORTEN RECORD FOR SHORTER STRING                                    MSB1675
930   DO 940 J=K1,IW                                                    MSB1680
      K9=J-N5                                                           MSB1685
940   IB(J)=IB(K9)                                                      MSB1690
      K9=IW+N5+1                                                        MSB1695
      DO 950 J=K9,IW                                                    MSB1700
950   IB(J)=IA(47)                                                      MSB1705
      IW=IW+N5                                                          MSB1710
      IF (N22) 860,860,800                                              MSB1715
960   IF (LOCK2) 970,980,970                                            MSB1720
970   CALL SULOCK (IA,IB,IW,IOTAPE)                                     MSB1725
980   RETURN                                                            MSB1730
C                                                                       MSB1735
990   FORMAT (1X,131A1)                                                 MSB1740
1000  FORMAT (3A1,I1)                                                   MSB1745
1010  FORMAT (1X,3A1,I1)                                                MSB1750
1020  FORMAT (77A1,I1,I2)                                               MSB1755
1030  FORMAT (1X,77A1,2I2)                                              MSB1760
1040  FORMAT (1X,2A1,23H IS A BAD TABLE NUMBER.)                        MSB1765
1050  FORMAT (15HOTHE CHARACTER ,A1,52H DID NOT APPEAR FOUR TIMES ON THEMSB1770
     1 CARD BELOW.  STOP./1X,80A1)                                      MSB1775
1060  FORMAT (65HOSUBSTITUTE LIST IS TOO LONG.  MAXIMUM CHARACTER LENGTHMSB1780
     1 OF IC IS ,I6,30H MAXIMUM NUMBER OF PHRASES IS ,I5,9H CURRENT ,11HMSB1785
     2VALUES ARE ,2I8,7H STOP. )                                        MSB1790
1070  FORMAT (6H TABLE,I3,10H BEGINS AT,I6,8H ENDS AT,I6,15H FIRST CHAR MSB1795
     1 IS,I6)                                                           MSB1800
1080  FORMAT (1X,1I6,80A1,3I6)                                          MSB1805
1090  FORMAT (3H   ,/,65H   ***YOU ARE WRITING MORE MATCH CHARACTERS THAMSB1810
     1N YOU READ. ***** )                                               MSB1815
1100  FORMAT (87HOTHE LINE FOLLOWING WOULD HAVE EXCEEDED 3000 CHARACTERSMSB1820
     1 IF SUBSTITUTION HAD CONTINUED  )                                 MSB1825
      END                                                               MSB1830
```

```
C
      SUBROUTINE SUNLK(IA,IB,IW,IOTAPE)                                 SSUK  10
      DIMENSION IA(86),IB(999)                                          SSUK  20
      MAXIW=998                                                         SSUK  30
      L=0                                                               SSUK  40
      J=0                                                               SSUK  50
      K=0                                                               SSUK  60
      DO 60 I=1,IW                                                      SSUK  70
      IF (IB(I)-IA(83)) 40,20,40                                        SSUK  80
20    L=L+1                                                             SSUK  90
      IF (L-J-1) 30,60,30                                               SSUK 100
30    K=1                                                               SSUK 110
      GO TO 60                                                          SSUK 120
40    IF (IB(I)-IA(85)) 60,50,60                                        SSUK 130
50    J=J+1                                                             SSUK 140
      IF (L-J) 30,60,30                                                 SSUK 150
60    CONTINUE                                                          SSUK 160
      IF (L-J) 80,70,90                                                 SSUK 170
70    IF (K) 80,120,80                                                  SSUK 180
80    WRITE (IOTAPE,280)                                                SSUK 190
      GO TO 150                                                         SSUK 200
90    IF (IA(83)-IA(85)) 80,100,80                                      SSUK 210
100   K=2*(L/2)-L                                                       SSUK 220
      IF (K) 110,120,110                                                SSUK 230
110   IW=IW+1                                                           SSUK 240
      IB(IW)=IA(85)                                                     SSUK 250
120   J=1                                                               SSUK 260
130   IF (IB(J)-IA(83)) 140,160,140                                     SSUK 270
140   J=J+1                                                             SSUK 280
      IF (J-(IW+1)) 130,150,150                                         SSUK 290
150   RETURN                                                            SSUK 300
160   IF (IB(J+1)-IA(85)) 190,170,190                                   SSUK 310
170   J=J+2                                                             SSUK 320
      DO 180 I=J,IW                                                     SSUK 330
180   IB(I-1)=IB(I)                                                     SSUK 340
      GO TO 220                                                         SSUK 350
190   IF (IB(J+2)-IA(85)) 230,200,230                                   SSUK 360
200   J=J+3                                                             SSUK 370
      DO 210 I=J,IW                                                     SSUK 380
210   IB(I-1)=IB(I)                                                     SSUK 390
220   IW=IW-1                                                           SSUK 400
      IB(IW+1)=IA(47)                                                   SSUK 410
      J=J-1                                                             SSUK 420
      GO TO 130                                                         SSUK 430
230   IF ((IW+1)-MAXIW) 250,250,240                                     SSUK 440
240   WRITE (IOTAPE,270)                                                SSUK 450
      GO TO 150                                                         SSUK 460
250   IW=IW+1                                                           SSUK 470
      J=J+3                                                             SSUK 480
      K=IW                                                              SSUK 490
      DO 260 L=J,IW                                                     SSUK 500
      IB(K)=IB(K-1)                                                     SSUK 510
260   K=K-1                                                             SSUK 520
      J=J-1                                                             SSUK 530
      IB(J)=IA(83)                                                      SSUK 540
      GO TO 160                                                         SSUK 550
270   FORMAT (116HOTHE WORK ON THE FOLLOWING LINE WAS HALTED JUST BEFORESSUK 560
     1 THE MAXIMUM CHARACTER LINE LIMIT WAS EXCEEDED IN  SUNLK      )   SSUK 570
280   FORMAT (69HOTHE FOLLOWING LINE DID NOT CONTAIN A BALANCED SET OF SSSUK 580
     1HIFT SYMBOLS. )                                                  SSUK 590
