ITED STATES
RTMENT OF
MMERCE
LICATION

NBS TECHNICAL NOTE 500

Edit-Insertion Programs for Automatic Typesetting of Computer Printout

U.S.
ARTMENT
OF
MMERCE
National
Bureau of
Standards

NATIONAL BUREAU OF STANDARDS

The National Bureau of Standards[1] was established by an act of Congress March 3, 1901. Today, in addition to serving as the Nation's central measurement laboratory, the Bureau is a principal focal point in the Federal Government for assuring maximum application of the physical and engineering sciences to the advancement of technology in industry and commerce. To this end the Bureau conducts research and provides central national services in four broad program areas. These are: (1) basic measurements and standards, (2) materials measurements and standards, (3) technological measurements and standards, and (4) transfer of technology.

The Bureau comprises the Institute for Basic Standards, the Institute for Materials Research, the Institute for Applied Technology, the Center for Radiation Research, 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 an Office of Measurement Services and the following technical divisions:

> Applied Mathematics—Electricity—Metrology—Mechanics—Heat—Atomic and Molecular Physics—Radio Physics[2]—Radio Engineering[2]—Time and Frequency[2]—Astrophysics[2]—Cryogenics.[2]

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; develops, produces, and distributes standard reference materials; relates the physical and chemical properties of materials to their behavior and their interaction with their environments; and provides advisory and research services to other Government agencies. The Institute consists of an Office of Standard Reference Materials and the following divisions:

> Analytical Chemistry—Polymers—Metallurgy—Inorganic Materials—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 in the development of technological standards, and test methodologies; and provides advisory and research services for Federal, state, and local government agencies. The Institute consists of the following technical divisions and offices:

> Engineering Standards—Weights and Measures — Invention and Innovation — Vehicle Systems Research—Product Evaluation—Building Research—Instrument Shops—Measurement Engineering—Electronic Technology—Technical Analysis.

THE CENTER FOR RADIATION RESEARCH engages in research, measurement, and application of radiation to the solution of Bureau mission problems and the problems of other agencies and institutions. The Center consists of the following divisions:

> Reactor Radiation—Linac Radiation—Nuclear Radiation—Applied Radiation.

THE CENTER FOR COMPUTER SCIENCES AND TECHNOLOGY conducts research and provides technical services designed to aid Government agencies in the selection, acquisition, and effective use of automatic data processing equipment; and serves as the principal focus 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, and provides appropriate services to ensure that the NBS staff has optimum accessibility to the scientific information of the world. The Office consists of the following organizational units:

> Office of Standard Reference Data—Clearinghouse for Federal Scientific and Technical Information[3]—Office of Technical Information and Publications—Library—Office of Public Information—Office of International Relations.

---

[1] Headquarters and Laboratories at Gaithersburg, Maryland, unless otherwise noted; mailing address Washington, D.C. 20234.
[2] Located at Boulder, Colorado 80302.
[3] Located at 5285 Port Royal Road, Springfield, Virginia 22151.

UNITED STATES DEPARTMENT OF COMMERCE
Maurice H. Stans, Secretary
NATIONAL BUREAU OF STANDARDS • Lewis M. Branscomb, Director

NBS TECHNICAL NOTE 500

ISSUED APRIL 1970

Nat. Bur. Stand. (U.S.), Tech. Note 500, 50 pages (Apr. 1970)
CODEN: NBTNA

Edit-Insertion Programs for Automatic Typesetting of Computer Printout

Carla G. Messina and Joseph Hilsenrath
Office of Standard Reference Data
National Bureau of Standards
Washington, D.C. 20234

NBS Technical Notes are designed to supplement the Bureau's regular publications program. They provide a means for making available scientific data that are of transient or limited interest. Technical Notes may be listed or referred to in the open literature.

For sale by the Superintendent of Documents, U.S. Government Printing Office, Washington, D.C., 20402
(Order by SD Catalog No. C13.46:500), Price 60 cents.

FOREWORD

This report describes one of a series of computer programs being developed by the Data Systems Design Group of the NBS Office of Standard Reference Data to assist the Data Centers affiliated with the National Standard Reference Data System. The text of this report was reproduced from a typescript prepared on a typewriter terminal connected to a time-shared computer system. The program listing was produced from a magnetic tape which was produced from one of the programs described in this report.

TABLE OF CONTENTS

APPENDIX I

APPENDIX II

Edit-Insertion Programs for Automatic Typesetting of Computer Printout

by

Carla G. Messina and Joseph Hilsenrath

SETLST and KWIND are FORTRAN programs which accept a card deck or Fortran records on magnetic tape and insert the appropriate flags and shift symbols required by many of the "standard" typesetting programs associated with phototypesetting devices. The programs are specialized to the particular application; the typesetting device and associated programs; and to the desired typeface, by means of control cards and substitution tables supplied at run time. Examples are shown of applications to program listings, KWIC indexes, and normal computer output. When the input is in tabular form, the program permits more sophisticated operations including rearrangement, removal of trailing blanks, typeface changes between columns, etc. These programs can handle any records which can be read by a FORTRAN READ statement under an "A" format control.

Key words: Applications, computers, computer-assisted typesetting, FORTRAN programs, KWIC index, phototypesetting, printing.

1. Introduction

The ease with which a computer is able to prepare a permuted title index has resulted in the proliferation of such indexes. Usually these indexes are produced by a photoreduction of the computer print-out. Often, the quality of the printed index leaves much to be desired. Even when extreme care is taken to see that the text is legible, the pages are not usually in the correct proportion for a standard size of printed page. Program listings are more often than not reproduced with marginal clarity.

A technique for automatic-typesetting of program listings and KWIC indexes enables one to produce a page with so called "graphic arts" quality. A suitably selected typeface and size and a correspondingly appropriate page depth (number of lines per page) offers additional opportunity for economy of space and money as well as improved readability.

A technique for automatic typesetting of tables direct from magnetic tapes was developed at NBS by W. R. Bozman in 1962 [1]. Since that time several books of data have been produced by this method. The production of each of these books entailed the preparation of special programs requiring the services of a programmer experienced in machine language programming and having detailed knowledge of the operation of the Linofilm machine.

A more general pioneering effort in computer-assisted typesetting was carried on at MIT under the leadership of Dr. Michael P. Barnett. While it is unfortunate that the programs produced by Barnett and coworkers have not been maintained in recent years, the results of that work - described too modestly as "experiments" - have been fully recorded [3].

In recent months the Data Systems Design group of the NBS Office of Standard Reference Data has addressed itself to the problem of preparing a series of general purpose programs for text preparation, editing and photocomposition. This report describes the two programs which will enable any computer user to prepare magnetic tapes for phototypesetting of program listings, of KWIC indexes, and other material normally run on a line printer.

The program SETLST is in the spirit of the pioneering work of Barnett [3]. In some respects it is less general than that of Barnett's TABPRINT, since we do not typeset column headings and rules, but rather rely on overlays (for these.) It is more flexible where character stream transformation is required in changing upper case characters to upper and lower case, and in substitution of Greek characters and special symbols for their designation in the text.

Unlike TABPRINT which produces output to drive a particular photounit, SETLST produces tapes which need to be run through a typesetting program before the material can be set. It is however a feature of SETLST that it can insert any flags or header as may be required and hence is not restricted to a particular typesetting program or a particular machine. The specific strings or headers or flags are supplied at run time.

The program KWIND uses the same subroutines as SETLST but is especially tailored for typesetting of KWIC indexes. It is a characteristic of KWIC indexes that they have a gutter in the middle of the page which the KWIND program recognizes in order to operate on each half separately. On either side of the gutter KWIC lines have one of five characteristics, the line is either set flush left, or flush right, or flush left and right with a gap in the middle, or completely full or completely blank. The KWIND program scans the line, determines which type of line it is, and proceeds in the following manner. The flush left line is set flush left, and the program goes on to the next line. The flush right line is set flush right and the program goes on to the next line. The third type of line requires fancier treatment. The left hand piece is set flush left exclusive of the trailing blanks, the rest of the line is reset flush right without film advance and ignoring the leading blanks, then the program goes on to the next line. The fourth type of line is justified to an appropriate pica width. In addition to the two main fields described above, KWIND allows the designation of two additional fields to carry an identification. As duplication of fields is allowed, it is possible to repeat a single identification segment on both the left and right side of the index.

2. Program Characteristics

The programs discussed here operate on a file consisting of a program deck, or a series of records on tape, or the output tape (print tape) of a KWIC index, to produce another tape in precisely the format required for phototypesetting systems at the Government Printing Office or on other comparable systems. The program - suitably instructed via control cards - inserts a sequence of flags or locators or format designators where needed. The program is specialized to the particular application; the typesetting device and associated programs; and to the desired typeface; by means of control cards and a substitution table supplied at run time.

The substitution table is required to provide capital letters where desired and to indicate the location of punctuation and other symbols, the distribution of which is not standard on keyboards or grids. Another important use of the substitution table is to insert instructions in the character stream to obtain characters (mathematical symbols, Greek letters, etc.) not on the primary grid.

In most cases a typeset page will be longer than the 60 or so lines on a normal computer listing, hence the headings or dates or footings or page numbers which often appear on each page of computer output are extraneous and must be deleted. This the program does in an interesting way.

A series of control cards are supplied which contain the exact contents of the lines that are to be deleted. There is one card for each type of line. Any line in the file whose first 80 characters match any of the control cards is automatically ignored. While this takes care of any number of lines of text which remain fixed from page to page, there is still the problem of ignoring lines which give the page number which will vary from page to page. For this purpose, a provision has been made for indicating which fields are to be ignored in making the match. Since information which varies from page to page (like a table or a page number) is in a fixed location, an "ignore" symbol in these positions will do the job. Which symbol is used as an "ignore" symbol is open to choice as it is specifically indicated on one of the control cards.

2.1 Modification for Improved Run Efficiency

The programs given in the following pages are written in as low level dialect of Fortran as possible to facilitate their use on machines of different manufacture and compilers of different vintage. As a consequence the programs may be somewhat less efficient than ones written without these restraints.

In particular, we have made it a practice to imbed arguments in CALL statements rather than placing them in labeled COMMON. Experience in running production jobs on the NBS compiler indicated that a 10% saving in run time was achieved by modifying the program by the introduction of a set of labeled common statements. These and other changes needed to take advantage of the features of the compiler on the NBS 1108 are given in Appendix II. The places where these changes are to be made are clearly marked on the program listings in Appendix I.

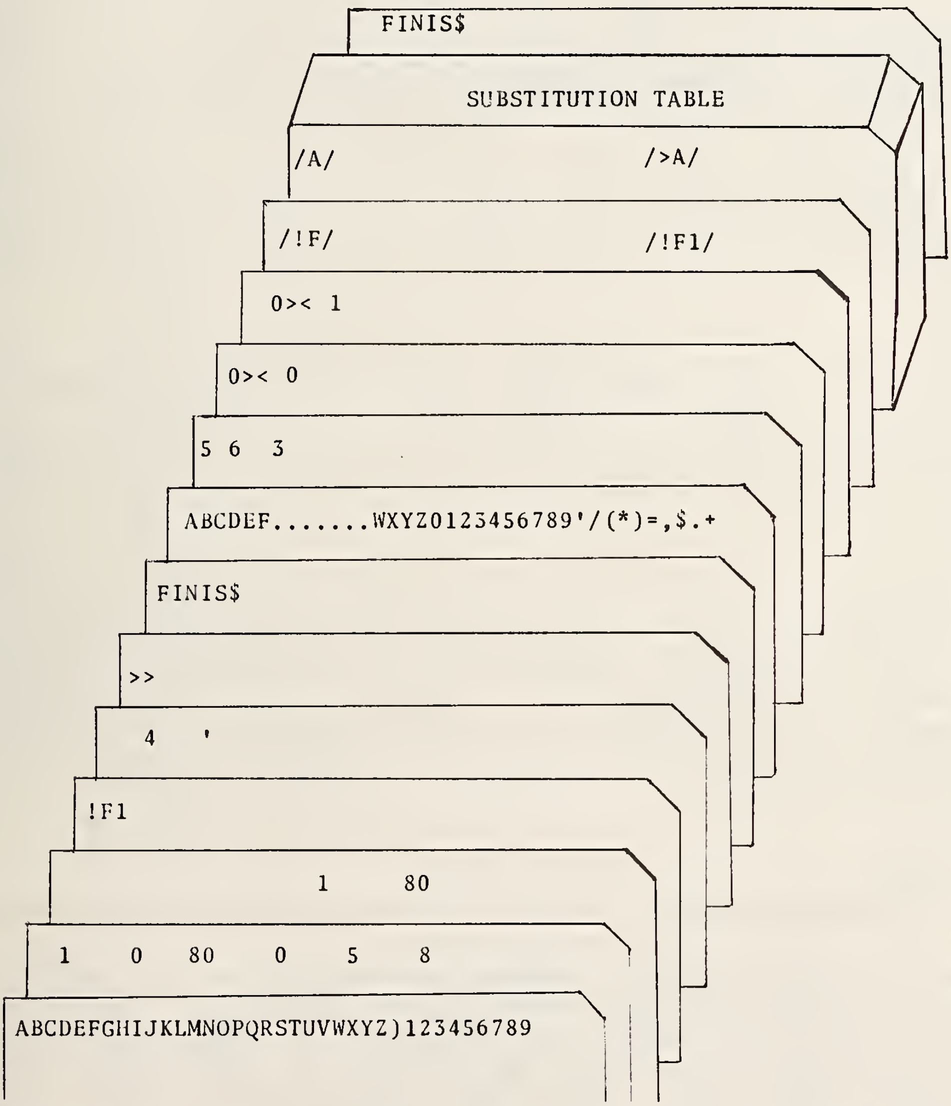


Figure 1a. The control cards and substitution table required by SETLST to format computer output for processing through the Mod I Autoset program at the Government Printing Office in order to utilize the monowidth typewriter grid shown in Figure 2.

```
ABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789 /                     \
   1     0    80     0     5     8
FIELD  1 GOES FROM   1 TO  80
THE FOLLOWING    1 CARDS CONTAIN THE INSERT STRINGS.

!F1
 0

SUBROUTINE AMATCH INPUT

  4  '
>>
FINIS$
ABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789'/(*)=,$.+
 5 6 3
   0>< 0
   0>< 1
/!F1/                                        /!F1/
/A/                                          />A/
/B/                                          />B/
/C/                                          />C/
/D/                                          />D/
 :                                             :
/X/                                          />X/
/Y/                                          />Y/
/Z/                                          />Z/
/+/                                          />8/
/=/                                          />9/
/*/                                          />0/
/// *                                        />//
/(/                                          />,/
/)/                                          />./
/@/                                          />2/
/#/                                          />1/
/&/                                          />-/
/:/                                          />;/
/!/                                          /// *
/?/                                          />?/
FINIS
```

Figure 1b. A printout supplied at the end of a SETLST run of the control cards shown in Figure 1a. Note that the substitution table which was entered in free-field format has been lined up for readability. The slash shown as string delimiters are uniformly supplied by the program regardless of what the actual string delimiter was on input. In particular, since the delimiter itself cannot be contained in the string it delimits, the control cards marked above with an asterisk had periods for delimiters.

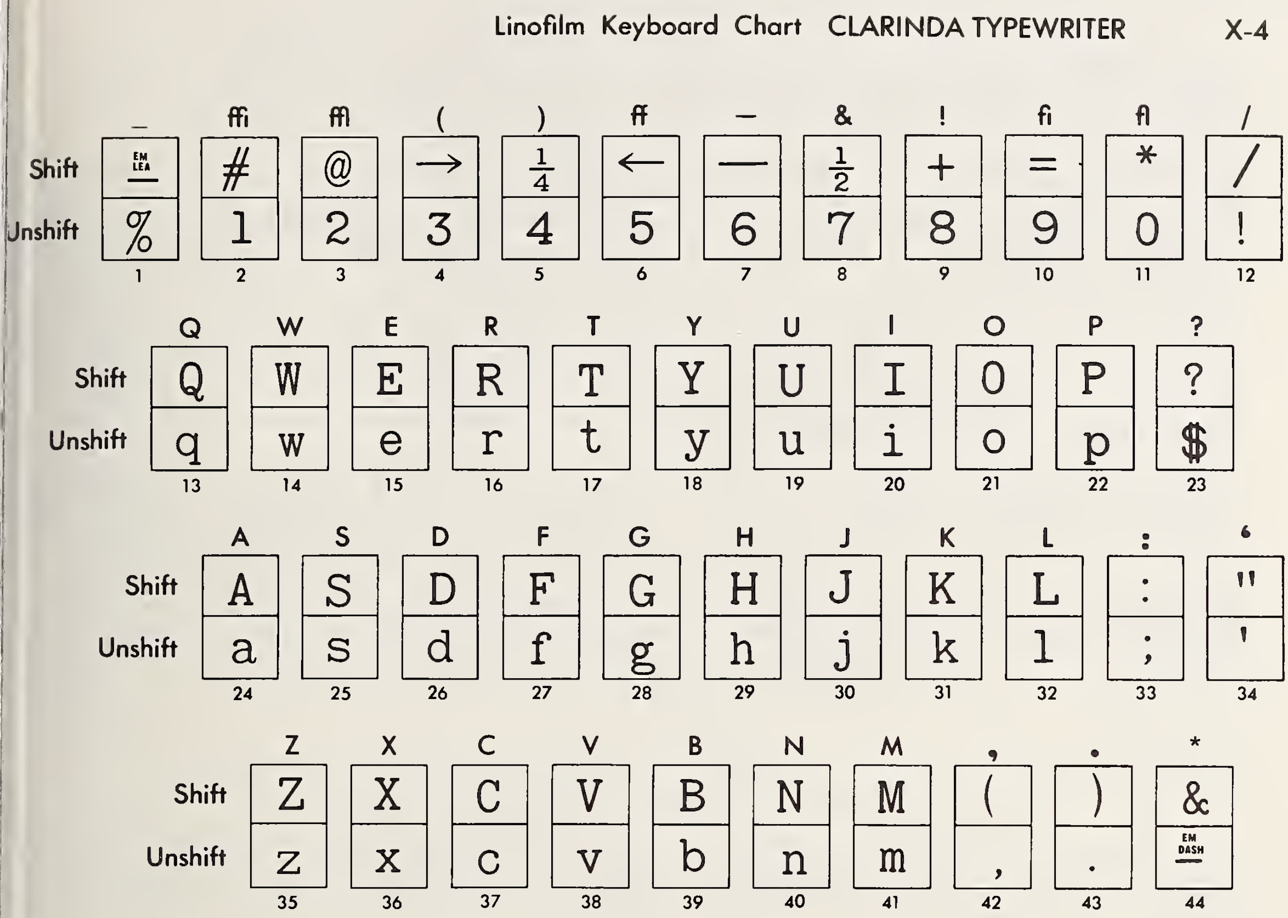


Figure 2. The layout of characters on the monowidth grid used to typeset the examples shown here. Note the connection between the layout of the graphics and the substitution table in the previous figure. Because other typefaces have different distributions of characters in the shift and unshift position, it is a great advantage to define the locations of the character via a substitution table rather than in the program proper. The advantage of this grid is that it contains all of the characters on a model 26 key punch. The use of most other grids for typesetting of computer listings require time consuming grid changes.