      END                                                               SSUK 600
```

```
C
      SUBROUTINE SULOCK(IA,IB,IW,IOTAPE)                                SSLK  10
      DIMENSION IA(86),IB(999)                                          SSLK  20
      MAXIW=998                                                         SSLK  30
      J=1                                                               SSLK  40
20    IF (IB(J)-IA(84)) 30,60,30                                        SSLK  50
30    J=J+1                                                             SSLK  60
      IF (J-(IW+1)) 20,20,50                                            SSLK  70
40    WRITE (IOTAPE,140)                                                SSLK  80
50    RETURN                                                            SSLK  90
60    J=J+2                                                             SSLK 100
70    IF (IB(J)-IA(84)) 110,80,110                                      SSLK 110
80    IW=IW-1                                                           SSLK 120
      DO 90 K=J,IW                                                      SSLK 130
90    IB(K)=IB(K+1)                                                     SSLK 140
      IB(IW+1)=IA(47)                                                   SSLK 150
      J=J+1                                                             SSLK 160
      IF (J-IW) 70,70,100                                               SSLK 170
100   IW=IW+1                                                           SSLK 180
      IB(IW)=IA(86)                                                     SSLK 190
      GO TO 50                                                          SSLK 200
110   IF (IW-MAXIW) 120,100,40                                          SSLK 210
120   IW=IW+1                                                           SSLK 220
      J=J+1                                                             SSLK 230
      K=IW                                                              SSLK 240
      IB(IW+1)=IA(47)
      IB(IW+2)=IA(47)
      IB(IW+3)=IA(47)
      DO 130 L=J,IW                                                     SSLK 250
      IB(K)=IB(K-1)                                                     SSLK 260
130   K=K-1                                                             SSLK 270
      IB(J-1)=IA(86)                                                    SSLK 280
      GO TO 20                                                          SSLK 290
140   FORMAT(116HOTBE WORK ON THE FOLLOWING LINE WAS HALTED JUST BEFORE
     1 THE MAXIMUM CHARACTER LINE LIMIT WAS EXCEEDED IN  SULOCK      )  SSLK 310
      END                                                               SSLK 320



C     MULTIPLE TABLE EXTENDED REFORM SUBROUTINE                         MXR   5
C                                                                       MXR  10
C     THIS REFORM PROGRAM ACCEPTS A LINE OF CHARACTERS, BREAKS IT INTO  MXR  15
C     PIECES, AND REARRANGES THE PIECES ACCORDING TO ONE OF 20 FORMATS, MXR  20
C     INSERTING AD HOC STRINGS.  INPUT RECORDS MAY BE FIXED-FIELD, FREE-MXR  25
C     FIELD STRUCTURED, OR UNSTRUCTURED FLAGGED.  OUTPUT MAY BE FIXED-  MXR  30
C     FIELD OR FREE-FIELD.                                              MXR  35
C                                                                       MXR  40
      SUBROUTINE MXREFM (M,KARD,LBUF,JWIDTH)                            MXR  45
C                                                                       MXR  50
C     THE ARGUMENTS IN THE CALL TO MXREFM ARE --                        MXR  55
C     M - IF SET TO ZERO, THE ROUTINE INITIALIZES ITSELF BY READING     MXR  60
C     CONTROL CARDS.  IF POSITIVE, IT IDENTIFIES THE FORMAT TO BE USED. MXR  65
C     KARD - THE CARD IMAGE OR LINE TO BE REFORMATTED                   MXR  70
C     LBUF - THE BUFFER WHICH RECEIVES THE REFORMATTED OUTPUT           MXR  75
C     JWIDTH - ON ENTRY, THE MAXIMUM AVAILABLE LENGTH OF LBUF, WHICH IF MXR  80
C     EXCEEDED WILL CAUSE AN ERROR RETURN - ON RETURN, THE ACTUAL LENGTHMXR  85
C     OF THE OUTPUT LINE.                                               MXR  90
C                                                                       MXR  95
C     STRINGS ARE NUMBERED IN THE ORDER OF THEIR DEFINITION BY THE      MXR 100
C     CONTROL CARDS.  PIECES OF AN INPUT RECORD ARE NUMBERED (WITHIN THEMXR 105
C     RECORD) IN THE ORDER OF THEIR DEFINITION FOR A FIXED-FIELD OR     MXR 110
C     FLAGGED RECORD, AND FROM LEFT TO RIGHT FOR A STRUCTURED FREE-FIELDMXR 115
C     RECORD.                                                           MXR 120
C                                                                       MXR 125
```

```
C     THE CONTROL CARDS ARE AS FOLLOWS --                                    MXR 130
C     THE FIRST CONTROL CARD SHOULD HAVE THE 26 LETTERS, FROM A TO Z, INMXR 135
C     COLUMNS 1 TO 26, AND THE DIGITS FROM 0 TO 9 IN COLUMNS 27-36.          MXR 140
C     COLUMN 47 IS BLANK.  COLUMN 50 CONTAINS A SCAN TERMINATOR, WHICH       MXR 145
C     ENDS THE SCANNING OF A CARD.  THE SYMBOL IN COLUMN 52 INDICATES        MXR 150
C     THAT THE CARD ON WHICH IT APPEARS IS CONTINUED ONTO THE NEXT CARD.MXR 155
C                                                                            MXR 160
C     THE SECOND CONTROL CARD SPECIFIES THE NUMBER OF IDFFERENT FORMATS MXR 165
C     TO BE READ.                                                            MXR 170
C                                                                            MXR 175
C     THEN THE PROGRAM READS IN THE STRINGS.  EACH IS ON A SEPARATE          MXR 180
C     CARD, WITH ITS END MARKED BY THE SCAN TERMINATOR.  THERE MAY BE UPMXR 185
C     UP TO 99 STRINGS.  A CARD WITH A TERMINATOR IN COLUMN 1 ENDS THE       MXR 190
C     READING OF STRINGS.                                                    MXR 195
C                                                                            MXR 200
C     THEN THE INPUT FORMAT DEFINITION FOR EACH FORMAT IS READ.  THE         MXR 205
C     FIRST NUMBER IS THE FORMAT NUMBER (1-20) - THESE SHOULD BE IN          MXR 210
C     ASCENDING ORDER.  THE NEXT NUMBER SPECIFIES THE FORMAT TYPE.  A 1 MXR 215
C     INDICATES FIXED-FIELD RECORDS, A 2 FREE-FIELD RECORDS WITH A 1-3 MXR 220
C     CHARACTER SEPARATOR, AND 3 OR 4 FOR FLAGGED RECORDS.                   MXR 225
C     IF FIXED-FIELD RECORDS ARE SPECIFIED, PAIRS OF NUMBERS FOLLOW.         MXR 230
C     THE FIRST NUMBER IN THE PAIR IS THE CARD COLUMN IN WHICH THE FIELDMXR 235
C     BEGINS.  THE SECOND IS ITS WIDTH.  FIELDS MAY BE CONTIGUOUS,           MXR 240
C     MAY OVERLAP, OR MAY FAIL TO MEET.                                      MXR 245
C     IF PIECES ARE DEFINED BY FLAGS, THE FLAGS FOLLOW.  EACH MUST BE        MXR 250
C     BRACKETED BY SOME CHARACTER, AS /AUTHORS/  *CODEN*                     MXR 255
C     IF THE RECORD IS TYPE 2, THE SEPARATOR (1-3 NON-NUMERIC                MXR 260
C     CHARACTERS) COMES NEXT.  IF IT IS MISSING, FIELDS ARE DELIMITED BYMXR 265
C     ONE OR MORE CONSECUTIVE BLANKS.  THE SEPARATOR IS FOLLOWED BY THE MXR 270
C     NUMBER OF FIELDS.                                                      MXR 275
C                                                                            MXR 280
C     THE LAST SET OF CONTROL CARDS DEFINES THE OUTPUT FORMAT.  THE          MXR 285
C     FIRST NUMBER IS THE FORMAT OR TYPE NUMBER (ASCENDING ORDER).  THE MXR 290
C     NEXT NUMBER MAY BE 1 OR 2.  IF 1, IT INDICATES THE INPUT FRAGMENTSMXR 295
C     ARE TO BE PLACED IN THE OUTPUT WITHOUT WIDTH CHANGE.  IF 2, EACH MXR 300
C     PIECE IS TO BE INSERTED IN A FIXED-WIDTH FIELD.                        MXR 305
C     IF THE FIRST OPTION IS ELECTED, THE CARD THEN LISTS WHAT ITEMS AREMXR 310
C     TO BE ASSEMBLED.  A NUMBER CALLS FOR A PIECE FROM THE INPUT.  THE MXR 315
C     LETTER S FOLLOWED BY A NUMBER CALLS FOR A STRING.                      MXR 320
C     ON OPTION 2, THE SAME SYSTEM IS USED, BUT EACH CALL FOR AN INPUT       MXR 325
C     PIECE IS FOLLOWED BY TWO NUMBERS.  THE FIRST IS THE WIDTH OF THE       MXR 330
C     OUTPUT FIELD.  THE SECOND MAY BE 0,1,OR 2.  IF 0, IT SPECIFIES         MXR 335
C     THAT THE PIECE IS TO BE CENTERED IN THE FIELD.  A 1 CALLS FOR LEFTMXR 340
C     ADJUSTMENT AND A 2 FOR RIGHT ADJUSTMENT.                               MXR 345
C                                                                            MXR 350
C     A FREE-FIELD CONTROL CARD READER, INPUT, IS USED BY THIS PROGRAM, MXR 355
C     FREEING THE USER FROM ARBITRARY FORMAT RULES.  IT CAN READ EITHER MXR 360
C     INTEGERS OR ALPHABETIC CHARACTERS.                                     MXR 365
C                                                                            MXR 370
C     -----                                                                  MXR 375
C     PROGRAMMING INFORMATION                                                MXR 380
C                                                                            MXR 385
C     IN AND IOUT ARE THE UNIT NUMBERS FOR THE CARD READER AND PRINTER, MXR 390
C     RESPECTIVELY.                                                          MXR 395
C     IC IS A VECTOR, SHARED WITH OTHER PROGRAMS, FOR STRING STORAGE         MXR 400
C     ICLEN IS ITS SIZE                                                      MXR 405
C     ICNOW POINTS TO THE FIRST AVAILABLE (UNUSED) LOCATION IN IC, AND MXR 410
C     MUST BE ADVANCED AFTER STORING STRINGS IN IC                           MXR 415
C     IABC IS THE IMAGE OF THE IFRST CONTROL CARD (ALPHABET, ETC.)           MXR 420
C     NFSTRT - THE DEFINED STARTING COLUMN OF A FIXED FIELD                  MXR 425
C     NFWDTH - THE DEFINED WIDTH OF A FIXED FIELD                            MXR 430
C     NFORG - THE ACTUAL STARTING COLUMN OF A FIXED OR FREE FIELD            MXR 435
C     NW                                                                     MXR 440
C     NFWIDE - THE ACTUAL WIDTH OF A FIELD                                   MXR 445
C     LIM - THE DEFINED WIDTH OF A FIXED OUTPUT FIELD                        MXR 450
C     ISIDE - THE CENTERING OR ADJUSTMENT INSTRUCTION                        MXR 455
```

```
C      NUMBR - THE STRING OR FIELD SPECIFICATION FOR OUTPUT                    MXR 460
C      TELLS WHETHER A STRING OR A FIELD IS SPECIFIED                          MXR 465
C      MODE1 - THE INPUT OPTION (FIXED,FREE,FLAGGED) FOR A FORMAT              MXR 470
C      MODE2 - THE OUTPUT OPTION FOR A FORMAT                                  MXR 475
C      KSEP - THE SEPARATORS                                                   MXR 480
C      LS - THE NUMBER OF CHARACTERS IN A SEPARATOR                            MXR 485
C      NFTAB - POINTERS TO THE IC POSITION OF THE FLAGS                        MXR 490
C      NUMFL - TABLE MATCHING FLAG NUMBER, FIELD NUMBER                        MXR 495
C      ISSTRT - POINTERS TO THE AD HOC STRINGS IN IC                           MXR 500
C      NFLDS - NUMBER OF INPUT FIELDS IN A RECORD FORMAT                       MXR 505
C      NOUT - NUMBER OF OUTPUT FIELDS IN A RECORD FORMAT                       MXR 510
C      IT1 - CONVERSION TABLE FROM RECORD*FIELD TO ONE NUMBER                  MXR 515
C      IRT1 - POINTERS FOR THE INPUT FORMAT TABLES                             MXR 520
C      IRT2 - POINTERS FOR THE OUTPUT FORMAT TABLES                            MXR 525
C      JDATA - THE INFORMATION RETURNED BY INPUT (INTEGERS, CHARACTERS)        MXR 530
C      KEY - DISTINGUISHES BETWEEN INTEGERS*CHARACTERS IN JDATA                MXR 535
C      NC - NUMBER OF FORMAT TYPES                                             MXR 540
C                                                                              MXR 545
       COMMON /IO/ IN,IOUT                                                     MXR 550
       COMMON /STR/ ICLEN,ICNOW,IC(8000)                                       MXR 555
       COMMON /IA/ IABC(120)                                                   MXR 560
       COMMON /W/ KL,NTABL                                                     MXR 565
       DIMENSION NFSTRT(200),NFWDTH(200),LIM(200),NFORG(50),NFWIDE(50)         MXR 570
       DIMENSION NUMBR(200),MTYPE(200),ISIDE(200),MODE1(20),MODE2(20)          MXR 575
       DIMENSION KSEP(20,3),LS(20),NFTAB(50),NUMFL(200),ISSTRT(99)             MXR 580
       DIMENSION JDATA(60),KEY(60),KARDIN(80),NFLDS(20),NOUT(20)               MXR 585
       DIMENSION KARD(999),LBUF(999),IT1(20,20),IRT1(200),IRT2(200)            MXR 590
10     FORMAT (80A1)                                                           MXR 595
20     FORMAT (11HOTHERE ARE ,I2,23H STRINGS.  THEY ARE -- /)                  MXR 600
30     FORMAT (1X,3H S ,I2,2X,I4,12H CHARACTERS ,100A1)                        MXR 605
40     FORMAT (30HOINPUT INSTRUCTIONS FOR RECORD,I2)                           MXR 610
50     FORMAT (60H RECORD IS FIXED-FIELD.  STARTING COLUMNS AND WIDTHS ARMXR 615
      1E -- )                                                                  MXR 620
60     FORMAT (41H RECORD IS FREE-FIELD.  THE SEPARATOR IS ,3A1)               MXR 625
70     FORMAT (32H RECORD HAS FLAGS.  THEY ARE -- )                            MXR 630
80     FORMAT (I4,2X,9A1)                                                      MXR 635
90     FORMAT (28H FLAGS NEED NOT BE IN ORDER )                                MXR 640
100    FORMAT (10(3X,2I3)/10(3X,2I3))                                          MXR 645
110    FORMAT (31HOOUTPUT INSTRUCTIONS FOR RECORD,I2)                          MXR 650
120    FORMAT (8H STRING ,I2)                                                  MXR 655
130    FORMAT (7H PIECE ,I2,11H OF RECORD ,I2)                                 MXR 660
140    FORMAT (7H PIECE ,I2,11H OF RECORD ,I2,15H ADJUSTED IN A ,I2,25H CMXR 665
      1HARACTER FIELD IN MODE ,I1)                                             MXR 670
150    FORMAT (1HO/1X,80A1/42HOTHE ABOVE CONTROL CARD IS IN ERROR. STOP.)MXR 675
160    FORMAT (49HOTOO MANY STRINGS, OR STRING TERMINATOR MISSING. )           MXR 680
170    FORMAT (44HONON-EXISTENT STRING OR FRAGMENT REQUESTED. )                MXR 685
180    FORMAT (45HOTHE OUTPUT LINE IS TOO LONG.  ERROR RETURN. )               MXR 690
190    FORMAT (64HOA PROGRAM PARAMETER HAS BEEN ASSIGNED AN ILLEGAL VALUEMXR 695
      1.  STOP. )                                                              MXR 700
200    FORMAT (33HOREFORM CONFUSED, CANNOT CONTINUE)                           MXR 705
C      IS THIS INITIALIZATION OR OPERATION                                     MXR 710
       IF (M) 1320,210,860                                                     MXR 715