3. The Subroutines

The program SETLST makes use of two general purpose subroutines. The first of these, AMATCH, compares an input record to see if the front part matches one of a number of character strings supplied at run time. SETLST uses this to throw away unwanted lines from the input such as page number, page headings, etc.

The second, SUBSTITUTE, is used here primarily to insert shift symbols ahead of those characters which must appear in upper case. The input for this subroutine, which is supplied at run time via a substitute table, also specifies the precise characters which must be inserted in the text stream in place of such punctuation and graphic characters as, :, ?, /, =, +, -, etc. Which characters are substituted for the above symbols depend on the layout of a particular typesetting grid. For this reason the string substitutions are not built into the program. They are supplied on cards as part of the input. This makes the program applicable to systems other than those in use at the Government Printing Office. Other applications of SUBSTITUTE can be found in NBS Technical Note 470 [2]. Figure 1 shows the control cards and substitution table for the character layout on the Clarinda Typewriter grid in use at the Government Printing Office.

The subroutines PACK and N PRINT are used in this program to repack the characters, six to a computer word, and to write them on tape in records longer than the 132 characters normally permitted in FORTRAN. The repacking is necessary as the original files are read in and manipulated in A1 format. The subroutine N PRINT is required in order to write longer records than is possible under FORTRAN available on the NBS machine. In this assembly the record length (NCOUT) is specified as 300 characters. In normal usage this figure should be set to coincide with the size of the input buffer of the typesetting program for which the output tape is being prepared.

The Control Cards

4.1 Control Cards for SETLST and AMATCH

The first control card contains the alphabet in order starting in card column one followed by the integers in increasing order. In card column 50 is the end of line symbol required by the typesetting program. The program logic makes use of the location of the characters on the first control card in such a way as to avoid entirely the need to know how a particular machine recognizes a character on a card, what the internal bit representation of that character is, and where that character is placed in a machine word. Nor is it dependent on whether a single character is stored left-adjusted, right-adjusted or any other way. In this way the program is independent of whether the particular machine stores away 3 characters per machine word, or 6, or any number. If a typesetting program requires a particular character as a halt signal, that character must be punched into card column 50. The program ends the tape with two characters - a blank and the character designated in card column 50. Pains are taken to ensure that these two characters appear in the same record.

The second control card has six switches, each switch takes up five card fields and must be right adjusted in the field. The first switch contains the number of fields to process (greater than zero). The second switch should be set to -1 when only the printer will have the output, to zero when the printer and a tape will have the output, and to 1 when only the tape will contain the output. The third switch contains the length in characters of the input records (80 if from cards, and up to 132 if from tape records.) The fourth switch should be zero when all fields and their locators are to be put out regardless of whether or not the field is completely blank. It should be set to one when blank fields are to be ignored completely. This provision alleviates the flashing of unnecessary blanks. The fifth switch is the unit number from which to read the records to be typeset. The sixth switch is the number of the tape unit on which to write the output.

The third control card contains two numbers for each field denoted by the first switch. These numbers must be separated by at least one space. They indicate the starting and ending locations of the data to be extracted from the input records.

The next group of control cards contain the typesetting flags to be inserted ahead of the segments of the input. The flags are terminated by a blank. There must be as many flags as data fields (segments specified in switch one.)

The fifth type of control card carries the header information in (I2,78A1) format as required by some of the typesetting programs. A blank card must be inserted for those programs which do not require a header.

The next set of controls are needed for the subroutine AMATCH. The first card contains 2 characters in free-field form. The first of these is the character used to terminate the "match strings" discussed below. The next character is treated as a universal character when found in the "match string." It is used to cope with variable pieces of an otherwise fixed context, such as a page number in a heading line. The "match" strings are punched starting in column 1 and extend up to the string deliminator or the end of the card. The control cards for AMATCH are terminated by a card carrying FINIS followed by the character used to terminate the "match" strings.

The last set of control cards following AMATCH carry the instructions for the SUBSTITUTE subroutine. They are described in the next section.

4.2 The Control Cards for SUBSTITUTE

The <u>first</u> control card serves to define the punch configuration for the characters in the text as well as the control characters upon which the operations depend. The presence of the characters on the first card obviates the need to define them explicitly in the program. This simple device makes the program independent of a variety of incompatabilities which are such a source of trouble in adapting programs to different computers.

The program logic uses the disposition of the characters on the first control card in such a way as to avoid entirely the need to know how a particular machine recognizes a character on a card, what the internal bit representation of that character is, and where that character is placed in a machine word. In this way the program is independent of whether the particular machine stores away 3 characters per machine word, or 6, or even 7. Nor is it dependent on whether a single character is stored left-adjusted, right-adjusted or any other way. The alphabet is punched in order into the first 26 card columns hereinafter referred to as cc, and the digits 0,1,..., 9 follow in cc27 through 36. The character to be used to delineate the strings in the output of this program is designated in cc38; while cc47 must be left blank in this program and in all programs in this series.

The <u>second</u> card contains three switches in FORMAT (3I2). They serve no purpose here but <u>must be</u> present nevertheless.

The <u>third</u> and <u>fourth</u> control cards in FORMAT (4X, 2A1, I2), specify the format of the <u>input</u> and <u>output</u> records, respectively. The three items on each card perform the following tasks:

a. The first two items designate the characters used for case-shift lock and case-shift unlock. Their use is required only under circumstances described below.

b. The third item instructs the program to insert on input and delete on output, the case-shift symbols designated in item a. If this number is set to zero, the option is bypassed, in which case the first and second items discussed in item a above may be left blank. If this item is a non-zero integer, it distributes, when present on the third control card, and deletes, if present on the fourth control card, the shift case symbols indicated by the two previous items on the control cards.

Immediately following the fourth control card for substitute is a deck of cards containing the instructions for the string substitutions. In this version of the program, each card carries two strings--the original one and its substitute. The length of the strings this program handles is limited to a total of 76 for the string and its substitute. Thus a "long" string can be replaced by a "short" one and vice versa. Replacement of "long strings" by "long strings" can often be achieved by breaking them up into pieces and substituting piecemeal.

Each of the strings is delimited by a balanced character which is read from the first column of the substitution card. In this way each card can have its own string delimiter. The only requirement is that the delimiter character must not be one which is in the string it delimits. See Figure 1 for a sample set of control cards for this program.

The substitution table must be followed by a card with the word FINIS starting in cc1. It may be followed by the text to be manipulated if the input is from a card reader.

A number of text editing systems reserve one character as a precedence symbol to indicate an upper case letter. Thus if we punch *WASHINGTON we would expect a suitable printer to print out Washington. A single symbol could be used to print the word in all caps if one were prepared to type *W*A*S*H*I*N*G*T*O*N. This is obviously too time consuming as well as wasteful of valuable computer space. The problem is easily solved by reserving another symbol such as an apostrophe to indicate shift lock and shift unlock. In that case our test word would be keyboarded as follows: 'WASHINGTON'.

Subsequent transformation of these symbols as would be required in going to automatic typesetting or converting from the BCD representation to EBDIC would have to treat the character following the W differently in the strings 'WASHINGTON' and *WASHINGTON. This problem is solved by SUBSTITUTE in the following way. When instructed to do so via the third control card, the program changes 'WASHINGTON' to 'W'A'S'H'I'N'G'T'O'N. If instructed to do so via the fourth control card and after carrying out the substitution, the interior shift symbols are deleted and the word is imbedded between the shift and lock symbol and unlock symbol.

4.3 Control Cards for KWIND

The first control card for this program is identical to the first card in SETLST and serves the same purpose.

On control card two, switches 1,2,3, and 4 perform the same functions respectively as switches 2,3,5, and 6 of control card two in SETLST.

The third control card defines the fields into which the input line is broken. The order in which the fields are defined is important as the first and fourth fields are taken to be identifiers and the middle two as the index information. Suitable use of 0,0 on this control card provides for omission of one or more of the four fields. Thus on a system which cannot handle the full width of a KWIC index, it is possible to break the job into two portions. The left half can be run through first and the second half on a succeeding run. The two halves can then be pasted together using the duplicate identification numbers for alignment. Figure 9 was produced in this manner on a photounit which permitted a maximum width of only 43 picas. This method may be troublesome if the photounit does not advance the film uniformly.

The next 8 cards serve to define the manner in which line segments will be set as outlined in the introductory remarks on KWIND. Each card carries an arbitrary string of characters which is required by either the photounit or its associated typesetting program to achieve a flush left line, a flush right, a justified line, etc. The order of the cards is important as the program performs different operations after inserting different flags.

The <u>first</u> and the <u>eighth</u> cards must carry the flag (a locator) for the first and the <u>fourth field</u> (in <u>this</u> case an identifier). How this field is set is open to control by the typesetting program external to this program. The remaining cards carry the flags required to achieve the following results:

the <u>2nd card</u> carries a flag for quad left setting of the first half of a line in the <u>second field</u>.

the <u>3rd card</u> carries a flag for quad right setting of the second half of a line in the <u>second field</u>.

the <u>4th card</u> carries a flag for setting a justified full line in the second field.

the <u>5th card</u> carries the flag for the quad left setting of the first half of the <u>third field</u>.

the <u>6th card</u> carries the flag for quad right setting of the second half of the third <u>field</u>.

the <u>7th card</u> carries the flag for setting a justified full line in third field.

The above flags are considered terminated by a blank. All of the eight cards must be present. A blank card is treated as a string of zero length. It will affect the result only in that no flag will be inserted. The last control card, and those which follow are the same as in SETLST.

5. Applications

Applications of these programs fall into three main classes. The first and most straightforward is where we wish the final product to be a facsimile of the page produced on a line printer. Computer listings and results of report generators fall in this category. These applications require the duplication of the results, line-for-line and character-for-character, as they appear on the line-printer. The typesetting of such material requires the use of a monowidth typeface as is ordinarily found on a typewriter.

In the second application, we wish to improve the readability of the output by an appropriate transformation of the alphabetic characters to read as if they were originally entered in upper and lower case. Straight text, KWIC indexes, and bibliographies are examples of material which benefit from such treatment. In this application it is often not necessary to restrict the final output to a monowidth typewriter face as the program has ample provision for lining up the output in columns as required. Thus a fancier typeface can be used.

In the third application we can include those cases where material needs to be highlighted through the use of italics or boldfaced characters or even special characters like mathematical symbols or Greek letters. Such applications may require the use of more than one grid on the phototypesetting unit.

```
COMPND          1  3-*BENZYL-2,6-DIPHENYL-2/H-THIOPYRAN-5-              BPTPCA
COMPND          2  CARBOXALDEHYDE                                       BPTPCA
AUTHOR          1  MAZHAR-UL-HAQUE, C.N.CAUGHLAN                        BPTPCA
JRNL            1  CHEM.COMMUNIC                           34 1967 066  BPTPCA
CDFRML          1  C25H2001S1                                           BPTPCA
CRYST1 11.004   11.062   16.855           96.5           P 21/N         BPTPCA
COMPND          1  2,6-*DIMETHYLBENZOIC ACID                            DMBNZA10
AUTHOR          1  R.ANCA, S.MARTINEZ-CARRERA, S.GARCIA-BLANCO          DMBNZA10
JRNL            1  ACTA CRYST.                       23  1010 1967 001  DMBNZA10
CDFRML          1  (C1H3)2 C3H3C10101H1                                 DMBNZA10
CRYST1 15.24    4.04     13.16            94*08        P 21/A           DMBNZA10
CRYST2  0.01    0.01      0.01                                          DMBNZA10
CRYST3   4 1.227 1.21                                                   DMBNZA10
COMPND          1  2 ALPHA-*BROMOARBORINONE                             BRARB010
AUTHOR          1  O.KENNARD, L.RIVA SDI SANSEVERINO, J.S.ROLLETT        BRARB010
JRNL            1  TETRAHEDRON                       23   131 1967  16  BRARB010
CDFRML          1  C30H47BR101                                          BRARB010
CRYST1 12.84    8.68     22.46                         P 21 21 21       BRARB010
CRYST2  0.01     0.01     0.04                              219.0       BRARB010
CRYST3   4 1.34  1.39                                                   BRARB010
COMPND          1  (*GLYCYL-/L-HISTIDINATO) COPPER(II) SESQUIHYDRATE    CUGLHI10
AUTHOR          1  J.F.BLOUNT, K.A.FRASER, H.C.FREEMAN, J.T.SZYMANSKI,  CUGLHI10
AUTHOR          2  C.-H.WANG                                            CUGLHI10
JRNL            1  ACTA CRYST.                       22   396 1967 001  CUGLHI10
CDFRML          1  C8H10CU103N4/ 1.5H201                                CUGLHI10
CRYST1 11.24             17.84                         P 43 21 2        CUGLHI10
CRYST2  0.02              0.04                                          CUGLHI10
CRYST3   8 1.772 1.72  0.02                                             CUGLHI10
COMPND          1  *MONOAQUO(BETA-ALANYL-/L-HISTIDINATO) COPPER(II)     ALHICU10
COMPND          2  MONOHYDRATE                                          ALHICU10
```

Figure 3. A portion of a punched card data file set in a monowidth typeface by SETLST using the control cards shown in Figure 1.

5.1 Applications to Program Listings

Program listings represent a class of applications where it is important to reproduce the material exactly as it appears on punched cards, or on the print tape which drives a line printer, or on magnetic tape records. A monowidth character set resembling the type on a typewriter such as is shown in Figure 2 is required here.

In this application we set switch 1 and 3 of the second control card to 1 and 80 respectively to define the entire card as a single field; designate the three characters !F1 to be inserted at the beginning of each line in the output; instruct the program via the substitution table to capitalize each letter of the alphabet and on which keys to find the graphics +, -, =, (,) etc. Figures 3, 4, and 5 show the variety of material which was set from a deck of cards and the control cards shown in Figure 1.

```
         CALL NUMFND(SCAN,A,80,START,GOOF)
         IF((A.GE.79).OR.(GOOF.LE.0)) CALL EXIT
         CALL NUMFND(SCAN,A,80,STOP,GOOF)
         IF(GOOF.LE.0) CALL EXIT
         READ(5,1) (SCAN(A), A=1,80)
         DO 7 A=1,72
         IF(SCAN(A).NE.BLANK) GO TO 8
   7     CONTINUE
   9     WRITE(6,10)
   10    FORMAT('ONO OR POOR SKIPCC CARD. RUN ABORTED.')
         CALL EXIT
   8     DO 11 B=1,6
         IF(SCAN(A).NE.CHK(B)) GO TO 9
   11    A=A+1
         DO 12 B=A,80
         IF(SCAN(B).NE.BLANK) GO TO 13
   12    CONTINUE
         GO TO 14
   13    SKIP=SCAN(B)
   14    READ(5,1) (SCAN(A),A=1,80)
   19    READ(7,1,END=20) (TAPE(A), A=1,80)
         DO 16 B=START,STOP
         IF(SCAN(B).EQ.SKIP) GO TO 16
         IF(TAPE(B).NE.SCAN(B)) GO TO 19
   16    CONTINUE
         WRITE(6,17) (TAPE(A), A=1,80)
   17    FORMAT(17X,80A1)
         GO TO 19
   18    CALL EXIT
   20    D=D+1
         GO TO 15
         END
```

Figure 4. A portion of a program listing phototypset at the Government Printing Office from a tape produced by SETLST from the original program deck, using the control cards shown in Figure 1.

5.2 Applications to KWIC Indexes

In order to improve the readability of a KWIC index, it is necessary to do more than set it in a fancy typeface. It is necessary to transform the alphabetic characters to appear in upper and lower case. This is easily accomplished in KWIND via a more extensive substitution table. A typical end result is seen in Figure 6 where most of the words appear in initial caps while articles, preposition, etc., are set in lower case and certain words like USA are set in all caps.

The original format of the index is shown in Figure 7 and the substitution table that accomplished this transformation is shown in Figure 8. The improved readability of Figure 6 results more from the conversion to upper and lower case than it does from the variable width typeface, as can be seen from Figure 9 which is set in Times Roman type but in all caps.

5.3 Applications Requiring Multiple Grids

Thus far the applications stressed the use of but a single grid. Both SETLST and KWIND can handle as many grids as are required or as are permitted by the typesetting program for which the tape is being prepared.

In a number of existing abstract journals (Nuclear Science Abstracts or U.S. Government Research and Development Reports), Greek letters are spelled out and superior letters or figures are preceded by the word "exp." If the tape which produced these publications were run through SETLST it would be possible to replace the word ALPHA or Alpha by a call (=G3, for example) to the appropriate grid which contained the Greek alphabet. Similarly it is possible to call for superior and inferior characters from an appropriate grid.

DETAILS FOR TABLE 1 — SINGLE PRECISION (8 DIGITS)

OMNITAB, USING ORTHO SUBROUTINE			7094	EXAMPLE 13
BETA-HAT (Y1)	COUNT	BETA-HAT (Y2)		COUNT
1.0012817	2.892	.99999949		6.292
.99780273	2.658	.10000013		5.886
.99932861	3.173	.0099999756		5.613
1.0001755	3.756	.0010000018		5.745
.99998569	4.844	.000099999866		5.873
1.0000004	6.398	.000010000004		6.398
AVERAGE =	3.954		AVERAGE =	5.968

OMNITAB, USING ORTHO SUBROUTINE			1108	EXAMPLE 14
BETA-HAT (Y1)	COUNT	BETA-HAT (Y2)		COUNT
1.0064697	2.189	.99999990		7.000
.99902344	3.010	.099999700		5.523
.99975586	3.612	.010000125		4.903
.99996948	4.515	.00099998200		4.745
1.0000100	5.000	.00010000109		4.963
.99999968	6.495	.0000099999778		5.654
AVERAGE =	4.137		AVERAGE =	5.464

ORTHO, WITH RE-ORTHOGONALIZATION OMITTED			1108	EXAMPLE 15
BETA-HAT (Y1)	COUNT	BETA-HAT (Y2)		COUNT
-1216.5426	-3.085	.98419483		1.801
2752.0557	-3.439	.13523918		.453
-1057.0931	-3.025	-.0034660707		-.129
146.97336	-2.164	.0028495983		-.267
-7.3080225	-.919	-.0000049256487		-.021
1.1663037	.779	.000012094996		.679
AVERAGE =	-1.976		AVERAGE =	.419

Figure 5. A typeset version of a formatted computer output. See Figure 14 for the same material photographed from the Computer output.