C      READ CONTROL CARDS                                                      MXR 720
210    JWMAX=JWIDTH                                                            MXR 725
       NC=NTABL                                                                MXR 730
C      READ STRINGS                                                            MXR 735
       K=ICNOW                                                                 MXR 740
       ISSTRT(1)=ICNOW                                                         MXR 745
       JJ=1                                                                    MXR 750
220    READ (IN,10) (KARDIN(JX),JX=1,80)                                       MXR 755
       N=1                                                                     MXR 760
       IF (KARDIN(1)-IABC(50)) 230,280,230                                     MXR 765
230    IF (KARDIN(N)-IABC(50)) 240,270,240                                     MXR 770
240    IC(K)=KARDIN(N)                                                         MXR 775
```

```
      K=K+1                                                                    MXR 780
      IF (K-ICLEN) 250,830,830                                                 MXR 785
250   N=N+1                                                                    MXR 790
      IF (N-80) 230,230,260                                                    MXR 795
260   READ (IN,10) (KARDIN(JX),JX=1,80)                                        MXR 800
      N=1                                                                      MXR 805
      GO TO 230                                                                MXR 810
270   JJ=JJ+1                                                                  MXR 815
      ISSTRT(JJ)=K                                                             MXR 820
C     GET ANOTHER STRING                                                       MXR 825
      GO TO 220                                                                MXR 830
C     END OF STRINGS - LIST THEM                                               MXR 835
280   NSTRNG=JJ-1                                                              MXR 840
      ICNOW=K                                                                  MXR 845
      WRITE (IOUT,20) NSTRNG                                                   MXR 850
      DO 290 J=1,NSTRNG                                                        MXR 855
      N1=ISSTRT(J)                                                             MXR 860
      N2=ISSTRT(J+1)-1                                                         MXR 865
      LL=N2-N1+1                                                               MXR 870
290   WRITE (IOUT,30) J,LL,(IC(JX),JX=N1,N2)                                   MXR 875
C     READ INPUT SPECIFICATIONS                                                MXR 880
      NFL=0                                                                    MXR 885
      N=1                                                                      MXR 890
      NP=1                                                                     MXR 895
      KF=ICNOW                                                                 MXR 900
      IRT1(1)=1                                                                MXR 905
      DO 600 J=1,NC                                                            MXR 910
      READ (IN,10) (KARDIN(JX),JX=1,80)                                        MXR 915
      CALL INPUT (3,KARDIN,JDATA,KEY,80)                                       MXR 920
      I=JDATA(1)                                                               MXR 925
      IF (I-J) 820,300,820                                                     MXR 930
300   IF (KEY(2)-1) 330,310,820                                                MXR 935
310   MODE1(J)=JDATA(2)                                                        MXR 940
      IF (IABS(MODE1(J)-2)-2) 320,320,820                                      MXR 945
320   MI=MODE1(J)+1                                                            MXR 950
C     WHAT OPTION                                                              MXR 955
      GO TO (330,340,410,460,470), MI                                          MXR 960
330   NFLDS(J)=0                                                               MXR 965
      GO TO 590                                                                MXR 970
C     FIXED-FIELD                                                              MXR 975
340   NN=NP                                                                    MXR 980
      WRITE (IOUT,50)                                                          MXR 985
      JN=3                                                                     MXR 990
350   IF (IABS(KEY(JN))-1) 360,360,820                                         MXR 995
360   IF (KEY(JN)) 390,400,370                                                 MXR1000
370   NFSTRT(NP)=JDATA(JN)                                                     MXR1005
      IF (KEY(JN+1)-1) 820,380,820                                             MXR1010
380   NFWDTH(NP)=JDATA(JN+1)                                                   MXR1015
      NP=NP+1                                                                  MXR1020
      JN=JN+2                                                                  MXR1025
      GO TO 350                                                                MXR1030
390   READ (IN,10) (KARDIN(JX),JX=1,80)                                        MXR1035
      CALL INPUT (4,KARDIN,JDATA,KEY,80)                                       MXR1040
      WRITE (IOUT,40) I                                                        MXR1045
      JN=1                                                                     MXR1050
      GO TO 350                                                                MXR1055
400   NFLDS(J)=NP-NN                                                           MXR1060
      NX=NP-1                                                                  MXR1065
      WRITE (IOUT,100) ((NFSTRT(JX),NFWDTH(JX)),JX=NN,NX)                      MXR1070
      GO TO 590                                                                MXR1075
C     FREE-FIELD - GET SEPARATOR                                               MXR1080
410   CALL INPUT (2,KARDIN,JDATA,KEY,80)                                       MXR1085
      DO 420 JJ=1,3                                                            MXR1090
      IF (KEY(JJ+2)-1) 440,440,420                                             MXR1095
420   KSEP(J,JJ)=JDATA(JJ+2)                                                   MXR1100
```

```
      IF (KEY(6)-1) 430,430,820                                    MXR1105
430   LS(J)=3                                                      MXR1110
      GO TO 450                                                    MXR1115
440   LS(J)=JJ-1                                                   MXR1120
      L=LS(J)                                                      MXR1125
450   NN=L*3                                                       MXR1130
      NFLDS(J)=JDATA(NN)                                           MXR1135
      NP=NP*NFLDS(J)                                               MXR1140
      WRITE (IOUT,60) (KSEP(J,JX),JX=1,L)                          MXR1145