Thermal Conductance Factors for Preformed	Above-Deck Roof Insulation (1955)	USC	R257
Grading Of	Abrasive Grain for Grinding Wheels (1965)	USC	CS271
Grading Of	Abrasive Grain On Coated Abrasive Products (1967)	USC	PS8
Coated	Abrasive Products (1955)	USC	R89
Grading of Abrasive Grain On Coated	Abrasive Products (1967)	USC	PS8
(1965) Acrylonitrile Butadiene Styrene	ABS Plastic Drain, Waste and Vent Pipe and Fittings	USC	CS270
Acrylonitrile Butadiene Styrene	ABS Plastic Pipe (SDR PR and Class T) (1963)	USC	CS254
Rigid	ABS Plastic Pipe, IPS Dimensions (1959)	USC	CS218
Colors for Kitchen	Accessories (1938)	USC	CS62
Colors for Bathroom	Accessories (1938)	USC	CS63
Stove Pipe And	Accessories (1942)	USC	R190
Metal Lath Expanded and Sheet and Metal Plastering	Accessories (1960)	USC	R3
	Accoustical Materials (1960)	USC	R262
Solvent Welded Swp Size Cellulose	Acetate Butyrate Pipe (1957)	USC	CS206
Girls, and Boys Knit Underwear Exclusive of Rayon,	Acetate, and Nylon (1955) Infants, Childrens,	USC	CS198
waste and Vent Pipe and Fittings (1965)	Acrylonitrile Butadiene Styrene ABS Plastic Drain,	USC	CS270
PR and Class T) (1963)	Acrylonitrile Butadiene Styrene ABS Plastic Pipe (SDR	USC	CS254
Hollow Metal Single	Acting Swing Doors, Frames and Trim (1928)	USC	R82
	Adhesive Plaster (1952)	USC	R85
Water Resistant Organic	Adhesives for Installation of Clay Tile (1952)	USC	CS181
Polystyrene Plastic Wall Tiles, And	Adhesives for Their Application (1950)	USC	CS168
Chasers for Self Opening And	Adjustable Die Heads (1929)	USC	R51
Hot Dipped Galvanized Ware Coated	After Fabrication (1959)	USC	CS161
Wire Diameters for Mineral	Aggregate Production Screens (1942)	USC	R147
Coarse	Aggregates, Crushed Stone, Gravel, and Slag (1948)	USC	R163
Polyethylene Sheeting Construction, Industrial And	Agricultural Application (1961)	USC	CS238
(1942)	Agricultural Insecticide and Fungicide Packages	USC	R41

Figure 6. A portion of a KWIC index produced by SETLST from records shown in Figure 7. See Figure 8 for an explanation of circled mistakes.

```
5                                    U. S. DEPT. OF COMMERCE STANDARDS                                          5
1ABOVE-DECK                                      TITLE WORD INDEX                                            BARS

       THERMAL CONDUCTANCE FACTORS FOR PREFORMED ABOVE-DECK ROOF INSULATION (1955)                          USC R257
                                      GRADING OF ABRASIVE GRAIN FOR GRINDING WHEELS (1965)                  USC CS271
                                      GRADING OF ABRASIVE GRAIN ON COATED ABRASIVE PRODUCTS (1967)          USC PS8
                                          COATED ABRASIVE PRODUCTS (1955)                                   USC R89
             GRADING OF ABRASIVE GRAIN ON COATED ABRASIVE PRODUCTS (1967)                                   USC PS8
(1965)        ACRYLONITRILE BUTADIENE STYRENE ABS PLASTIC DRAIN, WASTE AND VENT PIPE AND FITTINGS           USC CS270
              ACRYLONITRILE BUTADIENE STYRENE ABS PLASTIC PIPE (SDR PR AND CLASS T) (1963)                  USC CS254
                                           RIGID ABS PLASTIC PIPE, IPS DIMENSIONS (1959)                    USC CS218
                              COLORS FOR KITCHEN ACCESSORIES (1938)                                         USC CS62
                             COLORS FOR BATHROOM ACCFSSORIES (1938)                                         USC CS63
                                  STOVE PIPE AND ACCESSORIES (1942)                                         USC R190
  METAL LATH EXPANDED AND SHEET AND METAL PLASTERING ACCESSORIES (1960)                                     USC R3
                                                 ACCOUSTICAL MATERIALS (1960)                               USC R262
                  SOLVENT WELDED SWP SIZE CELLULOSE ACETATE BUTYRATE PIPE (1957)                            USC CS206
  GIRLS, AND BOYS KNIT UNDERWEAR EXCLUSIVE OF RAYON, ACETATE, AND NYLON (1955)      INFANTS, CHILDRENS,     USC CS198
WASTE AND VENT PIPE AND FITTINGS (1965)          ACRYLONITRILE BUTADIENE STYRENE ABS PLASTIC DRAIN,         USC CS270
 PR AND CLASS T) (1963)                          ACRYLONITRILE BUTADIENE STYRENE ABS PLASTIC PIPE (SDR      USC CS254
                             HOLLOW METAL SINGLE ACTING SWING DOORS, FRAMES AND TRIM (1928)                 USC R82
                                                 ADHESIVE PLASTER (1952)                                    USC R85
                         WATER RESISTANT ORGANIC ADHESIVES FOR INSTALLATION OF CLAY TILE (1952)             USC CS181
             POLYSTYRENE PLASTIC WALL TILES, AND ADHESIVES FOR THEIR APPLICATION (1950)                     USC CS168
                    CHASERS FOR SELF OPENING AND ADJUSTABLE DIE HEADS (1929)                                USC R51
               HOT DIPPED GALVANIZED WARE COATED AFTER FABRICATION (1959)                                   USC CS161
                      WIRE DIAMETERS FOR MINERAL AGGREGATE PRODUCTION SCREENS (1942)                        USC R147
                                          COARSE AGGREGATES, CRUSHED STONE, GRAVEL, AND SLAG (1948)         USC R163
  POLYETHYLENE SHEETING CONSTRUCTION, INDUSTRIAL AND AGRICULTURAL APPLICATION (1961)                        USC CS238
(1942)                                           AGRICULTURAL INSECTICIDE AND FUNGICIDE PACKAGES            USC R41
                              PACKAGING OF FIRST AID UNIT DRESSINGS AND TREATMENTS (1941)                   USC R178
                                    PACKAGING OF AIR BRAKE (ELECTRIC RAILWAY) PARTS (1935)                  USC R162
                                   TANK MOUNTED AIR COMPRESSORS (ONE-FOURTH TO 10 HORSEPOWER (1948)         USC R202
                                   TANK MOUNTED AIR COMPRESSORS (1956)                                      USC CS126
```

Figure 7. A portion of the input which produced the result shown in the previous figure. The programs SETLST and KWIND have facile provisions for discarding header lines such as appear in the first two lines above. See Figure 8 for the control cards and the substitution table which was used to achieve the transformation.

Figure 8.

```
ABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789
    0  120     7    8
 0 0    2 53    54 107     108 116
!I01
!I02
!I03
!I04
!I05
!I06
!I07
!I08

$ '
''''''''                                       U. S. DEPT. OF COMMERCE STAND$
''''''''''''''''''''''''                                TITLE WORD INDEX $

FINIS$
ABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789 /
 0 5 3
    →← 0
    →← 1
/(SRD/ /(→S→D→R/
/PVC/ /(→P→V→C/
/(PE)/ /(→P→E)/
/ PR)/ / →P→R)/
/ PR / / →P→R /
/ SDR/ / →S→D→R/
/ DWV/ / →D→W→V/
/ IPS/ / →I→P→S/
/ ABS / / →A→B→S /
/ USA/ / →U→S→A/
/ AND / / AND →/
/ OR / / OR →/
/ FOR / / FOR →/
/ TO / / TO →/
/ OF / / OF →/
/ WITH / / WITH →/
/ II/ / →I→I/
/ III/ / →I→I→I/
/ IV/ / →I→V/
/ VI/ / →V→I/
/ VII/ / →V→I→I/
/ VIII/ / →V→I→I→I/
/ IX/ / →I→X/
/ XI/ / →X→I/
/ XII/ / →X→I→I/
/ XIII/ / →X→I→I→I/
/ XIV/ / →X→I→V/
/ XV/ / →X→V/
/ XVI/ / →X→V→I/
/ XVII/ / →X→V→I→I/
/ XVIII/ / →X→V→I→I→I/
/ XIX// →X→I→X/
/ XX/ / →X→X/
/!I03A/ /!I03→A/
/!I03B/ /!I03→B/
/!I03C/ /!I03→C/
/!I03D/ /!I03→D/
/!I03E/ /!I03→E/
/!I03X/ /!I03→X/
/!I03Y/ /!I03→Y/
/!I03Z/ /!I03→Z/
/!I06A/ /!I06→A/
/!I06B/ /!I06→B/
/!I06C/ /!I06→C/
/!I06D/ /!I06→D/
/!I06E/ /!I06→E/
/!I06X/ /!I06→X/
/!I06Y/ /!I06→Y/
/!I06Z/ /!I06→Z/
/'A/ /'→A/
/'B/ /'→B/
/'C/ /'→C/
/'D/ /'→D/
/'E/ /'→E/
/'X/ /'→X/
/'Y/ /'→Y/
/'Z/ /'→Z/
```

Continued

Figure 8 concluded

```
/'S / /'S →/
/ A/ / →A/
/ B/ / →B/
/ C/ / →C/
/ D/ / →D/
/ E/ / →E/
/ X/ / →X/
/ Y/ / →Y/
/ Z/ / →Z/
/(A/ /(→A/
/(B/ /(→B/
/(C/ /(→C/
/(D/ /(→D/
/(E/ /(→E/
/(X/ /(→X/
/(Y/ /(→Y/
/(Z/ /(→Z/
/-A/ /-→A/
/-B/ /-→B/
/-C/ /-→C/
/-D/ /-→D/
/-E/ /-→E/
/-X/ /-→X/
/-Y/ /-→Y/
/-Z/ /-→Z/
/.A/ /.→A/
/.B/ /.→B/
/.C/ /.→C/
/.D/ /.→D/
/.E/ /.→E/
/.W/ /.→W/
/.X/ /.→X/
/.Y/ /.→Y/
/.Z/ /.→Z/
```

```
/+A/ /!G2!T3←U!G1!T1→A/
/+B/ /!G2!T3←U!G1!T1→B/
/+C/ /!G2!T3←U!G1!T1→C/
/+D/ /!G2!T3←U!G1!T1→D/
/+E/ /!G2!T3←U!G1!T1→E/
/+X/ /!G2!T3←U!G1!T1→X/
/+Y/ /!G2!T3←U!G1!T1→Y/
/+Z/ /!G2!T3←U!G1!T1→Z/
/=A/ /!G2!T3→:!G1!T1→A/
/=B/ /!G2!T3→:!G1!T1→B/
/=C/ /!G2!T3→:!G1!T1→C/
/=D/ /!G2!T3→:!G1!T1→D/
/=E/ /!G2!T3→:!G1!T1→E/
/=X/ /!G2!T3→:!G1!T1→X/
/=Y/ /!G2!T3→:!G1!T1→Y/
/=Z/ /!G2!T3→:!G1!T1→Z/
/'S!/ /'S!/
/+/  /!G2!T3←U!G1!T1/
/=/ /!G2!T3→:!G1!T1/
FINIS
```

Figure 8. Control cards and substitution table for transforming the character stream in Figure 7 to that shown in Figure 6. Note that four sets of alphabet cards are needed to capitalize characters following: a space, a right parenthesis, a hyphen, and the locator !I03 inserted by the program at the middle gutter. The other substitution cards override these and achieve the desired exceptions. The mistakes in capitalization of the words: "and", and "of", immediately to the left of the middle gutter in Figure 6 result from the omission of two cards from the substitution table. These are /AND!/ /AND!/ and /OF!/ /OF!/.

In many computerized data files, space limitations force the use of a shorthand notation. These can be expanded by appropriate string substitutions. If the data are arranged in columns, the program allows for selective substitution rather than universal substitution and rearrangement and omissions of columns of numbers, or other alphanumeric characters.

A good example of the versatility of SETLST is afforded by the transformations shown in Figure 10 of an index produced earlier on a line printer in all caps for the publication "CINDA 68 An Index to the Literature on Microscopic Neutron Data." This index is published jointly by the Division of Technical Information Extension of the U.S. Atomic Energy Commission and the Organization for Economic Cooperation and Development.

For this application it is necessary to take advantage of a feature of SETLST which enables one to divide the record into as many as 20 pieces of arbitrary length and to insert any arbitrary flags between them. The first set of control cards shown in Figure 12 causes the program to break the record into 12 pieces and to insert flags !I01, etc., in the order indicated, in front of each of the pieces which are defined on the third control card. It is this expanded record upon which the subroutine SUBSTITUTE operates. It is the use of these locators in the substitution table which allows for selective substitution.

It is interesting to note that while the locators were inserted primarily to retain the columnar arrangement when going from fixed width to variable width characters, their presence in the text stream makes it possible to perform sophisticated conditional substitutions, which would otherwise have required a much more complicated and much less versatile program. A brief examination of the substitution table shown in Figure 12 should reveal the flexibility which this feature affords.

5909025	NONLINEAR INDUCTOR	MAGNETIC	AMPLIFIER CIRCUITS. PART 1. THE SERIES CIRCUIT WITH A		5909025
5209061		MSAC PULSE POWER	AMPLIFIER. TECHNICAL MEMORANDUM		5209061
6309189		COMPARISON OF HOT-ELECTRON AND RELATED	AMPLIFIERS		6309189
6623603		DEVELOPMENT ON COUPLING SCHEMES FOR PLASMA	AMPLIFIERS	EXPLORATORY	6623603
6009174		BANDWIDTH IN WIDE-BAND TRANSISTOR AND ESAKI DIODE	AMPLIFIERS	THEORETICAL LIMITATIONS OF GAIN AND	6009174
6209190		YSTEMS WITH APPLICATION TO COMMON-EMITTER TRANSISTOR	AMPLIFIERS	/RTION IN FREQUENCY-DEPENDENT TWO-PORT S	6209190
5109058		RISE TIME IN PULSE POWER	AMPLIFIERS ASSUMING IDEAL TRANSFORMERS		5109058
6309162	TECHNIQUE	THE DESIGN OF WIDEBAND TRANSISTOR	AMPLIFIERS BY AN EXTENSION OF THE SAMPLED-PARAMETER		6309162
6009171	TY	OPTIMAL DESIGN OF MULTI-STAGE TUNED-TRANSISTOR	AMPLIFIERS CONSIDERING GAIN, STABILITY, AND SENSITIVI		6009171
5909268		BINARY QUANTIZATION OF SIGNAL	AMPLITUDES. EFFECT FOR RADAR ANGULAR ACCURACY		5909268
6723853	EXCURSIONS OF ORBITING SPACECRAFT	USE OF	ANALOG COMPUTATION IN PREDICTING DYNAMIC TEMPERATURE		6723853
5309245		A SIMPLE DIGITAL TO	ANALOG CONVERTER		5309245
6009170		APPLICATION OF THE MAGNETORESISTANCE EFFECT TO	ANALOG MULTIPLICATION		6009170
5909094		AN	ANALOG MULTIPLIER USING THE FIELD-EFFECT TRANSISTOR		5909094
6309011		DEVELOPMENT OF AN	ANALOG MULTIPLIER, BASED ON THE HALL EFFECT		6309011
6623441		CONTROL OF REMOTE MANIPULATORS. EXPERIMENTS USING	ANALOG SIMULATION	MANUAL	6623441
6623224	COMPUTER DISPLAYS	INVESTIGATION OF AN	ANALOG TECHNIQUE TO DECREASE PEN-TRACKING TIME IN		6623224
6724048		THE ANALYSIS AND CALIBRATION OF	ANALOG TO DIGITAL ENCODERS		6724048
6523547		GRAPHICAL PROCESSING USING HYBRID	ANALOG-DIGITAL CIRCUITRY		6523547
6623541		TEN-BIT	ANALOG-TO-DIGITAL CONVERTER		6623541
5909152		STUDY OF SEMICONDUCTOR DEVICES BY	ANALOGUE TECHNIQUES		5909152
6209361		COMPUTING FACILITY MANAGEMENT SURVEY RESULTS AND	ANALYSIS		6209361
6423218		AUTOMATIC SYNTACTIC	ANALYSIS		6423218
6523471		FORMAT	ANALYSIS		6523471
6623841		MAU-MACHINE COMMUNICATION IN ON-LINE MATHEMATICAL	ANALYSIS		6623841
6623255		MAP. A SYSTEM FOR ON-LINE MATHEMATICAL	ANALYSIS		6623255
6623257		PREDICATION-TYPING - A PILOT STUDY IN SEMANTIC	ANALYSIS		6623257
5609116		2 ON PROGRESS IN MACHINE COMPUTATION AND NUMERICAL	ANALYSIS	REPORT NO	5609116
6523430		AN ON-LINE SYSTEM FOR ALGEBRAIC COMPUTATION AND	ANALYSIS	CALCULAID.	6523430
6623488		N TECHNIQUES TO TEACH ELECTRICAL ENGINEERING NETWORK	ANALYSIS	/ED LEARNING AND COMPUTER-BASED INSTRUCTIO	6623488
6724048	ENCODERS	THE	ANALYSIS AND CALIBRATION OF ANALOG TO DIGITAL		6724048

Figure 9. Results from a test of the KWIND program set in Times Roman on a Linofilm at the U.S. Government Printing Office. Letter spacing would improve the readability somewhat but not enough to justify the added expense of phototypsetting. Figure 6 shows how much improvement is achieved by converting to upper and lower case.