      GO TO 590                                                    MXR1150
C     FLAGGED FORMAT                                               MXR1155
460   M=0                                                          MXR1160
      GO TO 480                                                    MXR1165
470   M=1                                                          MXR1170
480   WRITE (IOUT,70)                                              MXR1175
      N=1                                                          MXR1180
      CALL INPUT (-2,KARDIN,JDATA,KEY,K)                           MXR1185
      NN=NP                                                        MXR1190
490   K=K*1                                                        MXR1195
      IF (K-80) 500,500,570                                        MXR1200
500   IF (KARDIN(K)-IABC(52)) 510,560,510                          MXR1205
510   IF (KARDIN(K)-IABC(47)) 520,490,520                          MXR1210
C     BEGINNING OF FLAG                                            MXR1215
520   KX=KARDIN(K)                                                 MXR1220
      NFL=NFL*1                                                    MXR1225
      K=K*1                                                        MXR1230
      NFTAB(NFL)=KF                                                MXR1235
      NUMFL(NP)=NFL                                                MXR1240
      NP=NP*1                                                      MXR1245
      KK=KF                                                        MXR1250
530   IC(KF)=KARDIN(K)                                             MXR1255
      KF=KF*1                                                      MXR1260
      K=K*1                                                        MXR1265
      IF (K-80) 540,540,820                                        MXR1270
540   IF (KARDIN(K)-KX) 530,550,530                                MXR1275
C     END OF FLAG                                                  MXR1280
550   KJ=KF-1                                                      MXR1285
      NFTAB(NFL*1)=KF                                              MXR1290
      WRITE (IOUT,80) N,(IC(JX),JX=KK,KJ)                          MXR1295
      N=N*1                                                        MXR1300
      GO TO 490                                                    MXR1305
560   READ (IN,10) (KARDIN(JX),JX=1,80)                            MXR1310
      K=0                                                          MXR1315
      GO TO 490                                                    MXR1320
C     END OF CARD - NO MORE FLAGS                                  MXR1325
570   NFLDS(J)=N-1                                                 MXR1330
      IF (M) 580,590,580                                           MXR1335
580   WRITE (IOUT,90)                                              MXR1340
590   NN=NFLDS(J)                                                  MXR1345
      NX=NP-NN-1                                                   MXR1350
      IRT1(J*1)=NP                                                 MXR1355
      DO 600 JJ=1,NN                                               MXR1360
      IT1(JJ,J)=NX*JJ                                              MXR1365
600   CONTINUE                                                     MXR1370
      ICNOW=KF                                                     MXR1375
C     READ OUTPUT SPECIFICATIONS                                   MXR1380
      N=1                                                          MXR1385
      IRT2(1)=1                                                    MXR1390
      DO 810 J=1,NC                                                MXR1395
      READ (IN,10) (KARDIN(JX),JX=1,80)                            MXR1400
      CALL INPUT (3,KARDIN,JDATA,KEY,80)                           MXR1405
      I=JDATA(1)                                                   MXR1410
      NJ=N                                                         MXR1415
      IF (I-J) 820,610,820                                         MXR1420
610   WRITE (IOUT,110) I                                           MXR1425
      IF (KEY(2)-1) 640,620,820                                    MXR1430
```

```
620   MODE2(J)=JDATA(2)                                          MXR1435
      L=1                                                        MXR1440
      IF (IABS(MODE2(J)-1)-1) 630,630,820                        MXR1445
630   IF (MODE2(J)-1) 640,650,660                                MXR1450
640   NFLDS(J)=0                                                 MXR1455
      GO TO 800                                                  MXR1460
650   MX=0                                                       MXR1465
      GO TO 670                                                  MXR1470
660   MX=1                                                       MXR1475
670   JN=3                                                       MXR1480
680   MI=KEY(JN)+2                                               MXR1485
      GO TO (790,800,730,690), MI                                MXR1490
690   IF (JDATA(JN)-IABC(19)) 820,700,820                        MXR1495
C     STRING REQUESTED                                           MXR1500
700   IF (KEY(JN+1)-1) 820,710,820                               MXR1505
710   NN=JDATA(JN+1)                                             MXR1510
      IF (NN-NSTRNG) 720,720,820                                 MXR1515
720   NUMBR(N)=NN                                                MXR1520
      MTYPE(N)=2                                                 MXR1525
      N=N+1                                                      MXR1530
      JN=JN+2                                                    MXR1535
      WRITE (IOUT,120) NN                                        MXR1540
      GO TO 680                                                  MXR1545
C     INPUT PIECE REQUESTED                                      MXR1550
730   N1=JDATA(JN)                                               MXR1555
      NN=IT1(N1,J)                                               MXR1560
      IF (NN) 840,840,740                                        MXR1565
740   NUMBR(N)=N1                                                MXR1570
      MTYPE(N)=1                                                 MXR1575
      IF (MX) 750,780,750                                        MXR1580
C     FIXED OUTPUT FIELD DEFINITION                              MXR1585
750   IF (KEY(JN+1)-1) 820,760,820                               MXR1590
760   IF (IABS(JDATA(JN+2)-1)) 770,770,820                       MXR1595
770   LIM(N)=JDATA(JN+1)                                         MXR1600
      ISIDE(N)=JDATA(JN+2)                                       MXR1605
      JN=JN+3                                                    MXR1610
      WRITE (IOUT,140) N1,J,LIM(N),ISIDE(N)                      MXR1615
      N=N+1                                                      MXR1620
      GO TO 680                                                  MXR1625
780   JN=JN+1                                                    MXR1630
      WRITE (IOUT,130) N1,J                                      MXR1635
      N=N+1                                                      MXR1640