^{1}H	Total X-Sect	Expt Jour	3. 3	1.0 2	PR 76 1750	D/49	COL	MELKONIAN MOLECUL EFFECTS /HYDROCARB	39195
^{1}H	Total X-Sect	Theo Jour		6 6	PR 82 60	4/51	COR	SALPETER EFFECTIVE RANGES FROM SIGMA	39194
^{1}H	Elastic	Expt Jour	1 4 7		PR 75 1819	6/49	LAS	BARSCHALL FROM DEL ISOTROPY ASSUMED	39098
^{1}H	Elastic	Theo Jour	0 0	2.0 7	PR 76 38	7/49	COR	BETHE S-WAVE PHASESHIFT EFF RANGE TH	39099
^{1}H	Diff Elastic	Expt Jour	1 4 7		PR 75 1819	6/49	LAS	BARSCHALL RECOIL PROTONS ISOTR OBS	39097
^{1}H	Diff Elastic	Expt Abst	1 4 7		PR 87 223	7/52	CRC	CROSS 90TO180DEG C.M. FROM RECOIL PS	39192
^{1}H	ThrmlScatLaw	Theo Jour	THR		PR 60 18	7/41	GWU	SACHS TELLER CALC SEE H2.CH4.NH3.H2O	39171
^{1}H	ThrmlScatLaw	Theo Jour	−2	−1	PR 60 18	7/41	GWU	SACHS TELLER CALC SEE H2.CH4.NH3 H2O	39172
^{1}H	(n,γ)	Theo Jour		6. 6	PR 82 60	4/51	COR	SALPETER EFFECTIVE RANGES FROM SIGMA	39193
								OCT 1, 1967 PAGE 2 1	DEUTERIUM
ELEMENT	QUANTITY	TYPE	ENERGY		DOCUMENTATION		LAB	COMMENTS	SERIAL
			MIN	MAX	REF VOL PAGE	DATE			NO
^{2}D	Total X-Sect	Expt Prog	1.0 5	1.3 6	AECU-2128 43	5/52	MIT	(ES LI(P.M) SOURCE TRANSMISSION	39633
^{2}D	Diff Elastic	Eval Rept	1.0 5	1 4 7	NAA-SR-11980IV	4/67	AI	CAMPBELL+ LEG COEFS TBL+CURVS C-MSYS	38742
								OCT. 1, 1967 PAGE 3	1 TRITIUM
ELEMENT	QUANTITY	TYPE	ENERGY		DOCUMENTATION		LAB	COMMENTS	SERIAL
			MIN	MAX	REF VOL PAGE	DATE			NO
^{3}T	Diff Elastic	Eval Rept	1.0 5	1.2 7	NAA-SR-11980IV	4/67	AI	CAMPBELL+ LEG COEFS TBL+CURVS C-MSYS	38741
								OCT. 1, 1967 PAGE 4	2 HELIUM
ELEMENT	QUANTITY	TYPE	ENERGY		DOCUMENTATION		LAB	COMMENTS	SERIAL
			MIN	MAX	REF VOL PAGE	DATE			NO
He	Diff Elastic	Eval Rept	4.0 6	1.4 7	NAA-SR-11980IV	4/67	AI	CAMPBELL+ LEG COEFS TBL+CURVS C-MSYS	38739
^{3}He	Diff Elastic	Eval Rept	1.0 6	1.8 7	NAA-SR-11980IV	4/67	AI	CAMPBELL+ LEG COEFS TBL+CURVS C-MSYS	38740
^{3}He	(n, p)	Expt Jour			PR 79 929	9/50	LAS	JARVIS INV N ANG DIST SEVERAL E(P)	39190
^{3}He	(n, p)	Expt Jour			PR 90 865	6/53	ORL	WILLARD INV N ANG DIST SEVERAL E(P)	39191
^{4}He	Elastic	Theo Jour	2.5 6		PR 58 682	0/40	PTN	WHEELER SPIN-ORBIT EFFECT PR 58 590	39186
^{4}He	Elastic	Theo Jour	0	5. 6	PR 82 750	6/51	WIS	ADAIR 1LVL DISPERSION TH PHASESHIFTS	39187
^{4}He	Diff Elastic	Theo Jour	2.5 6		PR 58 682	0/40	PTN	WHEELER SPIN-ORBIT EFFECT PR 58 590	39188
^{4}He	Diff Elastic	Theo Jour	NDG		PR 58 829	N/40	STF	BLOCH NEW DESCR RESON IN DISPERSN TH	39185
^{4}He	Diff Elastic	Theo Jour	0	5. 6	PR 82 750	6/51	WIS	ADAIR 1LVL DISPERSION TH PHASESHIFTS	39189
								OCT. 1, 1967 PAGE 5	3 LITHIUM
ELEMENT	QUANTITY	TYPE	ENERGY		DOCUMENTATION		LAB	COMMENTS	SERIAL
			MIN	MAX	REF VOL PAGE	DATE			NO
Li	Diff Elastic	Eval Rept	2.6 5	1.4 7	NAA-SR-11980IV	4/67	AI	CAMPBELL+ LEG COEFS TBL+CURVS C-MSYS	38738
Li	Absorption	Expt Jour	THR		PR 83 641	3/51	ORL	POMERANCE PILE OSC IN REAC SHIELD	39692
^{6}Li	Evaluation	Eval Rept	1. −3	1.5 7	LA-3695	3/67	LAS	BATTAT+ SIG DATA IN ENDF/B FORMAT	38837
^{6}Li	Total X-Sect	Eval Rept	1. −3	1.5 7	LA-3695	3/67	LAS	BATTAT+ SIG DATA IN ENDF/B FORMAT	38835
^{6}Li	Elastic	Eval Rept	1. −3	1.5 7	LA-3695	3/67	LAS	BATTAT+ SIG DATA IN ENDF/B FORMAT	38834
^{6}Li	Diff Elastic	Eval Rept	2.1 5	1.4 7	NAA-SR-11980IV	4/67	AI	CAMPBELL+ LEG COEFS TBL+CURVS C-MSYS	38737
^{6}Li	Nonelastic	Eval Rept	1. −3	1.5 7	LA-3695	3/67	LAS	BATTAT+ SIG DATA IN ENDF/B FORMAT	38833
^{6}Li	Tot Inelastic	Eval Rept	4.2 6	1.5 7	LA-3695	3/67	LAS	BATTAT+ SIG DATA IN ENDF/B FORMAT	38832
^{6}Li	(n,γ)	Eval Rept	1. −3	1.5 7	LA-3695	3/67	LAS	BATTAT+ SIG DATA IN ENDF/B FORMAT	38830
^{6}Li	(n, p)	Eval Rept	3. 6	1.5 7	LA-3695	3/67	LAS	BATTAT+ SIG DATA IN ENDF/B FORMAT	38829
^{6}Li	(n,α)	Eval Rept	3.2 6	1.5 7	LA-3695	3/67	LAS	BATTAT+ SIG DATA IN ENDF/B FORMAT	38828
^{6}Li	(n, nα)	Eval Rept	1.7 6	1.5 7	LA-3695	3/67	LAS	BATTAT+ NNA+N2NA DATA IN ENDF/B FORM	38831
^{7}Li	Evaluation	Eval Rept	1. −3	1.5 7	LA-3695	3/67	LAS	BATTAT+ SIG DATA IN ENDF/B FORMAT	38836
^{7}Li	Total X-Sect	Eval Rept	1. −3	1.5 7	LA-3695	3/67	LAS	BATTAT+ SIG DATA IN ENDF/B FORMAT	38827
^{7}Li	Elastic	Eval Rept	1. −3	9.9 6	LA-3695	3/67	LAS	BATTAT+ SIG DATA IN ENDF/B FORMAT	38826

Figure 10. A table containing superscripts, subscripts, and Greek characters, produced by SETLST from a print tape that produced the output shown in the next figure. This table requires use of more than one grid. Note that the isotope number was put to the left of the element symbol and made a superior figure. Note also the substitution of Greek letters for the words alpha, gamma, etc. See Figure 12 for the control cards and substitution table used here.

Among the more interesting transformations performed by the control cards shown in Figure 12 are the following:

a. The flag !I02 in front of the isotopic number (the second value in the original line) caused the numbers following it to be set as superior figures and flush-right in front of, instead of behind, the element symbol. Two cards in the substitution table served to delete the leading zeros in front of the isotopic number.

b. The columns flagged !I03 required quite special treatment. For example, had the word GAMMA appeared elsewhere it would have been simply set with all caps. Here, however, GAMMA is replaced by a character stream which brings in grid two so as to produce the Greek letter Γ . Other items in the substitute table transformed N, PROTON to (n,p) and N, ALPHA to (n, ∇), etc.

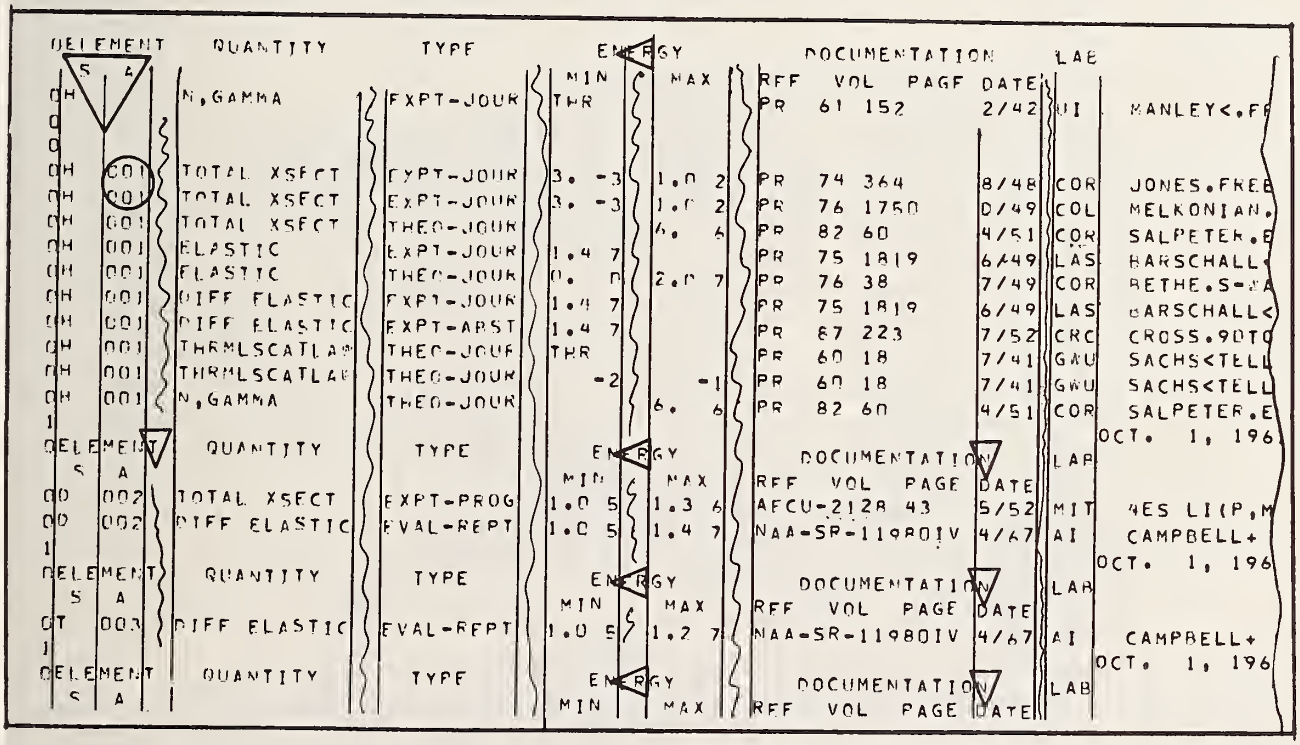
c. If a relatively few lines of the file or relatively few words have text extending across the boundaries defined on the third control card, the substitution table can be used to reattach the pieces in the two columns and even move them to another location. The following is an example of such a manipulation: /!I05 EN!I06GY/ /!I13 > E > N > E>R>G>Y/ where the lost characters ER are reinserted and the heading ENERGY is shifted to a new location.

d. The inclusion in the substitution table of /!I02 A/ // and /!I01 S/ /I15/ removes from the original records the heading S A and inserts the new line locator !I15.

e. While ordinarily the user need not concern himself with the fine points of how the program operates, there is one feature which is important to emphasize.

That program deletes trailing blanks in the defined fields when inserting the locators. For that reason the instruction to delete the N from field !I08 in Figures 11 and 13 is written as:

/!I08N!I09/ /!I09/ instead of /!I08N / //



```
0ELEMENT   QUANTITY       TYPE          ENERGY          DOCUMENTATION        LAB
  S   A                              MIN      MAX    REF   VOL   PAGE  DATE
0H        N,GAMMA       EXPT-JOUR   THR              PR    61  152     2/42  UI    MANLEY<.F
0
0
0H   001  TOTAL XSECT   EXPT-JOUR   3.  -3  1.0  2   PR    74  364     8/48  COR   JONES.FRE
0H   001  TOTAL XSECT   EXPT-JOUR   3.  -3  1.0  2   PR    76  1750    0/49  COL   MELKONIAN.
0H   001  TOTAL XSECT   THEO-JOUR           6.   6   PR    82  60      4/51  COR   SALPETER.E
0H   001  ELASTIC       EXPT-JOUR   1.4  7           PR    75  1819    6/49  LAS   BARSCHALL
0H   001  ELASTIC       THEO-JOUR   0.   0  2.0  7   PR    76  38      7/49  COR   BETHE.S-A
0H   001  DIFF ELASTIC  EXPT-JOUR   1.4  7           PR    75  1819    6/49  LAS   BARSCHALL<
0H   001  DIFF ELASTIC  EXPT-ABST   1.4  7           PR    67  223     7/52  CRC   CROSS.9DTO
0H   001  THRMLSCATLAW  THEO-JOUR   THR              PR    60  18      7/41  GWU   SACHS<TELL
0H   001  THRMLSCATLAW  THEO-JOUR      -2       -1   PR    60  18      7/41  GWU   SACHS<TELL
0H   001  N,GAMMA       THEO-JOUR           6.   6   PR    82  60      4/51  COR   SALPETER.E
1                                                                            OCT.  1, 196
0ELEMENT   QUANTITY       TYPE          ENERGY          DOCUMENTATION        LAB
  S   A                              MIN      MAX    REF   VOL   PAGE  DATE
0D   002  TOTAL XSECT   EXPT-PROG   1.0  5  1.3  6   AFCU-2128 43      5/52  MIT   4ES LI(P,M
0D   002  DIFF ELASTIC  EVAL-REPT   1.0  5  1.4  7   NAA-SR-11980IV    4/67  AI    CAMPBELL+
1                                                                            OCT.  1, 196
0ELEMENT   QUANTITY       TYPE          ENERGY          DOCUMENTATION        LAB
  S   A                              MIN      MAX    REF   VOL   PAGE  DATE
0T   003  DIFF ELASTIC  EVAL-REPT   1.0  5  1.2  7   NAA-SR-11980IV    4/67  AI    CAMPBELL+
1                                                                            OCT.  1, 196
0ELEMENT   QUANTITY       TYPE          ENERGY          DOCUMENTATION        LAB
  S   A                              MIN      MAX    REF   VOL   PAGE  DATE
```

Figure 11. A sample page from the CINDA publication produced on a line printer. The magnetic tape which produced this listing was used as input to SETLST. The control cards which produced the transformation shown in Figure 10 are shown in Figures 12 and 13. The vertical lines indicate where the program inserted the locators !I14, !I01, etc., in accord with the instructions on the third and subsequent control cards shown in the next figure. See the next figures for an explanation of the circumscribed items.

```
ABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789                \
   12     0  120     1     7     8
1-1,2-4,5-7,10-21,24-32,35-39,42-46,49-63,64-67,69-71,72-109,110-1
!I14
!I01
!I02
!I03
!I04
!I05
!I06
!I07
!I08
!I09
!I10
!I11
 6!F1192
 $'
1EDITION      99 LIGNES      $
FINIS$
ABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789 /
 0 0 0
    >< 0
    >< 1
/!I03EVALUATION/ /!I03>EVALUATION/
/!I03TOTAL XSECT/ /!I03>TOTAL >X->SECT/
/!I03RESON PARAMS/ /!I03>RESON >PARAMS/
/!I03ELASTIC/ /!I03>ELASTIC/                        a
/!I03DIFF ELASTIC/ /!I03>DIFF >ELASTIC/
/!I03N PRODUCTION/ /!I03N >PRODUCTION/
/!I03NONELASTIC/ /!I03>NONELASTIC/
/!I03NONEL GAMMAS/ /!I03>NONEL !G2!T3D!G1!T1'S/
/!I03TOT INELASTIC/ /!I03>TOT >INELASTIC/            b
/!I03DIFF INELAST/ /!I03>DIFF >INELAST/
/!I03INELST GAMMA/ /!I03>INELST !G2!T3D!G1!T1/
/!I03N2N REACTION/ /!I03(N,2N) >REACTION/
/!I03THRMLSCATLAW/ /!I03>THRML>SCAT>LAW/
/!I03ABSORPTION/ /!I03>ABSORPTION/
/!I03DISAPPERANC/ /!I03>DISAPPERANCE/
/!I03N,GAMMA/  /!I03(N, !G2!T3D!G1!T1)/
/!I03SPECT NGAMMA/ /!I03>SPECT (N, !G2!T3D!G1!T1)/
/!I03N,PROTON/ /!I03(N, P)/
/!I03N,DEUTERON/ /!I03(N, D)/                        c
/!I03N,ALPHA/ /!I03(N, !G2!T3A!G1!T1)/
```

Figure 12. The control cards and a portion of the substitution table used to produce the output shown in Figure 10. The items marked by a circle show, respectively, how the character stream is modified to: a. provide initial caps; b. produce Greek letters; c. provide contractions and insert parentheses, etc. See Figure 13 for the rest of the substitution table.

```
/!I07   DOCUMENTATIO/  /!I07 >D>O>C>U>M>E>N>T>A>T>I>O>N/
/!I08N!I09/ /!I09/
/!I05   EN!I06GY/ /!I13>E>N>E>R>G>Y/
/!I01ELE!I02MEN/ /!I12>E>L>E>M>E>N>T/
/!I01BE/ /!I01>BE/
/!I01LI/ /!I01>LI/
/!I01HE/ /!I01>HE/
/!I0200/ /!I02/
/!I020/ /!I02/
/!I02 A/ //
/!I01 S/ /!I15/
/!I141/    /!I14/
/!I140/  /!I16 /
/!I/ /!I/
/+-/ /!G2!T3>)!G1!T1/
/     /  /!30/
/    / /!24/
/   /  /!18/
/  / /!12/
/[/ /(/
/]/ /)/
/*/ / /
/+/  /!G2!T3>1!G1!T1/
/=/  /!G2!T3>;!G1!T1/
/A/ />A/
/W/ />W/
/X/ />X/
/Y/ />Y/
/Z/ />Z/
/ -/  /!08!G2!T31!G1!T1/
/  -/ /!14!G2!T31!G1!T1/
/   -/        /!20!G2!T31!G1!T1/
/    -/       /!26!G2!T31!G1!T1/
/     -/      /!32!G2!T31!G1!T1/
FINIS
```

Figure 13. A continuation of the control cards used for the CINDA publication. The tagged items show how the character stream was modified to: a. restore the letters ER in ENERGY which might otherwise have been lost; b. remove the leading zeros in the isotopic number; c. replace the characters S A by a new line signal; d. ensure that the letter I in the locator remains lower case; e. go to another grid for mathematical symbols; f. generate initial caps everywhere except as indicated in the longer substitution strings; g. compensate by inserting spaces for different width values of a minus sign and a decimal point; h. replace successive blanks by appropriate width spaces.

OMNITAB, USING ORTHO SUBROUTINE			7094	EXAMPLE 13
BETA-HAT (Y1)	COUNT	BETA-HAT (Y2)		COUNT
1.0012617	2.892	.99999949		6.292
.99780273	2.658	.10000013		5.886
.99932661	3.173	.0099999756		5.613
1.0001755	3.756	.0010000018		5.745
.99998569	4.844	.00009999866		5.873
1.0000004	6.398	.000010000004		6.398
AVERAGE =	3.954		AVERAGE =	5.968

OMNITAB, USING ORTHO SUBROUTINE			1108	EXAMPLE 14
BETA-HAT (Y1)	COUNT	BETA-HAT (Y2)		COUNT
1.0064697	2.189	.99999990		7.000
.99902344	3.010	.099999700		5.523
.99975586	3.612	.010000125		4.903
.99996948	4.515	.00099998200		4.745
1.0000100	5.000	.00010000109		4.963
.99999968	6.495	.000009999778		5.654
AVERAGE =	4.137		AVERAGE =	5.464

ORTHO, WITH RE-ORTHOGONALIZATION OMITTED			1108	EXAMPLE 15
BETA-HAT (Y1)	COUNT	BETA-HAT (Y2)		COUNT
-1216.5426	-3.085	.98419463		1.801
2752.0557	-3.439	.13523918		.453
-1057.0931	-3.025	-.0034660707		-.129
146.97336	-2.164	.0028495983		-.267
-7.3080225	-.919	-.0000049256487		-.021
1.1663037	.779	.0000012094996		.679
AVERAGE =	-1.976		AVERAGE =	.419

Figure 14. A sample of a formatted output which can be typeset easily using SETLST and a monowidth typeface. See Figure 5 for the typeset version of this table.