      GO TO 680                                                  MXR1645
790   READ (IN,10) (KARDIN(JX),JX=1,80)                          MXR1650
      CALL INPUT (3,KARDIN,JDATA,KEY,80)                         MXR1655
      JN=1                                                       MXR1660
      GO TO 680                                                  MXR1665
800   NOUT(J)=N-NJ                                               MXR1670
      IRT2(J+1)=N                                                MXR1675
810   CONTINUE                                                   MXR1680
      RETURN                                                     MXR1685
C     ERRORS                                                     MXR1690
      WRITE (IOUT,190)                                           MXR1695
      GO TO 850                                                  MXR1700
820   WRITE (IOUT,150) (KARDIN(JX),JX=1,80)                      MXR1705
      GO TO 850                                                  MXR1710
C     IC OVERFLOW                                                MXR1715
830   WRITE (IOUT,160)                                           MXR1720
      GO TO 850                                                  MXR1725
C     SORRY, NO SUCH NUMBER                                      MXR1730
840   WRITE (IOUT,170)                                           MXR1735
```

```
850   WRITE (IOUT,200)                                              MXR1740
      STOP                                                          MXR1745
C                                                                   MXR1750
C     REFORMAT A LINE                                               MXR1755
860   J=M                                                           MXR1760
      DO 870 I=1,JWIDTH                                             MXR1765
870   LBUF(I)=IABC(47)                                              MXR1770
C     SET UP FIELD POINTERS                                         MXR1775
      KNTR=1                                                        MXR1780
      N=1                                                           MXR1785
      I=MODE1(J)                                                    MXR1790
      NN=NFLDS(J)                                                   MXR1795
      DO 880 J=1,NN                                                 MXR1800
880   NFWIDE(J)=0                                                   MXR1805
      NP=IRT1(J)                                                    MXR1810
      MI=I+1                                                        MXR1815
C     WHAT INPUT MODE                                               MXR1820
      GO TO (1100,1000,890,1020,1020), MI                           MXR1825
C     SEARCH FOR SEPARATORS                                         MXR1830
890   DO 900 JJ=1,JWIDTH                                            MXR1835
      JX=JWIDTH+1-JJ                                                MXR1840
      IF (KARD(JX)-IABC(47)) 900,910,900                            MXR1845
900   CONTINUE                                                      MXR1850
      LEN=0                                                         MXR1855
      GO TO 1100                                                    MXR1860
910   LEN=JJ                                                        MXR1865
      KK=LEN-1                                                      MXR1870
      NJ=N+NN-1                                                     MXR1875
      II=LS(J)                                                      MXR1880
      DO 980 JJ=1,LEN                                               MXR1885
      NFORG(1)=1                                                    MXR1890
      IF (II) 940,940,920                                           MXR1895
920   DO 930 JN=1,II                                                MXR1900
      JZ=JJ+JN-1                                                    MXR1905
      IF (KARD(JZ)-KSEP(J,JN)) 980,930,980                          MXR1910
930   CONTINUE                                                      MXR1915
      GO TO 970                                                     MXR1920
940   IF (KARD(JJ)-IABC(47)) 980,950,980                            MXR1925
950   IF (NFORG(N)-JJ) 970,960,970                                  MXR1930
960   NFORG(N)=JJ+LS(J)                                             MXR1935
      NFWIDE(N)=KK-JJ                                               MXR1940
      GO TO 980                                                     MXR1945
970   NFWIDE(N)=JJ-NFORG(N)                                         MXR1950
      N=N+1                                                         MXR1955
      NFORG(N)=JJ+1                                                 MXR1960
      NFWIDE(N)=KK-JJ                                               MXR1965
980   CONTINUE                                                      MXR1970
C     BLANK FIELDS                                                  MXR1975
      DO 990 JJ=N,NJ                                                MXR1980
      NFWIDE(JJ)=0                                                  MXR1985
990   NFORG(JJ)=KK                                                  MXR1990
      N=NJ+1                                                        MXR1995
      GO TO 1100                                                    MXR2000
C     FIXED-FIELD - USE DEFINED FORMAT                              MXR2005
1000  NN=NFLDS(J)                                                   MXR2010
      NJ=N+NN-1                                                     MXR2015
      DO 1010 JJ=N,NJ                                               MXR2020
      I=JJ+NP-1                                                     MXR2025
```

```
      NFORG(JJ)=NFSTRT(I)                                              MXR2030
1010  NFWIDE(JJ)=NFWDTH(I)                                             MXR2035
      N=NJ+1                                                           MXR2040
      GO TO 1100                                                       MXR2045
C     LOOK FOR FLAGS                                                   MXR2050
1020  NN=NFLDS(J)                                                      MXR2055
      K=1                                                              MXR2060
      NJ=N+NN-1                                                        MXR2065
      NFORG(N)=K                                                       MXR2070
      NFWIDE(N)=JWIDTH                                                 MXR2075
      NF=0                                                             MXR2080
1030  CONTINUE                                                         MXR2085
      DO 1080 JJ=K,JWIDTH                                              MXR2090
C     DOES IT MATCH                                                    MXR2095
      DO 1070 JN=N,NJ                                                  MXR2100
      IZ=NUMFL(JN)                                                     MXR2105
      KF=NFTAB(IZ)                                                     MXR2110
      KS=NFTAB(IZ+1)-1                                                 MXR2115
      DO 1040 JZ=KF,KS                                                 MXR2120
      II=JJ+JZ-KF                                                      MXR2125
      IF (IC(JZ)-KARD(II)) 1070,1040,1070                              MXR2130
1040  CONTINUE                                                         MXR2135
      IF (I-3) 1050,1050,1060                                          MXR2140
1050  IF (JN-NF) 1070,1070,1060                                        MXR2145
1060  NFWIDE(NF)=JJ-NFORG(NF)                                          MXR2150
      NF=JN                                                            MXR2155
      NFORG(NF)=JJ+KS-KF+1                                             MXR2160
      GO TO 1090                                                       MXR2165
1070  CONTINUE                                                         MXR2170
1080  CONTINUE                                                         MXR2175
      NFWDTH(NF)=JWIDTH-NFORG(NF)                                      MXR2180
      GO TO 1100                                                       MXR2185
1090  K=NFORG(NF)                                                      MXR2190
      GO TO 1030                                                       MXR2195
1100  CONTINUE                                                         MXR2200
C     ASSEMBLE THE OUTPUT                                              MXR2205
      I=MODE2(J)                                                       MXR2210
      N=IRT2(J)                                                        MXR2215
      IF (I) 1310,1310,1110                                            MXR2220
1110  NJ=NOUT(J)+N-1                                                   MXR2225
      KNTR=1                                                           MXR2230
      DO 1300 JJ=N,NJ                                                  MXR2235
C     WHAT TYPE ITEM                                                   MXR2240
      IF (MTYPE(JJ)-1) 1300,1140,1120                                  MXR2245
C     TRANSFER A STRING                                                MXR2250
1120  NX=NUMBR(JJ)                                                     MXR2255
      K1=ISSTRT(NX)                                                    MXR2260
      K2=ISSTRT(NX+1)-1                                                MXR2265
      DO 1130 JZ=K1,K2                                                 MXR2270
      LBUF(KNTR)=IC(JZ)                                                MXR2275
1130  KNTR=KNTR+1                                                      MXR2280
      GO TO 1300                                                       MXR2285
C     A PIECE OF THE INPUT LINE                                        MXR2290
1140  N1=NUMBR(JJ)                                                     MXR2295
      N2=NFORG(N1)                                                     MXR2300
      N3=NFWIDE(N1)                                                    MXR2305
      IF (KNTR+N3-JWMAX) 1150,1150,1330                                MXR2310
1150  IF (I-2) 1160,1170,1160                                          MXR2315
```

```
C       LL IS THE OUTPUT WIDTH                                          MXR2320
1160    LL=N3                                                           MXR2325
        NX=0                                                            MXR2330
        GO TO 1180                                                      MXR2335
1170    LL=LIM(JJ)                                                      MXR2340
        NX=LL-N3                                                        MXR2345
1180    IF (LL) 1190,1300,1190                                          MXR2350
1190    DO 1290 JZ=1,LL                                                 MXR2355
        IF (NX) 1200,1270,1270                                          MXR2360
1200    IF (ISIDE(JJ)-1) 1210,1230,1250                                 MXR2365
C       CENTERING                                                       MXR2370
1210    N4=(LL+1)/2                                                     MXR2375
        N5=N3/2                                                         MXR2380
        IF (IABS(JZ-N4)-N5) 1220,1220,1280                              MXR2385
1220    II=JZ+N2+N5-N4-1                                                MXR2390
        LBUF(KNTR)=KARD(II)                                             MXR2395
        GO TO 1290                                                      MXR2400
C       LEFT-ADJUST                                                     MXR2405
1230    IF (JZ-N3) 1240,1240,1280                                       MXR2410
1240    II=JZ+N2-1                                                      MXR2415
        LBUF(KNTR)=KARD(II)                                             MXR2420
        GO TO 1290                                                      MXR2425
C       RIGHT-ADJUST                                                    MXR2430
1250    IF (JZ-NX) 1280,1280,1260                                       MXR2435
1260    II=JZ+N2-NX-1                                                   MXR2440
        LBUF(KNTR)=KARD(II)                                             MXR2445
        GO TO 1290                                                      MXR2450
C       MUST TRUNCATE TO FIT THE FIELD                                  MXR2455
1270    II=JZ+N2-1                                                      MXR2460
        LBUF(KNTR)=KARD(II)                                             MXR2465
        GO TO 1290                                                      MXR2470
1280    LBUF(KNTR)=IABC(47)                                             MXR2475
1290    KNTR=KNTR+1                                                     MXR2480
1300    CONTINUE                                                        MXR2485
1310    CONTINUE                                                        MXR2490
        JWIDTH=KNTR-1                                                   MXR2495
        RETURN                                                          MXR2500
1320    GO TO 850                                                       MXR2505
C       NO ROOM                                                         MXR2510
1330    WRITE (IOUT,180)                                                MXR2515
        RETURN                                                          MXR2520
        END                                                             MXR2525-
```

FORM NBS-114A (1-71)

U.S. DEPT. OF COMM. BIBLIOGRAPHIC DATA SHEET	1. PUBLICATION OR REPORT NO. NBS-TN-700	2. Gov't Accession No.	3. Recipient's Accession No.

4. TITLE AND SUBTITLE

COMBO: A General-Purpose Program for Searching, Annotating, Encoding-Decoding, and Reformatting Data Files

5. Publication Date: September 1972

6. Performing Organization Code

7. AUTHOR(S)

Robert McClenon and Joseph Hilsenrath

8. Performing Organization

9. PERFORMING ORGANIZATION NAME AND ADDRESS

NATIONAL BUREAU OF STANDARDS
DEPARTMENT OF COMMERCE
WASHINGTON, D.C. 20234

10. Project/Task/Work Unit No.

11. Contract/Grant No.

12. Sponsoring Organization Name and Address

Same as No. 9.

13. Type of Report & Period Covered: Final

14. Sponsoring Agency Code

15. SUPPLEMENTARY NOTES

16. ABSTRACT (A 200-word or less factual summary of most significant information. If document includes a significant bibliography or literature survey, mention it here.)