6. Acknowledgements

The authors wish to express their appreciation for the valuable advice and guidance provided by Mr. Rubin Wagner, Chief of the Computer-Assisted Printing Section of the NBS Office of Technical Information and Publications. His willingness to share his expertise - based on long experience in traditional and modern methods of typesetting and composition - has contributed significantly to our progress in programming for computer-assisted printing in general, and to the preparation of this report in particular. Valuable assistance was rendered by Mrs. Daris Jones in the typing of the manuscript and in the preparation of the camera-ready copy from which this report was produced.

7. References

[1] William R. Bozman, Phototypesetting of Computer Output National Bureau of Standards Technical Note 170 (June 25, 1962) available from the Superintendent of Documents, U.S. Government Printing Office, Washington, D.C. 20402.

[2] Carla G. Messina and Joseph Hilsenrath, Edpac: Utility Programs for Computer-Assisted Editing, Copy-Production, and Data Retrieval, NBS Technical Note 470 (January 1969) available from the Superintendent of Documents, U.S. Government Printing Office, Washington, D.C. 20402.

[3] Michael P. Barnett, Computer Typesetting/Experiments and Prospects, The MIT Press, Cambridge, Massachusetts 1965.

APPENDIX I

The program listings given here were written in such a fashion as to permit their easy implementation on various computers and compilers of different vintage. If a compiler permits use of labeled common blocks, the operating efficiency of these programs can, possibly, be increased by certain indicated changes. The markings to the right of the listings indicate which lines of the program must be replaced by the corresponding marked lines in APPENDIX II to take advantage of labeled common. The < means insert and the period and brace denote lines to be replaced.

367-112 O - 70 - 2

```
C                              SETLST                                   TYST   1
C    PROGRAM DEVELOPED AT NBS-NSRDS   WASH D.C.  BY C. MESSINA 1/68     TYST   2
C                                                                       TYST   3
C        THE PROGRAM SETLST WILL GENERATE SUITABLE OUTPUT RECORDS FOR   TYST   4
C    TYPESETTING FROM FIXED FIELD RECORDS.                              TYST   5
C        ALL INPUT RECORDS MUST HAVE THE SAME FORMAT EXCEPT FOR THOSE   TYST   6
C    RECORDS WHICH ARE DELETED FROM THE FILE BY THE SUBROUTINE (AMATCH).TYST   7
C        TYPESETTING GRID CHANGES AND OTHER STRING SUBSTITUTIONS ARE    TYST   8
C    INSERTED BY THE SUBROUTINE (SUBST).                                TYST   9
C        THIS PROGRAM REMOVES THE TRAILING BLANKS FROM EVERY FIELD TO   TYST  10
C    BE TYPESET AND WRITES OUT PACKED RECORDS OF VARIABLE LENGTH        TYST  11
C                                                                       TYST  13
C                            CONTROL CARDS                              TYST  14
C        THE FIRST CARD IS SIMILAR TO THE FIRST CARD OF THE SUBROUTINE  TYST  15
C    (SUBST).  THE 47TH CARD COLUMN CONTAINS A BLANK AND THE 50TH CARD  TYST  16
C    COLUMN CONTAINS THE END OF RECORD SYMBOL.  THE END OF RECORD       TYST  17
C    SYMBOL MUST NOT APPEAR IN THE RECORDS TO BE PROCESSED FROM         TYST  18
C    UNIT IRTAPE.                                                       TYST  19
C                                                                       TYST  20
C        THE SECOND CARD HAS SIX SWITCHES IN (6I5) FORMAT.              TYST  21
C    ICOL     - CONTAINS THE NUMBER OF CARD FIELDS TO BE PROCESSED.     TYST  22
C    ITEST    - IS -1 FOR PRINTING.                                     TYST  23
C             -    0 FOR PRINTING AND TAPE WRITING ON UNIT IPTAPE.      TYST  24
C             -    1 FOR TAPE WRITING ON UNIT IPTAPE.                   TYST  25
C    LENGTH   - IS THE NUMBER OF CHARACTERS IN AN INPUT RECORD READ     TYST  26
C               FROM UNIT IRTAPE.                                       TYST  27
C    KEEP     - IS ZERO WHEN ALL FIELDS ARE TO BE PRESENT IN THE OUTPUT TYST  28
C               REGARDLESS OF CONTENT.                                  TYST  29
C             - IS NON ZERO WHEN ANY FIELD WHICH IS COMPLETELY BLANK IS TYST  30
C               TO BE IGNORED IN THE OUTPUT (THE INSERT STRING IS       TYST  31
C               ALSO OMITTED.).                                         TYST  32
C    IRTAPE   - IS THE UNIT ON WHICH THE INPUT RECORDS ARE EXPECTED.    TYST  33
C    IPTAPE   - IS THE UNIT ON WHICH TO WRITE THE OUTPUT RECORDS(NOT    TYST  34
C               THE PRINTER).                                           TYST  35
C                                                                       TYST  36
C        THE THIRD CARD CONTAINS THE FIELD DELIMITERS IN THE ORDER      TYST  37
C    WISHED IN THE OUTPUT.  THE FORMAT IS FREE FIELD AND THE DELIMITERS TYST  38
C    ARE SEPARATED BY ONE OR AS MANY SPACES AS DESIRED.                 TYST  39
C    IN THE FOLLOWING EXAMPLE OF 3 COLUMN FIELDS                        TYST  40
C   73 80      1 70     73 80                                           TYST  41
C    THE INFORMATION LOCATED STARTING AT THE 73RD CHARACTER AND         TYST  42
C    EXTENDING TO INCLUDE THE 80TH CHARACTER IS TYPESET, FOLLOWED       TYST  43
C    BY THE INFORMATION CONTAINED IN CHARACTER FIELDS 1 TO 70 AND       TYST  44
C    ENDING WITH THE INFORMATION IN FIELDS 73 TO 80.  REPEATING FIELD   TYST  45
C    DESIGNATIONS ON THE THIRD CONTROL CARD, DUPLICATES THE INFORMATION TYST  46
C    IN THE OUTPUT RECORD.                                              TYST  47
C                                                                       TYST  48
C        THE FOLLOWING ICOL NUMBER OF CARDS CONTAIN THE TYPESETTING     TYST  49
C    INSERT STRINGS.  IN THE OUTPUT RECORDS, THESE INSERT STRINGS       TYST  50
C    ARE PLACED BEFORE THE INFORMATION TAKEN FROM THE INPUT RECORDS.    TYST  51
C    THESE STRINGS ARE PUNCHED STARTING IN CARD COLUMN ONE AND ARE      TYST  52
C    CONSIDERED TERMINATED BY A BLANK CHARACTER.  IF NO STRING IS       TYST  53
C    TO BE INSERTED, A BLANK CARD IS PLACED IN THE INSERT STRING TABLE  TYST  54
C    AT THE APPROPRIATE POSITION.                                       TYST  55
C                                                                       TYST  56
C        AFTER THE INSERT STRING TABLE, A CARD IN (1I2,78A1) FORMAT     TYST 570
C    CONTAINS THE IPTAPE HEADER. THE LENGTH OF THE HEADER IS FOLLOWED   TYST 571
C    IMMEDIATELY BY THE HEADER.                                         TYST 572
```

```
C          THE CONTROL CARDS FOR                                             TYST 573
C     SUBROUTINE (AMATCH) ARE PLACED FOLLOWED BY THE CONTROL CARDS FOR      TYST 58
C     SUBROUTINE (SUBST).                                                    TYST 59
C                                                                            TYST 60
C                                                                            TYST 62
      DIMENSION IA(84),ID(136),INS(400),N(40),NA(40),NB(100),IC(700),       TYST 63
     1IB(4100)                                                               TYST 64  a
      ITAPE=5                                                                TYST 65
      IOTAPE=6                                                               TYST 66
      READ (ITAPE,9) (IA(I),I=1,80)                                          TYST 67
      WRITE (IOTAPE,19) (IA(I),I=1,80)                                       TYST 68
      READ (ITAPE,89) ICOL,ITEST,LENGTH,KEEP,IRTAPE,IPTAPE                   TYST 69
      WRITE (IOTAPE,99) ICOL,ITEST,LENGTH,KEEP,IRTAPE,IPTAPE                 TYST 70
      IF (ICOL) 2,2,1                                                        TYST 71
1     IF (ICOL-20) 3,3,2                                                     TYST 72
2     WRITE (IOTAPE,129)                                                     TYST 73
      GO TO 9999                                                             TYST 74
3     READ (ITAPE,9) (ID(I),I=1,80)                                          TYST 75
      L=2*ICOL                                                               TYST 76
7     DO 8 I=1,L                                                             TYST 77
      N(I)=0                                                                 TYST 78
      NB(I)=0                                                                TYST 79
8     NA(I)=0                                                                TYST 80
      K=0                                                                    TYST 81
      DO 100 I=1,L                                                           TYST 82
10    K=K+1                                                                  TYST 83
      IF (ID(K)-IA(47)) 30,20,30                                             TYST 84
20    IF (K-80) 10,25,25                                                     TYST 85
25    WRITE (IOTAPE,49) L,(ID(J),J=1,80)                                     TYST 86
      GO TO 9999                                                             TYST 87
30    DO 40 J=27,36                                                          TYST 88
      IF (ID(KA- A(J)) 40,50,40                                              TYST 89
40    CONTINUE                                                               TYST 90
      GO TO 10                                                               TYST 91
50    N(I)=J-27                                                              TYST 92
      K=K+1                                                                  TYST 93
      DO 60 J=27,36                                                          TYST 94
      IF (ID(K)-IA(J)) 60,70,60                                              TYST 95
60    CONTINUE                                                               TYST 96
      GO TO 100                                                              TYST 97
70    N(I)=10*N(I)+J-27                                                      TYST 98
      K=K+1                                                                  TYST 99
      DO 80 J=27,36                                                          TYST100
      IF (ID(K)-IA(J)) 80,90,80                                              TYST101
80    CONTINUE                                                               TYST102
      GO TO 100                                                              TYST103
90    N(I)=10*N(I)+J-27                                                      TYST104
100   CONTINUE                                                               TYST105
      J=0                                                                    TYST106
      DO 120 I=1,L                                                           TYST107
      IF (N(I)) 110,105,110                                                  TYST108
105   J=J+1                                                                  TYST109
110   IF (N(I)-132) 120,120,122                                              TYST110
120   CONTINUE                                                               TYST111
      IF (J-L) 125,9999,9999                                                 TYST112
121   I=3                                                                    TYST113
      N(I)=LENGTH                                                            TYST114
122   WRITE (IOTAPE,39) I,N(I)                                               TYST115
```

```
      GO TO 9999                                                        TYST116
  125 J=1                                                               TYST117
      K=L-1                                                             TYST118
      DO 145 I=1,K,2                                                    TYST119
      IF (N(I)-N(I+1)) 140,140,130                                      TYST120
  130 WRITE (IOTAPE,59) N(I),N(I+1),I                                   TYST121
      GO TO 9999                                                        TYST122
  140 WRITE (IOTAPE,29)    J,N(I),N(I+1)                                TYST123
  145 J=J+1                                                             TYST124
      WRITE (IOTAPE,69)  ICOL                                           TYST125
      N3=1                                                              TYST126
      DO 190 I=1,ICOL                                                   TYST127
      READ ( ITAPE,9) (ID(K),K=1,80)                                    TYST128
      WRITE (IOTAPE,19) (ID(K),K=1,80)                                  TYST129
      J=1                                                               TYST130
  150 IF (ID(J)-IA(47)) 160,180,160                                     TYST131
  160 INS(N3)=ID(J)                                                     TYST132
      N3=N3+1                                                           TYST133
      J=J+1                                                             TYST134
      IF (N3-400) 150,150,170                                           TYST135
  170 WRITE (IOTAPE,79)                                                 TYST136
      GO TO 9999                                                        TYST137
  180 NA(I)=J-1                                                         TYST138
  190 CONTINUE                                                          TYST139
      K=1                                                               TYST140
      NB(1)=1                                                           TYST141
      DO 191 I=2,L,2                                                    TYST142
      NB(I)=NA(K)+NB(I-1)-1                                             TYST143
      NB(I+1)=NA(K)+NB(I-1)                                             TYST144
  191 K=K+1                                                             TYST145
      K1=0                                                              TYST1451
      READ (ITAPE,139) K1,(IB(J),J=1,78)                                TYST1452
      WRITE (IOTAPE,149) K1,(IB(I),I=1,78)                              TYST1453
      IF (K1)     9999,201,201                                          TYST1454
  201 CALL AMATCH(IA,ID,LENGTH,0,MATCH)                                 TYST146
      CALL SUBST(IC,K,0)                                                TYST147
      IF (LENGTH) 121,121,192                                           TYST148
  192 IF (LENGTH-133) 193,193,121                                       TYST149
  193 IF (IPTAPE) 194,195,196                                           TYST150
  194 WRITE (IOTAPE,119) IRTAPE,IPTAPE                                  TYST151
      GO TO 9999                                                        TYST152
  195 IPTAPE=3                                                          TYST153
  196 IF (IRTAPE) 194,197,198                                           TYST154
  197 IRTAPE=5                                                          TYST155
  198 K=0                                                               TYST156
  200 READ (IRTAPE,9,END=900,ERR=900) (ID(I),I=1,LENGTH)                TYST158
      CALL AMATCH(IA,ID,LENGTH,1,MATCH)                                 TYST159
      IF (MATCH) 210,210,200                                            TYST160
  210 DO 300 L=1,ICOL                                                   TYST161
      L1=2*L-1                                                          TYST162
      L2=2*L                                                            TYST163
      N1=N(L1)                                                          TYST164
      N2=N(L2)                                                          TYST165
      IF (N1) 300,300,220                                               TYST166
  220 IF (NA(L)) 250,250,230                                            TYST167
  230 N3=NB(L1)                                                         TYST168
      N4=NB(L2)                                                         TYST169
      DO 240 I=N3,N4                                                    TYST170
```

```
      K=K+1                                                             TYST171
240   IC(K)=INS(I)                                                      TYST172
250   J=N2                                                              TYST173
      DO 260 I=N1,N2                                                    TYST174
      IF (ID(J)-IA(47)) 270,260,270                                     TYST175
260   J=J-1                                                             TYST176
      IF (KEEP) 262,268,262                                             TYST177
262   IF (NA(L)) 300,300,264                                            TYST178
264   K=K-(N4-N3+1)                                                     TYST179
      GO TO 300                                                         TYST180
268   J=J+1                                                             TYST181
270   DO 280 I=N1,J                                                     TYST182
      K=K+1                                                             TYST183
280   IC(K)=ID(I)                                                       TYST184
300   CONTINUE                                                          TYST185
800   K2=K1+1                                                           TYST186
      CALL SUBST(IC,K,1)                                                TYST187 • d
      K1=K1+K                                                           TYST188
      J=0                                                               TYST189
      DO 805 I=K2,K1                                                    TYST190
      J=J+1                                                             TYST191
805   IB(I)=IC(J)                                                       TYST192
      K=0                                                               TYST193
807   CALL NPRINT (K1,ITEST,IB,IA(47),IOTAPE,IPTAPE,0)                  TYST194 • e
      GO TO 200                                                         TYST205
900   K1=K1+4                                                           TYST206
      IB(K1-3)=IA(47)                                                   TYST207
      IB(K1-2)=IA(50)                                                   TYST208
      IB(K1-1)=IA(47)                                                   TYST209
      IB(K1)=IA(50)                                                     TYST210
      CALL NPRINT (K1,ITEST,IB,IA(47),IOTAPE,IPTAPE,1)                  TYST211 • f
9999  STOP                                                              TYST226
9     FORMAT (132A1)                                                    TYST227
19    FORMAT (1X,120A1)                                                 TYST228
29    FORMAT ( 7H FIELD  ,1I2, 11H GOES FROM  ,1I3,4H TO ,1I3)          TYST229
39    FORMAT (57H ONLY LINES WITH 132 OR LESS CHARACTERS ARE ALLOWED, THTYST230
     1E  ,1I6,7H FIELD  ,/11H SPECIFIED  ,1I6,18H CHARACTERS. STOP.  )  TYST231
49    FORMAT (1X,1I4,53H NUMBERS WERE NOT GIVEN ON THE FOLLOWING CARD. STYST232
     1TOP.   /80A1)                                                     TYST233
59    FORMAT (20H THE PAIR OF NUMBERS  ,1I6, 5H AND ,1I6, 25H SPECIFYINGTYST234
     1 FIELD NUMBER  ,1I6, /42H ARE NOT GIVEN IN THE CORRECT ORDER. STOPTYST235
     2.  )                                                              TYST236
69    FORMAT (15H THE FOLLOWING  ,1I4,34H CARDS CONTAIN THE INSERT STRINTYST237
     1GS.  /)                                                           TYST238
79    FORMAT (60H0THE INSERT STRINGS TAKE UP MORE THAN 400 CHARACTERS.STTYST239
     1OP.      )                                                        TYST240
89    FORMAT (10I5)                                                     TYST241
99    FORMAT (1X,10I5)                                                  TYST242
119   FORMAT (47H ILLEGAL UNIT SPECIFIED ON SECOND CONTROL CARD. 2I6)   TYST245
129   FORMAT ( 70H0CURRENT DIMENSION STATEMENTS ALLOW FOR A MAXIMUM OF 2TYST246
     10 FIELDS. STOP.        )                                          TYST247
139   FORMAT (1I2,78A1)                                                 TYST248
149   FORMAT (1X,1I2,78A1)                                              TYST249
       END                                                              TYST250
```

```
C                    KWIND, TYPESETTING FOR KWIC INDEXES                KWIC0090
      DIMENSIONIA(84),IC(700),ID(136),INS(80),N(8),NA(8),NB(20),IB(4100)KWIC0100
      ITAPE=5                                                           KWIC0110
      IOTAPE=6                                                          KWIC0120
      READ (ITAPE,9) (IA(I),I=1,80)                                     KWIC0130
      WRITE (IOTAPE,19) (IA(I),I=1,80)                                  KWIC0140
      READ (ITAPE,89) ITEST,LENGTH,IRTAPE,IPTAPE                        KWIC0150
      WRITE (IOTAPE,99) ITEST,LENGTH,IRTAPE,IPTAPE                      KWIC0160
      READ (ITAPE,9) (ID(I),I=1,80)                                     KWIC0170
      DO 5 I=1,8                                                        KWIC0180
      N(I)=0                                                            KWIC0190
      NB(I)=0                                                           KWIC0198
5     NA(I)=0                                                           KWIC0200
      K=0                                                               KWIC0210
      DO 100 I=1,8                                                      KWIC0220
10    K=K+1                                                             KWIC0230
      IF (ID(K)-IA(47)) 30,20,30                                        KWIC0240
20    IF (K-80) 10,25,25                                                KWIC0250
25    WRITE (IOTAPE,49) (ID(J),J=1,80)                                  KWIC0260
      GO TO 9999                                                        KWIC0270
30    DO 40 J=27,36                                                     KWIC0280
      IF (ID(K)-IA(J)) 40,50,40                                         KWIC0290
40    CONTINUE                                                          KWIC0300
      GO TO 10                                                          KWIC0310
50    N(I)=J-27                                                         KWIC0320
      K=K+1                                                             KWIC0330
      DO 60 J=27,36                                                     KWIC0340
      IF (ID(K)-IA(J)) 60,70,60                                         KWIC0350
60    CONTINUE                                                          KWIC0360
      GO TO 100                                                         KWIC0370
70    N(I)=10*N(I)+J-27                                                 KWIC0380
      K=K+1                                                             KWIC0390
      DO 80 J=27,36                                                     KWIC0400
      IF (ID(K)-IA(J)) 80,90,80                                         KWIC0410
80    CONTINUE                                                          KWIC0420
      GO TO 100                                                         KWIC0430
90    N(I)=10*N(I)+J-27                                                 KWIC0440
100   CONTINUE                                                          KWIC0450
      J=0                                                               KWIC0460
      DO 120 I=1,8                                                      KWIC0470
      IF (N(I)) 110,105,107                                             KWIC0480
105   J=J+1                                                             KWIC0490
107   IF (N(I)-136) 120,120,110                                         KWIC0500
110   WRITE (IOTAPE,39) I,N(I)                                          KWIC0510
      GO TO 9999                                                        KWIC0520
120   CONTINUE                                                          KWIC0530
      IF (J-8)    125,9999,9999                                         KWIC0540
125   J=1                                                               KWIC0550
      DO 145 I=1,7,2                                                    KWIC0560
      IF (N(I)-N(I+1)) 140,140,130                                      KWIC0570
130   WRITE (IOTAPE,59) N(I),N(I+1),I                                   KWIC0580
      GO TO 9999                                                        KWIC0590
140   WRITE (IOTAPE,29)    J,N(I),N(I+1)                                KWIC0600
145   J=J+1                                                             KWIC0610
      WRITE (IOTAPE,69)                                                 KWIC0620
      N3=1                                                              KWIC0630
      DO 190 I=1,8                                                      KWIC0640
```

```
      READ ( ITAPE,9) (ID(K),K=1,80)                                  KWIC0650
      WRITE (IOTAPE,19) (ID(K),K=1,80)                                KWIC0660
      J=1                                                             KWIC0670
150   IF (ID(J)-IA(47)) 160,180,160                                   KWIC0680
160   INS(N3)=ID(J)                                                   KWIC0690
      N3=N3+1                                                         KWIC0700
      J=J+1                                                           KWIC0710
      IF (N3-80) 150,150,170                                          KWIC0720
170   WRITE (IOTAPE,79)                                               KWIC0730
      GO TO 9999                                                      KWIC0740
180   NA(I)=J-1                                                       KWIC0750
190   CONTINUE                                                        KWIC0760
      K=1                                                             KWIC0770
      NB(1)=1                                                         KWIC0780
      DO 195 I=2,16,2                                                 KWIC0790

      NB(I)=NA(K)+NB(I-1)-1                                           KWIC0800
      NB(I+1)=NA(K)+NB(I-1)                                           KWIC0810
195   K=K+1                                                           KWIC0820
      READ (ITAPE,139) K1,(IB(J),J=1,78)
      WRITE (IOTAPE,149) K1,(IB(I),I=1,78)
      CALL AMATCH(IA,ID,LENGTH,0,MATCH)                               KWIC0830 }
      CALL SUBST(ID,1,0)                                              KWIC0832 } h
      IT = N(6)                                                       KWIC0834
      K=0                                                             KWIC0840
200   READ (IRTAPE,9,END=900,ERR=900) (ID(I),I=1,LENGTH)              KWIC0850
201   N1=N(1)                                                         KWIC0860
      N2=N(2)                                                         KWIC0870
      CALL AMATCH(IA,ID,LENGTH,1,MATCH)                               KWIC0880
      IF (MATCH) 205,205,200                                          KWIC0890
205   IF (N(1)) 270,270,210                                           KWIC0900
210   IF (NA(1)) 240,240,220                                          KWIC0910
220   N3=NB(1)                                                        KWIC0920
      N4=NB(2)                                                        KWIC0930
      DO 230 I=N3,N4                                                  KWIC0940
      K=K+1                                                           KWIC0950 }
230   IC(K)=INS(I)                                                    KWIC0960 }
240   DO 250 I=N1,N2                                                  KWIC0970 } i
      K=K+1                                                           KWIC0980 }
250   IC(K)=ID(I)                                                     KWIC0990
270   IF (N(3)) 500,500,280                                           KWIC1000
280   N1=N(3)+1                                                       KWIC1010
      N2=N(4)                                                         KWIC1020
      J=N1-1                                                          KWIC1030
      DO 300 I=N1,N2                                                  KWIC1040
      IF (ID(I-1)-IA(47)) 300,290,300                                 KWIC1050
290   IF (ID(I)-IA(47)) 300,350,300                                   KWIC1060
300   CONTINUE                                                        KWIC1070
      IF (NA(4)) 330,330,310                                          KWIC1080
310   N3=NB(7)                                                        KWIC1090
      N4=NB(8)                                                        KWIC1100
      DO 320 I=N3,N4                                                  KWIC1110
      K=K+1                                                           KWIC1120
320   IC(K)=INS(I)                                                    KWIC1130
330   N1=N(3)                                                         KWIC1140
      DO 340 I=N1,N2                                                  KWIC1150
      K=K+1                                                           KWIC1160
340   IC(K)=ID(I)                                                     KWIC1170
      GO TO 500                                                       KWIC1180
```

```
350   IF (I-N1) 410,410,360                                KWIC1190
360   IF (NA(2)) 390,390,370                               KWIC1200
370   N3=NB(3)                                             KWIC1220
      N4=NB(4)                                             KWIC1230
      DO 380 J=N3,N4                                       KWIC1240
      K=K+1                                                KWIC1250
380   IC(K)=INS(J)                                         KWIC1260
390   J=I-2                                                KWIC1270
      N1=N(3)                                              KWIC1280
      DO 400 I=N1,J                                        KWIC1290
      K=K+1                                                KWIC1300
400   IC(K)=ID(I)                                          KWIC1310
410   J=J+1                                                KWIC1320
      DO 430 I=J,N2                                        KWIC1330
      IF (ID(I)-IA(47)) 440,430,440                        KWIC1340


430   CONTINUE                                             KWIC1350
      GO TO 500                                            KWIC1360
440   IF (NA(3)) 470,470,450                               KWIC1370
450   N3=NB(5)                                             KWIC1380
      N4=NB(6)                                             KWIC1390
      DO 460 J=N3,N4                                       KWIC1400
      K=K+1                                                KWIC1410
460   IC(K)=INS(J)                                         KWIC1420
470   DO 480 J=I,N2                                        KWIC1430
      K=K+1                                                KWIC1440
480   IC(K)=ID(J)                                          KWIC1450
500   IF (N(5)) 720,720,510                                KWIC1460
510   N1=N(5)+1                                            KWIC1470
      N2=N(6)                                              KWIC1480
      J=N1-1                                               KWIC1490
      DO 530 I=N1,N2                                       KWIC1500
      IF (ID(I-1)-IA(47)) 530,520,530                      KWIC1510
520   IF (ID(I)-IA(47)) 530,590,530                        KWIC1520
530   CONTINUE                                             KWIC1530
      IF (NA(7)) 560,560,540                               KWIC1540
540   N3=NB(13)                                            KWIC1550
      N4=NB(14)                                            KWIC1560
      DO 550  I=N3,N4                                      KWIC1570
      K=K+1                                                KWIC1580
550   IC(K)=INS(I)                                         KWIC1590
560   N1=N(5)                                              KWIC1600
      DO 570 I=N1,N2                                       KWIC1610
      K=K+1                                                KWIC1620
570   IC(K)=ID(I)                                          KWIC1630
      GO TO 720                                            KWIC1640
590   IF(I-N1) 650,650,600                                 KWIC1650
600   IF (NA(5)) 630,630,610                               KWIC1660
610   N3=NB(9)                                             KWIC1670
      N4=NB(10)                                            KWIC1680
      DO 620 J=N3,N4                                       KWIC1690
      K=K+1                                                KWIC1700
620   IC(K)=INS(J)                                         KWIC1710
630   J=I-2                                                KWIC1730
      N1=N(5)                                              KWIC1740
      DO 640 I=N1,J                                        KWIC1750
      K=K+1                                                KWIC1760
640   IC(K)=ID(I)                                          KWIC1770
```

```
650    J=J+1                                                             KWIC1780
       DO 660 I=J,N2                                                     KWIC1790
       IF (ID(I)-IA(47)) 670,660,670                                     KWIC1800
660    CONTINUE                                                          KWIC1810
       GO TO 720                                                         KWIC1820
670    IF (NA(6)) 700,700,680                                            KWIC1830
680    N3=NB(11)                                                         KWIC1840
       N4=NB(12)                                                         KWIC1850
       DO 690 J=N3,N4                                                    KWIC1860
       K=K+1                                                             KWIC1870
690    IC(K)=INS(J)                                                      KWIC1880
700    DO 710 J=I,N2                                                     KWIC1890
       K=K+1                                                             KWIC1900
710    IC(K)=ID(J)                                                       KWIC1910
720    IF (N(7)) 800,800,730                                             KWIC1920
730    N1=N(7)                                                           KWIC1930

       N2=N(8)                                                           KWIC1940
       IF (NA(8)) 770,770,740                                            KWIC1950
740    N3=NB(15)                                                         KWIC1960
       N4=NB(16)                                                         KWIC1970
       DO 760 I=N3,N4                                                    KWIC1980
       K=K+1                                                             KWIC1990
760    IC(K)=INS(I)                                                      KWIC2000
770    DO 780 I=N1,N2                                                    KWIC2010
       K=K+1                                                             KWIC2020
780    IC(K)=ID(I)                                                       KWIC2030
800    K2=K1+1                                                           KWIC2031
       CALL SUBST(IC,K,1)                                                KWIC2032 • j
       K1=K1+K                                                           KWIC2033
       J=0                                                               KWIC2034
       DO 805 I=K2,K1                                                    KWIC2035
       J=J+1                                                             KWIC2036
805    IB(I)=IC(J)                                                       KWIC2037
       K=0                                                               KWIC2038
807    CALL NPRINT(K1,ITEST,IB,IA(47),IOTAPE,IPTAPE,0)                   KWIC2050 • k
850    GO TO 200                                                         KWIC2060
900    K1=K1+4                                                           KWIC2070
C IA(50) CONTAINS THE END OF RECORD SYMBOL,IA(47) CONTAINS A BLANK.      KWIC2080
       IB(K1-3)=IA(47)                                                   KWIC2090
       IB(K1-2)=IA(50)                                                   KWIC2100
       IB(K1-1)=IA(47)                                                   KWIC2180
       IB(K1)=IA(50)                                                     KWIC2190
       CALL NPRINT(K1,ITEST,IB,IA(47),IOTAPE,IPTAPE,1)                   KWIC2200 • l
9999   STOP                                                              KWIC2350
9      FORMAT (136A1)                                                    KWIC2360
19     FORMAT (1X,131A1)                                                 KWIC2370
29     FORMAT ( 7H FIELD  ,1I2, 11H GOES FROM  ,1I3,4H TO ,1I3)          KWIC2380
39     FORMAT (57H ONLY LINES WITH 136 OR LESS CHARACTERS ARE ALLOWED, THKWIC2390
      1E  ,1I6,7H FIELD  ,/11H SPECIFIED  ,1I6,18H CHARACTERS. STOP.  )  KWIC2400
49     FORMAT(58H EIGHT NUMBERS WERE NOT GIVEN ON THE FOLLOWING CARD. STOKWIC2410
      1P.  /1X,131A1)                                                    KWIC2420
59     FORMAT (20H THE PAIR OF NUMBERS  ,1I6, 5H AND ,1I6, 25H SPECIFYINGKWIC2430
      1 FIELD NUMBER  ,1I6, /42H ARE NOT GIVEN IN THE CORRECT ORDER. STOPKWIC2440
      2.  )                                                              KWIC2450
69     FORMAT (50H THE FOLLOWING 8 CARDS CONTAIN THE INSERT STRINGS    /)KWIC2460
79     FORMAT (60H THE INSERT STRINGS TAKE UP MORE THAN 80 CHARACTERS. STKWIC2470
      1OP.      )                                                        KWIC2480
89     FORMAT (10I5)                                                     KWIC2490
99     FORMAT (1X,10I5)                                                  KWIC2500
```

```
109   FORMAT ( 61HOTHE LAST LINE WRITTEN BEFORE A TAPE ERROR OR END OF FKWIC2510
     1ILE IS   /1X,131A1 /7 H STOP.   )                                 KWIC2520
139   FORMAT (1I2,78A1)
149   FORMAT (1X,1I2,78A1)
      END                                                               KWIC2530

      SUBROUTINE SUBST (IB,IW,ITYPE)                                    SUBS   1
C                                                                       SUBS   2
C                          SUBSTITUTE SUBROUTINE                        SUBS   3
C                                                                       SUBS   4
C     TEXTUAL SUBSTITUTION PROGRAM WRITTEN BY C. MESSINA NSRDS-NBS       SUBS  40
C     IB(999) IS THE STRING TO BE PROCESSED. ON RETURN FROM SUBST, IB   SUBS  41
C     CONTAINS THE REMADE LINE.                                         SUBS  42
C     IW IS THE LENGTH OF THE INPUT STRING IN IB.  ON RETURN FROM SUBST,SUBS  43
C     IW CONTAINS THE NEW LENGTH OF IB.                                 SUBS  44

C     ITYPE = 0 WHEN THE SUBSTITUTION TABLE IS READ IN AND ITYPE = 1    SUBS   5
C     WHEN THE SUBSTITUTION IS TO TAKE PLACE.                           SUBS   6
C                                                                       SUBS   7
C     THE INPUT DECK AT OBJECT TIME IS THE FOLLOWING SET OF CARDS       SUBS   8
C     THE FIRST CARD IS A DICTIONARY OF THE ALPHABET STARTING WITH THE  SUBS   9
C     LETTER A IN CARD COL ONE, A LETTER B IN COL 2 AND SO FORTH. THE   SUBS  10
C     NUMBERS FOLLOW THE ALPHABET STARTING WITH ZERO. COL 38 CONTAINS   SUBS  11
C     THE PRINT OUT STRING DELIMITER.  COL 47 CONTAINS A BLANK.         SUBS  12
C     THE SECOND CARD HAS A ZERO IN COL 2 IF NO CARDS ARE TO BE PUNCHED SUBS  13
C     1 IF THE PUNCH TAPE IS TO BE WRITTEN. THE NEXT 2I2 FIELDS ON THIS SUBS  14
C     CARD, IF POSITIVE NON ZERO INTEGERS, CONTAIN  THE IRTAPE NUMBER   SUBS  15
C     AND IPTAPE NUMBER, OTHERWISE THEY ARE SET TO IRTAPE=5 AND IPTAPE=3SUBS  16
C     THE THIRD AND FOURTH CARDS ARE BOTH IN A1,I3,2A1,I2 FORMATS.      SUBS  18
C     THE FIRST TWO FIELDS ARE IGNORED ON BOTH CARDS IN THIS VERSION.   SUBS  19
C     THE 3RD FIELD IS THE SHIFT TO UPPER CASE SYMBOL, THE 4TH IS SHIFT SUBS  21
C     TO LOWER CASE SYMBOL,  AND THE FIFTH IS THE SHIFT AND LOCK SWITCH SUBS  22
C     THAT IS 0 IF THE MODE IS  NOT SHIFT AND LOCK AND 1 IF IT IS.      SUBS  23
C     CARDS FIVE ET SEQ CONTAIN THE LIST OF STRINGS TO BE EXCHANGED.    SUBS  24
C     ON EACH CARD THE OLD RECORD OR STRING APPEARS ON THE LEFT SIDE ANDSUBS  25
C     THE NEW STRING ON THE RIGHT.  THE CHARACTER WHICH APPEARS IN CARD SUBS  26
C     COLUMN 1 IS THE STRING DELIMITER WHICH REMAINS IN FORCE FOR THAT  SUBS  27
C     CARD.  IT MAY, HOWEVER, CHANGE FROM CARD TO CARD.                 SUBS  28
C     THE  FORMAT IS PRESCRIBED.  A CHARACTER IN COL 1 DEFINES THE STARTSUBS  29
C     OF A STRING.  THE SAME CHARACTER MUST APPEAR AFTER THE END        SUBS  30
C     OF THE STRING.  THE THIRD APPEARANCE OF THE COLUMN 1 CHARACTER ON SUBS  31
C     THE CARD STARTS THE 2ND STRING AND THE FOURTH APPEARANCE ENDS IT. SUBS  32
C                                 EXAMPLE                               SUBS  33
C     /REAL/  /TRUE/                                                    SUBS  34
C     AFTER THE SUBSTITUTION LIST MUST COME A CARD WITH THE WORD  FINIS SUBS  36
C     STARTING IN CARD COLUMN ONE.                                      SUBS  37
C                                                                       SUBS  38
      DIMENSION IA(86),N(1000),IC(8000),IB(999)                         SUBS  39
      ITAPE=5                                                           SUBS  40
      IOTAPE=6                                                          SUBS  41
      IEND=0                                                            SUBS  42
      MAXIW=998                                                         SUBS  43
      IF (ITYPE) 20,20,560                                              SUBS  44
20    READ (ITAPE,840) (IA(J),J=1,80)                                   SUBS  45
      WRITE (IOTAPE,890) (IA(J),J=1,80)                                 SUBS  46
      READ (ITAPE,850) ITEST,IRTAPE,IPTAPE                              SUBS  47
      IF (IRTAPE) 30,30,60                                              SUBS  48
30    IRTAPE=5                                                          SUBS  49
60    IF (IPTAPE) 70,70,100                                             SUBS  53
70    IPTAPE=3                                                          SUBS  54
100   WRITE (IOTAPE,930) ITEST,IRTAPE,IPTAPE                            SUBS  58
```

```
      READ (ITAPE,860) IA(81),IWIDE1,IA(83),IA(85),LOCK1                SUBS 59
      WRITE (IOTAPE,920) IA(81),IWIDE1,IA(83),IA(85),LOCK1              SUBS 60
      READ (ITAPE,860) IA(82),IWIDE2,IA(84),IA(86),LOCK2                SUBS 61
      WRITE (IOTAPE,920) IA(82),IWIDE2,IA(84),IA(86),LOCK2              SUBS 62
  150 N1=0                                                              SUBS 69
      N3=1                                                              SUBS 70
C        START OF READING IN SUBSTITUTE LISTS                           SUBS 71
  160 READ (ITAPE,840) (IB(J),J=1,80)                                   SUBS 72
      N2=0                                                              SUBS 73
      N22=0                                                             SUBS 74
      IF (IB(1)-IA(6)) 210,170,210                                      SUBS 75
  170 IF (IB(2)-IA(9)) 210,180,210                                      SUBS 76
  180 IF (IB(3)-IA(14)) 210,190,210                                     SUBS 77

  190 IF (IB(4)-IA(9)) 210,200,210                                      SUBS 78
  200 IF (IB(5)-IA(19)) 210,440,210                                     SUBS 79
  210 DO 230 I=2,78                                                     SUBS 80
      IF (IB(I)-IB(1)) 230,220,230                                      SUBS 81
  220 IF (N2) 160,160,240                                               SUBS 82
  230 N2=I-1                                                            SUBS 83
  240 J=N2+3                                                            SUBS 84
      IF (J-79) 260,250,250                                             SUBS 85
  250 WRITE (IOTAPE,900) IB(1),(IB(I),I=1,80)                           SUBS 86
      GO TO 830                                                         SUBS 87
  260 K=J+1                                                             SUBS 88
      DO 270 I=J,79                                                     SUBS 89
      IF (IB(I)-IB(1)) 270,280,270                                      SUBS 90
  270 K=I+2                                                             SUBS 91
      GO TO 250                                                         SUBS 92
  280 DO 290 I=K,80                                                     SUBS 93
      IF (IB(I)-IB(1)) 290,300,290                                      SUBS 94
  290 N22=I-K+1                                                         SUBS 95
      GO TO 250                                                         SUBS 96
  300 N1=N1+2                                                           SUBS 97
      N(N1-1)=N2                                                        SUBS 98
      N(N1)=N22                                                         SUBS 99
      N4=N3+N2-1                                                        SUBS100
      IF (N4-7920) 320,320,310                                          SUBS101
  310 WRITE (IOTAPE,880) N4,N1                                          SUBS102
      GO TO 830                                                         SUBS103
  320 IF (N1-1000) 330,330,310                                          SUBS104
  330 J=2                                                               SUBS105
      DO 340 I=N3,N4                                                    SUBS106
      IC(I)=IB(J)                                                       SUBS107
  340 J=J+1                                                             SUBS108
      N3=N3+N2                                                          SUBS109
      IF (N22) 370,370,350                                              SUBS110
  350 N4=N3+N22-1                                                       SUBS111
      J=K                                                               SUBS112
      DO 360 I=N3,N4                                                    SUBS113
      IC(I)=IB(J)                                                       SUBS114
  360 J=J+1                                                             SUBS115
  370 IF (N2-38) 380,380,390                                            SUBS116
  380 K=42                                                              SUBS117
  390 K1=K+N22-1                                                        SUBS118
      J=N2+3                                                            SUBS119
      DO 400 L=J,80                                                     SUBS120
```

```
400   IB(L)=IA(47)                                              SUBS121
      IB(1)=IA(38)                                              SUBS122
      IB(N2+2)=IA(38)                                           SUBS123
      IB(K-1)=IA(38)                                            SUBS124
      IB(K1+1)=IA(38)                                           SUBS125
      IF (N22) 430,430,410                                      SUBS126
410   DO 420 I=N3,N4                                            SUBS127
      IB(K)=IC(I)                                               SUBS128
420   K=K+1                                                     SUBS129
      N3=N3+N22                                                 SUBS130
430   WRITE (IOTAPE,890) (IB(J),J=1,80)                         SUBS131
      GO TO 160                                                 SUBS132
440   IF (N1-4) 550,450,450                                     SUBS133
450   N7=N1+2                                                   SUBS134
460   N3=1                                                      SUBS135

      K1=0                                                      SUBS136
      N7=N7-2                                                   SUBS137
      IF (N7-4) 470,480,480                                     SUBS138
470   N7=N7+2                                                   SUBS139
480   DO 540 I=4,N7,2                                           SUBS140
      N2=N(I-3)+N(I-2)                                          SUBS141
      N22=N(I-1)+N(I)                                           SUBS142
      IF (N(I-3)-N(I-1)) 500,490,490                            SUBS143
490   N3=N3+N2                                                  SUBS144
      GO TO 540                                                 SUBS145
500   N4=N(I-3)                                                 SUBS146
      N(I-3)=N(I-1)                                             SUBS147
      N(I-1)=N4                                                 SUBS148
      N4=N(I)                                                   SUBS149
      N(I)=N(I-2)                                               SUBS150
      N(I-2)=N4                                                 SUBS151
      K1=K1+1                                                   SUBS152
      N4=N3+N2-1                                                SUBS153
      K=0                                                       SUBS154
      DO 510 J=N3,N4                                            SUBS155
      K=K+1                                                     SUBS156
510   IB(K)=IC(J)                                               SUBS157
      DO 520 J=1,N22                                            SUBS158
      K=N3+J-1                                                  SUBS159
      N6=N4+J                                                   SUBS160
520   IC(K)=IC(N6)                                              SUBS161
      N3=N3+N22                                                 SUBS162
      DO 530 J=1,N2                                             SUBS163
      K=N3+J-1                                                  SUBS164
530   IC(K)=IB(J)                                               SUBS165
540   CONTINUE                                                  SUBS166
      IF (K1) 550,550,460                                       SUBS167
550   WRITE (IOTAPE,890) IA(6),IA(9),IA(14),IA(9),IA(19)        SUBS168
      GO TO 820                                                 SUBS169
C        START OF SUBSTITUTION                                  SUBS170
560   IF (IW-MAXIW) 580,580,570                                 SUBS171
570   WRITE (IOTAPE,940) IW,MAXIW,(IB(I),I=1,IW)                SUBS172
      GO TO 830                                                 SUBS173
580   CALL CHECKI (IA,IB,ITEST,IOTAPE,IPTAPE,IEND,1)            SUBS174
      IF (IEND) 830,590,830                                     SUBS175
590   N3=1                                                      SUBS176
      IB(IW+1)=IA(47)                                           SUBS177
      IF (LOCK1) 600,610,600                                    SUBS178
```

```
 600   CALL SUNLK (IA,IB,IW,IOTAPE)                    SUBS179
 610   K2=1                                            SUBS180
       ILK=0                                           SUBS181
 620   N6=0                                            SUBS182
       N7=0                                            SUBS183
       DO 650 K=K2,IW                                  SUBS184
       N3=1                                            SUBS185
       DO 640 I=2,N1,2                                 SUBS186
       IF (IC(N3)-IB(K)) 640,625,640                   SUBS1861
 625   N2=N(I-1)                                       SUBS187
       N22=N(I)                                        SUBS188
       IF (IW-K-N2+1) 640,631,631                      SUBS1885
 631   K1 = K                                          SUBS189
       N4=N3+N2-1                                      SUBS190
       DO 630 J=N3,N4                                  SUBS191
       IF (IB(K1)-IC(J)) 640,630,640                   SUBS192
 630   K1=K1+1                                         SUBS193
       K1=K1-1                                         SUBS194
       N7=I                                            SUBS195
       N6=N3                                           SUBS196
       GO TO 660                                       SUBS197
 640   N3=N3+N(I-1)+N(I)                               SUBS198
 650   CONTINUE                                        SUBS199
       GO TO 800                                       SUBS200
 660   IF (K1-IW) 680,680,670                          SUBS201
 670   IW=K1                                           SUBS202
       ILK=1                                           SUBS203
       IF (IW-MAXIW) 680,680,790                       SUBS204
 680   K1=K                                            SUBS205
       N2=N(N7-1)                                      SUBS206
       N22=N(N7)                                       SUBS207
       N3=N6+N2                                        SUBS208
       N4=N3+N22-1                                     SUBS209
       N5=N22-N2                                       SUBS210
       IF (N5) 760,690,730                             SUBS211
 690   DO 700 J=N3,N4                                  SUBS212
       IB(K1)=IC(J)                                    SUBS213
 700   K1=K1+1                                         SUBS214
 710   IB(IW+1)=IA(47)                                 SUBS2141
       IF (ILK) 720,720,800                            SUBS215
 720   K2=K1                                           SUBS216
       IF (K2 - IW) 620,620,800                        SUBS217
 730   IF (IW+N5-MAXIW) 740,740,790                    SUBS218
 740   IW=IW+N5                                        SUBS219
       K2=IW                                           SUBS220
       DO 750 J=K1,IW                                  SUBS221
       K9=K2-N5                                        SUBS222
       IB(K2)=IB(K9)                                   SUBS223
 750   K2=K2-1                                         SUBS224
       GO TO 690                                       SUBS225
 760   DO 770 J=K1,IW                                  SUBS226
       K9=J-N5                                         SUBS227
 770   IB(J)=IB(K9)                                    SUBS228
       K9=IW+N5+1                                      SUBS229
       DO 780 J=K9,IW                                  SUBS230
 780   IB(J)=IA(47)                                    SUBS231
       IW=IW+N5                                        SUBS232
       IF (N22) 710,710,690                            SUBS233
```