COMBO, a FORTRAN program for searching magnetic tape files, generating reports, and reformatting the file, is described and listed. The program is capable of reading separate card images from a file blocked in physical records and recognizing logical blocks marked by a fixed-field ID. Up to 99 different types of lines, each with its own format, can be recognized by examining a special code or label. The program can be instructed to search for the occurrence of each of certain character strings, using a different list for each type of line and two levels of Boolean logic. Lines can be broken into pieces, using either a fixed-field format or a single separator or flags to define the pieces, and the pieces can be rearranged, with labels or comments optionally inserted between them. Editing, in which specified strings are replaced by other strings, can also be performed. The program can accommodate a variable number of cards of each type per block. It was assembled from general-purpose subroutines of modular design and is substantially machine-independent.

17. KEY WORDS (Alphabetical order, separated by semicolons) Alphanumeric data files; data retrieval; editing program; file editor; FORTRAN program; general-purpose modular programming; reformatting program; report generator; searching program.

18. AVAILABILITY STATEMENT

[XX] UNLIMITED.

[ ] FOR OFFICIAL DISTRIBUTION. DO NOT RELEASE TO NTIS.

19. SECURITY CLASS (THIS REPORT): UNCLASSIFIED

21. NO. OF PAGES: 68

20. SECURITY CLASS (THIS PAGE): UNCLASSIFIED

22. Price: 75 cents

*U.S. GOVERNMENT PRINTING OFFICE:1972 477-993

USCOMM-DC 66244-P71

This is one in a series of general-purpose programs for text editing, data retrieval, file management and computer-assisted typesetting. Other programs and subroutines are now being documented for publication in the near future. If you wish to be notified of the next publication in this series, please fill out the post card below.

(Type or print)

Name _____

Affiliation _____

Address _____

City _____ **State** _____ **ZIP Code** _____

◇ I would welcome a notification of the availability of the next publication in the subject matter area of NBS Technical Note 500.

◇ Please send me price and ordering instructions for the magnetic tape version of this and other programs in the series.

Place
stamp
here

National Bureau of Standards
Office of Standard Reference Data
Data Systems Design Group
Room A534, Administration Building
Washington, D.C. 20234

NBS TECHNICAL PUBLICATIONS

PERIODICALS

JOURNAL OF RESEARCH reports National Bureau of Standards research and development in physics, mathematics, and chemistry. Comprehensive scientific papers give complete details of the work, including laboratory data, experimental procedures, and theoretical and mathematical analyses. Illustrated with photographs, drawings, and charts. Includes listings of other NBS papers as issued.

Published in two sections, available separately:

- **Physics and Chemistry**

Papers of interest primarily to scientists working in these fields. This section covers a broad range of physical and chemical research, with major emphasis on standards of physical measurement, fundamental constants, and properties of matter. Issued six times a year. Annual subscription: Domestic, $9.50; $2.25 additional for foreign mailing.

- **Mathematical Sciences**

Studies and compilations designed mainly for the mathematician and theoretical physicist. Topics in mathematical statistics, theory of experiment design, numerical analysis, theoretical physics and chemistry, logical design and programming of computers and computer systems. Short numerical tables. Issued quarterly. Annual subscription: Domestic, $5.00; $1.25 additional for foreign mailing.

TECHNICAL NEWS BULLETIN

The best single source of information concerning the Bureau's measurement, research, developmental, cooperative, and publication activities, this monthly publication is designed for the industry-oriented individual whose daily work involves intimate contact with science and technology—*for engineers, chemists, physicists, research managers, product-development managers, and company executives.* Includes listing of all NBS papers as issued. Annual subscription: Domestic, $3.00; $1.00 additional for foreign mailing.

Bibliographic Subscription Services

The following current-awareness and literature-survey bibliographies are issued periodically by the Bureau: Cryogenic Data Center Current Awareness Service (weekly), Liquefied Natural Gas (quarterly), Superconducting Devices and Materials (quarterly), and Electromagnetic Metrology Current Awareness Service (monthly). Available only from NBS Boulder Laboratories. Ordering and cost information may be obtained from the Program Information Office, National Bureau of Standards, Boulder, Colorado 80302.

NONPERIODICALS

Applied Mathematics Series. Mathematical tables, manuals, and studies.

Building Science Series. Research results, test methods, and performance criteria of building materials, components, systems, and structures.

Handbooks. Recommended codes of engineering and industrial practice (including safety codes) developed in cooperation with interested industries, professional organizations, and regulatory bodies.

Special Publications. Proceedings of NBS conferences, bibliographies, annual reports, wall charts, pamphlets, etc.

Monographs. Major contributions to the technical literature on various subjects related to the Bureau's scientific and technical activities.

National Standard Reference Data Series. NSRDS provides quantitative data on the physical and chemical properties of materials, compiled from the world's literature and critically evaluated.

Product Standards. Provide requirements for sizes, types, quality, and methods for testing various industrial products. These standards are developed cooperatively with interested Government and industry groups and provide the basis for common understanding of product characteristics for both buyers and sellers. Their use is voluntary.

Technical Notes. This series consists of communications and reports (covering both other-agency and NBS-sponsored work) of limited or transitory interest.

Federal Information Processing Standards Publications. This series is the official publication within the Federal Government for information on standards adopted and promulgated under the Public Law 89–306, and Bureau of the Budget Circular A–86 entitled, Standardization of Data Elements and Codes in Data Systems.

Consumer Information Series. Practical information, based on NBS research and experience, covering areas of interest to the consumer. Easily understandable language and illustrations provide useful background knowledge for shopping in today's technological marketplace.

CATALOGS OF NBS PUBLICATIONS

NBS Special Publication 305, Publications of the NBS. 1966-1967. When ordering, include Catalog No. C13.10:305. Price $2.00; 50 cents additional for foreign mailing.

NBS Special Publication 305, Supplement 1, Publications of the NBS, 1968-1969. When ordering, include Catalog No. C13.10:305/Suppl. 1. Price $4.50; $1.25 additional for foreign mailing.

NBS Special Publication 305, Supplement 2, Publications of the NBS, 1970. When ordering, include Catalog No. C13.10:305/Suppl. 2. Price $3.25; 85 cents additional for foreign mailing.

Order NBS publications (except Bibliographic Subscription Services) from: Superintendent of Documents, Government Printing Office, Wash-

U.S. DEPARTMENT OF COMMERCE
National Bureau of Standards
Washington, D.C. 20234

OFFICIAL BUSINESS

Penalty for Private Use, $300

POSTAGE AND FEES PAID
U.S. DEPARTMENT OF COMMERCE
215