367-112 O - 70 - 3

```
790   WRITE (IOTAPE,910) MAXIW                                           SUBS234
800   IF (LOCK2) 810,820,810                                             SUBS235
810   CALL SULOCK (IA,IB,IW,IOTAPE)                                      SUBS236
820   RETURN                                                             SUBS237
830   STOP                                                               SUBS238
840   FORMAT (132A1)                                                     SUBS240
850   FORMAT (40I2)                                                      SUBS241
860   FORMAT (1A1,1I3,2A1,1I2)                                           SUBS242
880   FORMAT (33H LIST OF REPLACEMENTS IS TOO LONG/67H MAXIMUM CHARACTERSUBS245
     1 LENGTH IS 8000, MAXIMUM NUMBER OF PHRASES IS 400/20H CURRENT VALUSUBS246
     2ES ARE ,2I6,6H STOP.)                                              SUBS247
890   FORMAT (1X,131A1)                                                  SUBS248
900   FORMAT (16H THE CHARACTER  ,1A1,48H DID NOT APPEAR 4 TIMES ON THE SUBS249
     1CARD BELOW. STOP./1X,80A1)                                         SUBS250
910   FORMAT (40H0THE LINE FOLLOWING WOULD HAVE EXCEEDED ,1I6,43H CHARACSUBS251
     1TERS IS SUBSTITUTION HAD CONTINUED.   )                            SUBS252

920   FORMAT (1X,1A1,1I3,2A1,1I2)                                        SUBS253
930   FORMAT (1X,50I2)                                                   SUBS254
940   FORMAT (19H STRING OF LENGTH =,1I6,43H IS TOO LONG FOR SUBROUTINE SUBS255
     1SUBST. LENGTH =,1I6,6H STOP./,1X,120A1)                            SUBS256
      END                                                                SUBS257-

      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 (116H0THE WORK ON THE FOLLOWING LINE WAS HALTED JUST BEFORESSUK 560
       1 THE MAXIMUM CHARACTER LINE LIMIT WAS EXCEEDED IN  SUNLK       )   SSUK 570
 280    FORMAT (69H0THE FOLLOWING LINE DID NOT CONTAIN A BALANCED SET OF SSSUK 580
       1HIFT SYMBOLS. )                                                  SSUK 590
        END                                                              SSUK 600

        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)                                                  SSLK 245
        IB(IW+2)=IA(47)                                                  SSLK 246
        IB(IW+3)=IA(47)                                                  SSLK 247
        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 (116H0THE WORK ON THE FOLLOWING LINE WAS HALTED JUST BEFORESSLK 300
       1 THE MAXIMUM CHARACTER LINE LIMIT WAS EXCEEDED IN  SULOCK      )   SSLK 310
        END                                                              SSLK 320
```

```
      SUBROUTINE CHECKI(IA,IB,ITEST,IOTAPE,IPTAPE,K,J)
      DIMENSION IA(86),IB(999)
      IF (K-1) 20,40,20
20    L=J-1
      DO 30 I=1,26
      L=L+1
      IF (IA(I)-IB(L)) 70,30,70
30    CONTINUE
      IF (K-2) 40,90,90
40    K=1
      WRITE (IOTAPE,100) (IA(I),I=1,80)
      IF (ITEST) 50,80,50
50    WRITE (IPTAPE,110) (IA(I),I=1,80)
      IF (IPTAPE-6) 80,80,60
60    END FILE IPTAPE
      GO TO 80


70    K=0
80    RETURN
90    K=1
      GO TO 80
100   FORMAT (1X,80A1)
110   FORMAT (80A1)
      END
```

```
      SUBROUTINE AMATCH(IA,ID,LENGTH,ITYPE,MATCH)                        AMAT 010
C                                                                        
C  IC() IS 1000 CHARACTERS LONG AND 40 MATCH STRINGS ARE ALLOWED.        AMAT 020
C  ID() CONTAINS THE STRING WHICH IS TO BE COMPARED WITH THE DICTIONARY AMAT 030
C         STRINGS.                                                       AMAT 040
C LENGTH IS    E LENGTH OF ID(). WHEN LENGTH IS SMALLER THAN A DICTIONARYAMAT 050
C         STRING, THERE IS NO MATCH.                                     AMAT 060
C  IA() CONTAINS THE DICTIONARY OF SPECIAL CHARACTERS NEEDED.            AMAT 070
C  IA(47) IS A BLANK AND THE ALPHABET OCCURS BETWEEN IA(1) AND IA(26)    AMAT 080
C  ITYPE IS THE ENTRY POINT, ZERO FOR READING IN THE MATCH STRINGS, ONE AMAT 090
C         FOR THE PROCESS OF MATCHING THE STRING ID() WITH THE DICTIONARYAMAT 100
C  MATCH IS ZERO IF A MATCH IS NOT FOUND, OTHERWISE MATCH CONTAINS THE  AMAT 110
C         NUMBER OF THE MATCHED STRING.                                  AMAT 120
C  IEND IS THE CHARACTER USED TO INDICATE THE END OF AN INPUT STRING.   AMAT 130
C  IGN IS THE CHARACTER INDICATING A FIELD ON THE MATCH STRING IN WHICH AMAT 140
C         THE PRESENCE OF ANY CHARACTER IS ALLOWED.                      AMAT 150
C                             INPUT FORMAT                               AMAT 160
C     THE FIRST CARD CONTAINS THE IEND AND IGN CHARACTERS IN THAT ORDER.AMAT 170
C THE FIRST NONBLANK CHARACTER IS ASSUMED TO BE IEND AND THE SECOND     AMAT 180
C NONBLANK CHARACTERS TO BE IGN. IF NEITHER CHARACTERS ARE TO BE USED,  AMAT 190
C A BLANK CARD MUST BE INSERTED.                                         AMAT 200
C     THE FOLLOWING CARDS CONTAIN A MATCH STRING PER CARD WITH EACH     AMAT 210
C STRING STARTING IN CARD COLUMN ONE. IF THE STRING IS NOT TO BE        AMAT 220
C TREATED AS BEING 80 CHARACTERS LONG, IT MUST BE TERMINATED BY AN IEND AMAT 230
C CHARACTER. ANY CARD FIELD POSITIONS HAVING AN IGN CHARACTER WILL BE   AMAT 240
C CONSIDERED TO MATCH AUTOMATICALLY.                                     AMAT 250
C     THE LAST CARD HAS STARTING IN CARD COLUMNS 1 TO 6 THE WORD FINIS  AMAT 260
C WITH THE IEND CHARACTER APPEARING IN CARD COLUMN 7 IF AN IEND WAS     AMAT 270
C SPECIFIED ON THE FIRST INPUT CARD.                                     AMAT 280
C                                                                        
      DIMENSION IA(84),IB(80),IC(1000),ID(136),N(40)                     AMAT 290
      ITAPE=5                                                            AMAT 300
      IOTAPE=6                                                           AMAT 310
      IF (ITYPE) 300,10,300                                              AMAT 320
10    IGNORE=2                                                           AMAT 330
      N1=0                                                               AMAT 340
      N3=1                                                               AMAT 350
      READ (ITAPE,9) (IB(I),I=1,80)                                      AMAT 360
      DO 20 I=1,80                                                       AMAT 370
      IF (IB(I)-IA(47)) 30,20,30                                         AMAT 380
20    CONTINUE                                                           AMAT 390
      IGNORE=1                                                           AMAT 400
      GO TO 80                                                           AMAT 410
30    IEND=IB(I)                                                         AMAT 420
      K=I+1                                                              AMAT 430
      IF (K-80) 40,40,60                                                 AMAT 440
40    DO 50 I=K,80                                                       AMAT 450
      IF (IB(I)-IA(47)) 70,50,70                                         AMAT 460
50    CONTINUE                                                           AMAT 470
60    IGNORE=0                                                           AMAT 480
      GO TO 80                                                           AMAT 490
70    IGN=IB(I)                                                          AMAT 500
80    WRITE (IOTAPE,39) (IB(K),K=1,I)                                    AMAT 510
90    READ (ITAPE,9) (IB(J),J=1,80)                                      AMAT 520
      N2=0                                                               AMAT 530
      IF (IB(1)-IA(6)) 160,100,160                                       AMAT 540
```

```
100   IF (IB(2) - IA(9)) 160,110,160
110   IF (IB(3) - IA(14)) 160,120,160
120   IF (IB(4) - IA(9))  160,130,160
130   IF (IB(5) - IA(19)) 160,140,160
140   IF (IGNORE - 1) 150,240,150
150   IF (IB(6) - IEND) 160,240,160
160   DO 180 I=1,80
      IF (IB(I) - IEND) 180,170,180
170   IF (N2) 90,90,190
180   N2=I
190   N1=N1+1
      IF (N1 - 40) 200,200,210
200   N(N1)=N2
      N4=N3+N2-1
      IF (N4-1000) 220,220,210
210   WRITE (IOTAPE,29) N4,N1
9999  STOP
220   J=1
      DO 230 I=N3,N4
      IC(I)=IB(J)
230   J=J+1
      N3=N3+N2
      WRITE (IOTAPE,19) (IB(J),J=1,80)
      GO TO 90
240   WRITE (IOTAPE,19) (IB(J),J=1,80)
1000  RETURN
300   IF (N4) 340,340,310
310   IF (1000-N4) 340,320,320
320   IF (N1) 340,340,330
330   IF (40 - N1) 340,350,350
340   WRITE (IOTAPE,49)
      GO TO 9999
350   N3=1
      DO 410 J=1,N1
      N2=N(J)
      N4=N2+N3-1
      IF (N2 - LENGTH) 360,360,410
360   K=1
      DO 390 I=N3,N4
      IF (IC(I) - ID(K)) 370,390,370
370   IF (IGNORE-1) 410,410,380
380   IF (IC(I) - IGN) 410,390,410
390   K=K+1
400   MATCH=J
      GO TO 1000
410   N3=N3+N2
      MATCH=0
      GO TO 1000
9     FORMAT (136A1)
19    FORMAT (1X,135A1)
29    FORMAT (79H0SUBROUTINE AMATCH WAS GIVEN TOO MANY STRINGS. MAXIMUM
     1SIZE IS 1000 CHARACTERS    /34H OR 40 STRINGS, CURRENT VALUES ARE
     2    ,2I6,6H STOP.)
39    FORMAT (25H0SUBROUTINE AMATCH INPUT    //1X,80A1)
49    FORMAT (62H0SUBROUTINE AMATCH WAS NOT CALLED ON TO READ IN STRINGS
     1. STOP.  )
      END
```

```
      SUBROUTINE  NPRINT(K,ITEST,IC,IBLANK,IOTAPE,IPTAPE,IEND)
C     THIS VERSION OF NPRINT USES A FORTRAN WRITE STATEMENT
C        SUBROUTINE NPRINT -NTRAN PRINT- PRINTS OUT RECORDS OF LENGTH
C     NCOUT FROM THE STRING IC -1A1 FORMAT- THE CURRENT LENGTH OF
C     CHARACTERS IN IC IS K.  IF K IS LESS THAN NCOUT NOTHING IS DONE
C     UNLESS  THE LAST RECORD IS TO BE WRITTEN INDICATED BY IEND=1
C     WHEN IEND=-1 THE ENTIRE CONTENTS OF IC IS WRITTEN OUT BUT THE
C     TAPE IS NOT ENDFILED AND EACH RECORD IS NCOUT CHARACTERS LONG
C     WHEN IEND=1 THE LAST RECORD IS FILLED WITH IBLANKS FROM K+1 TO
C     NCOUT AND AN END OF FILE IS PLACED ON IPTAPE.  ITEST IS NEGATIVE
C     FOR PRINTING, ZERO FOR PRINTING AND WRITING TAPE AND POSTIVE FOR
C     WRITING TAPE.  IOTAPE IS THE SYSTEM PRINTER. IPTAPE IS THE TAPE.
      DIMENSION IC(4100)
      NCOUT=132
10    IF (K) 20,20,80
20    IF (IEND) 70,70,40
40    IF (ITEST) 60,50,50
50    ENDFILE IPTAPE
60    WRITE (IOTAPE,19) (IC(I),I=1,NCOUT)
      WRITE (IOTAPE,9)
70    RETURN
80    IF (IEND) 90,110,90
90    IF (NCOUT*(K/NCOUT)-K) 100,110,100
100   K = NCOUT*((K/NCOUT) + 1)
110   IF (K-NCOUT) 20,160,160
160   N=N+1
      IF (ITEST) 170,170,180
170   WRITE (IOTAPE,19) (IC(I),I=1,NCOUT)
      WRITE (IOTAPE,59) N,NCOUT
180   IF (ITEST) 210,200,200
200   WRITE (IPTAPE,39)  (IC(I),I=1,NCOUT)
210   IF (K - NCOUT) 10,220,240
220   K=0
      DO 230 I=1,NCOUT
230   IC(I)=IBLANK
      GO TO 10
240   J=K
      K=0
      K1=NCOUT+1
      DO 250 I=K1,J
      K=K+1
250   IC(K)=IC(I)
      K1=K+1
      DO 260 I=K1,J
260   IC(I)=IBLANK
      GO TO 110
9     FORMAT (45H0****** THE ABOVE IS THE LAST RECORD WRITTEN    )
19    FORMAT (1X,100A1)
39    FORMAT (132A1)
59    FORMAT ( 27H0****** ABOVE IS RECORD NO  ,1I6,7H IT IS  1I6,
     1 7H LONG.    /)
      END
```

APPENDIX II

This Appendix shows how the programs in Appendix I were modified for the NBS UNIVAC 1108 in order to take full advantage of buffered tape read, buffered tape write, and labeled common. These changes should serve also as a guide for optimizing the programs when run under other systems.

```
      SUBROUTINE  NPRINT(K,ITEST,IC,IBLANK,IOTAPE,IPTAPE,IEND)          NTPR  10
C     THIS VERSION OF NPRINT IS A BUFFERED WRITE USING NTRAN AT NBS.    NTPR  15
C        SUBROUTINE NPRINT -NTRAN PRINT- PRINTS OUT RECORDS OF LENGTH   NTPR  20
C     NCOUT FROM THE STRING IC -1A1 FORMAT- THE CURRENT LENGTH OF       NTPR  30
C     CHARACTERS IN IC IS K.  IF K IS LESS THAN NCOUT NOTHING IS DONE   NTPR  40
C     UNLESS  THE LAST RECORD IS TO BE WRITTEN INDICATED BY IEND=1      NTPR  50
C     WHEN IEND=-1 THE ENTIRE CONTENTS OF IC IS WRITTEN OUT BUT THE TAPENTPR  60
C     IS NOT ENDFILED AND EACH RECORD IS NCOUT CHARACTERS LONG          NTPR  70
C     WHEN IEND=1 THE LAST RECORD IS FILLED WITH IBLANKS FROM K+1 TO    NTPR  80
C     NCOUT AND AN END OF FILE IS PLACED ON IPTAPE.  ITEST IS NEGATIVE  NTPR  90
C     FOR PRINTING, ZERO FOR PRINTING AND WRITING TAPE AND POSTIVE FOR  NTPR 100
C     WRITING TAPE.  IOTAPE IS THE SYSTEM PRINTER. IPTAPE IS THE TAPE.  NTPR 110
      DIMENSION IC(4100),IWORDS(500)                                    NTPR 120
      NCOUT = 300                                                       NTPR 130
10    IF (K) 20,20,80                                                   NTPR 140
20    IF (IEND) 70,70,30                                                NTPR 150
30    IF (L+1)  190,30,40                                               NTPR 160
40    IF (ITEST) 60,50,50                                               NTPR 170
50    CALL NTRAN (IPTAPE,9)                                             NTPR 180
60    WRITE (IOTAPE,39) (IWORDS(I),I=1,NWOUT)                           NTPR 190
      WRITE (IOTAPE,9)                                                  NTPR 200
70    RETURN                                                            NTPR 210
80    IF (IEND) 90,110,90                                               NTPR 220
90    IF (NCOUT*(K/NCOUT)-K) 100,110,100                                NTPR 230
100   K = NCOUT*((K/NCOUT) + 1)                                         NTPR 240
110   IF (K-NCOUT) 20,160,160                                           NTPR 250
160   IF (ITEST) 170,170,180                                            NTPR 260
170   WRITE (IOTAPE,19) (IC(I),I=1,NCOUT)                               NTPR 270
      N = N+1                                                           NTPR 280
      WRITE (IOTAPE,59) N,NCOUT,L                                       NTPR 290
      IF (ITEST) 210,180,180                                            NTPR 300
180   IF (L+1) 190,180,200                                              NTPR 310
190   WRITE (IOTAPE,29) L                                               NTPR 320
      CALL NTRAN (IPTAPE,22)                                            NTPR 330
      CALL NTRAN (IPTAPE,9)                                             NTPR 340
      WRITE (IOTAPE,39) (IWORDS(I),I=1,NWOUT)                           NTPR 350
      WRITE (IOTAPE,49)                                                 NTPR 360
      STOP                                                              NTPR 370
200   CALL PACK (NCOUT,NWOUT,IWORDS,IC,IBLANK)                          NTPR 380
      CALL NTRAN (IPTAPE,1,NWOUT,IWORDS,L)                              NTPR 390
210   IF (K - NCOUT) 10,220,240                                         NTPR 400
220   K=0                                                               NTPR 410
      DO 230 I=1,NCOUT                                                  NTPR 420
230   IC(I)=IBLANK                                                      NTPR 430
      GO TO 10                                                          NTPR 440
240   J=K                                                               NTPR 450
      K=0                                                               NTPR 460
      K1=NCOUT+1                                                        NTPR 470
      DO 250 I=K1,J                                                     NTPR 480
      K=K+1                                                             NTPR 490
```

```
250   IC(K)=IC(I)                                                       NTPR 500
      K1=K+1                                                            NTPR 510
      DO 260 I=K1,J                                                     NTPR 520
260   IC(I)=IBLANK                                                      NTPR 530
      GO TO 110                                                         NTPR 540
9     FORMAT (45H0****** THE ABOVE IS THE LAST RECORD WRITTEN     )    NTPR 550
19    FORMAT (1X,100A1)                                                 NTPR 560
29    FORMAT(35H0****** NTRAN WRITE ERROR, STATUS= 1I6,7H ****** ///)   NTPR 570
39    FORMAT (1X,20A6)                                                  NTPR 580
49    FORMAT(//51H ****** ABOVE RECORD NOT WRITTEN DUE TO NTRAN ERROR   )NTPR 590
59    FORMAT ( 27H0****** ABOVE IS RECORD NO   ,1I6,7H IT IS  1I6,      NTPR 600
     123H LONG. STATUS WORD IS    1I6 /)                                NTPR 610
      END                                                               NTPR 620
      SUBROUTINE PACK (ICHAR,IWOUT,IWORDS,ISTRIN,IBLANK)                PACK  10 • y
C THIS SUBROUTINE CHANGES INFORMATION STORED IN (A1) FORMAT INTO WORDS  PACK  20
C PACKED SIX CHARACTERS PER WORD ON A UNIVAC 1108.                      PACK  30
C  ICHAR IS THE NUMBER OF BCD CHARACTERS TO BE PACKED.                  PACK  40
C  IWOUT IS THE NUMBER OF WORDS CONTAINING THE PACKED INFORMATION       PACK  50
C  IWORDS(500) CONTAINS THE PACKED INFORMATION .  IF ICHAR IS NOT A     PACK  60
C MULTIPLE OF SIX THE LAST WORD IS FILLED OUT WITH BLANKS.              PACK  70
C  ISTRIN(4000) CONTAINS THE UNPACKED INFORMATION                       PACK  80
C  IBLANK CONTAINS AN UNPACKED BLANK CHARACTER                          PACK  90
      DIMENSION IWORDS(500),ISTRIN(4100),IX(6)                          PACK 100
      IW=ICHAR                                                          PACK 110
      IF (ICHAR-6*(ICHAR/6)) 10,30,10                                   PACK 120
10    IW=6*(ICHAR/6)+6                                                  PACK 130
      L=ICHAR+1                                                         PACK 140
      K=0                                                               PACK 150
      DO 20 I=L,IW                                                      PACK 160
      K=K+1                                                             PACK 170
      IX(K)=ISTRIN(I)                                                   PACK 180
20    ISTRIN(I)=IBLANK                                                  PACK 190
30    IWOUT=IW/6                                                        PACK 200
      I=0                                                               PACK 210
      DO 40 IZ=1,IWOUT                                                  PACK 220
      I=I+1                                                             PACK 230
      FLD ( 0,6,IWORDS(IZ))=FLD(0,6,ISTRIN(I))                          PACK 240
      I=I+1                                                             PACK 250
      FLD ( 6,6,IWORDS(IZ))=FLD(0,6,ISTRIN(I))                          PACK 260
      I=I+1                                                             PACK 270
      FLD (12,6,IWORDS(IZ))=FLD(0,6,ISTRIN(I))                          PACK 280
      I=I+1                                                             PACK 290
      FLD (18,6,IWORDS(IZ))=FLD(0,6,ISTRIN(I))                          PACK 300
      I=I+1                                                             PACK 310
      FLD (24,6,IWORDS(IZ))=FLD(0,6,ISTRIN(I))                          PACK 320
      I=I+1                                                             PACK 330
      FLD (30,6,IWORDS(IZ))=FLD(0,6,ISTRIN(I))                          PACK 340
40    CONTINUE                                                          PACK 350
      IF (ICHAR-6*(ICHAR/6)) 50,70,50                                   PACK 360
50    K=0                                                               PACK 370
      DO 60 I=L,IW                                                      PACK 380
      K=K+1                                                             PACK 390
60    ISTRIN(I)=IX(K)                                                   PACK 400
70    RETURN                                                            PACK 410
      END                                                               PACK 420
```

```
C                               SUBROUTINE TPNRD     (TAPE READ USING NTRAN)
C
      SUBROUTINE TPNRD
      DIMENSION IA(84),IB(136)
      COMMON /A/ ITEST,ITAPE,IOTAPE,IRTAPE,IPTAPE,ITYPE,IA
      COMMON /B/ LENTH
       COMMON /E/ IRD
      COMMON /IID/ IB
      DIMENSION IREC(100)
C     THIS PROGRAM USES NTRAN TO READ A RECORD FROM IRTAPE INTO IREC.
C     IT THEN UNPACKS THE RECORD TO A1 FORMAT PLACING IT INTO IB.
      ILEN = 100
      JEND = 1
      JE = 0
      ICHK = 0
10    IF (IRD) 20,70,100
20    IF (IRD+1) 30,10,10
30    IF (IRD+2) 80,40,80
40    JE=JE+1
      WRITE (IOTAPE,190) JE
      IF (JEND-JE) 50,50,60
50    RETURN
60    CALL NTRAN (IRTAPE,22)
70    CALL NTRAN (IRTAPE,2,ILEN,IREC,IRD)
      GO TO 10
C     READ ERROR
80    WRITE (IOTAPE,200) IRTAPE,IRD
      CALL NTRAN (IRTAPE,22)
      IERR=IERR+1
      CALL NTRAN (IRTAPE,7,1)
      IF (IERR-10) 70,90,90
90    WRITE (IOTAPE,210)
      STOP
100   IW=6*IRD
      IF (ICHK-2) 140,130,110
110   IRI=IRD+19
      DO 120 JI1=1,IRI,21
      JI2=JI1+20
120   WRITE (IOTAPE,220) (IREC(J),J=JI1,JI2)
130   WRITE (6,230) IRD,ITAPE,IOTAPE,IRTAPE,IPTAPE,IW
      CALL CLOCK
  140 J = 136
      DO 150 I=1,J
150   IB(I)=IA(47)
      DO 160 I = 1,LENTH
      J=I-((I-1)/6)*6
      IZ=(I-1)/6+1
160   FLD(0,6,IB(I))=FLD(6*(J-1),6,IREC(IZ))
      DO 170 J = 1,LENTH
      K = LENTH-J+1
      IF (IB(K)) 180,170,180
  170 IB(K) = IA(47)
      WRITE (IOTAPE,240)
      IERR=IERR+1
      IF (IERR-10) 70,70,90
180   IW=K
      CALL NTRAN (IRTAPE,2,ILEN,IREC,IRD)
      RETURN
C
190   FORMAT (13H END OF FILE ,1I4)
200   FORMAT (25H INPUT/OUTPUT ERROR UNIT ,1I3,8H STATUS ,1I3)
210   FORMAT (39H NTRAN READ ERRORS REACHED LIMIT. STOP.)
220   FORMAT (1X,21A6)
230   FORMAT (1X,13I10)

240   FORMAT (23H RECORD WAS ALL BLANKS.)
      END
```

```
C              PATCHES TO         SETLST
C
      COMMON /A/ ITEST,ITAPE,IOTAPE,IRTAPE,IPTAPE,ITYPE,IA           TYST 641   a
      COMMON /B/ LENGTH,MATCH                                        TYST 642
      COMMON /C/ K                                                   TYST 643
      COMMON /D/ K1,IEND,IBLANK                                      TYST 644
       COMMON /E/ IRD                                                TYST 645
      COMMON /IIB/  IB                                               TYST 646
      COMMON /IIC/ IC                                                TYST 647
      COMMON /IID/ ID                                                TYST 648

201   ITYPE=0                                                        TYST146    b
      IEND=0                                                         TYST1461
      IBLANK=IA(47)                                                  TYST1462
      CALL AMATCH                                                    TYST1463
      CALL SUBST                                                     TYST1464
      ITYPE=1                                                        TYST147

  200 IF (IRTAPE-5) 196,410,405                                      TYST157    c
405   CALL TPNRD                                                     TYST1571
      IF (IRD+1) 900,395,395                                         TYST1572
  410 READ (IRTAPE,9,END=900,ERR=900)(ID(I),I=1,LENGTH)              TYST158
  395 CALL AMATCH                                                    TYST159

      CALL SUBST                                                     TYST187    d

807   CALL NPRINT                                                    TYST194    e

      IEND=1                                                         TYST211    f
      CALL NPRINT                                                    TYST212


C              PATCHES TO         KWIND
C
      COMMON /A/ ITEST,ITAPE,IOTAPE,IRTAPE,IPTAPE,ITYPE,IA           KWIC002    g
      COMMON /B/ LENGTH,MATCH                                        KWIC003
      COMMON /C/ K                                                   KWIC004
      COMMON /D/ K1,IEND,IBLANK                                      KWIC005
       COMMON /E/ IRD                                                KWIC006
      COMMON /IIB/  IB                                               KWIC007
      COMMON /IIC/ IC                                                KWIC008
      COMMON /IID/ ID                                                KWIC009

      ITYPE=0                                                        KWIC0827   h
      IEND=0                                                         KWIC0828
      IBLANK=IA(47)                                                  KWIC0829
      CALL AMATCH                                                    KWIC0830
      CALL SUBST                                                     KWIC0831
      ITYPE=1                                                        KWIC0832

200   IF (IRTAPE-5) 9999,203,202                                     KWIC0850   i
202   CALL TPNRD                                                     KWIC0851
      IF (IRD+1)  900,201,201                                        KWIC0852
203   READ (IRTAPE,9,END=900,ERR=900) (ID(I),I=1,LENGTH)             KWIC0860
201   N1=N(1)                                                        KWIC0870
      N2=N(2)                                                        KWIC0871
      CALL AMATCH                                                    KWIC0880

      CALL SUBST                                                     KWIC2032   j
```

```
      CALL NPRINT                                                       KWIC2050

      IEND=1                                                            KWIC2200
      CALL NPRINT                                                       KWIC2210


C              PATCHES TO        SUBST
C
      SUBROUTINE SUBST                                                  SUBST 10
      COMMON /A/ ITEST,ITAPE,IOTAPE,IRTAPE,IPTAPE,ITYPE,IA              SUBS  11
      COMMON /C/ IW                                                     SUBS  12
      COMMON /IIC/ IB                                                   SUBS  13

      READ (ITAPE,850) ITES,IRTAP,IPTAP                                 SUBS 47
C                                                                       SUBS 48
C                                                                       SUBS 49
C                                                                       SUBS 53
C                                                                       SUBS 54
      WRITE (IOTAPE,930) ITES,IRTAP,IPTAP                               SUBS 58

C                                                                       SUBS174
580   CONTINUE                                                          SUBS175

600   CALL SUNLK                                                        SUBS179

810   CALL SULOCK                                                       SUBS236


C              PATCHES TO        SUNLK
C
      SUBROUTINE SUNLK                                                  SSUK  10
      COMMON /A/ ITEST,ITAPE,IOTAPE,IRTAPE,IPTAPE,ITYPE,IA              SSUK  11
      COMMON /C/ IW                                                     SSUK  12
      COMMON /IIC/  IB                                                  SSUK  13


C              PATCHES TO        SULOCK
C
      SUBROUTINE SULOCK                                                 SSLK  10
      COMMON /A/ ITEST,ITAPE,IOTAPE,IRTAPE,IPTAPE,ITYPE,IA              SSLK  11
      COMMON /C/ IW                                                     SSLK  12
      COMMON /IIC/  IB                                                  SSLK  13


C              PATCHES TO        AMATCH
C
      SUBROUTINE AMATCH                                                 AMAT  10
C
      COMMON /A/ ITEST,ITAPE,IOTAPE,IRTAPE,IPTAPE,ITYPE,IA              AMAT 291
      COMMON /B/ LENGTH,MATCH                                           AMAT 292
      COMMON /IID/ ID                                                   AMAT 293


C              PATCHES TO        NPRINT WITH FORTRAN WRITE
C
      SUBROUTINE NPRINT                                                 NPRI  10
      COMMON /A/ ITEST,ITAPE,IOTAPE,IRTAPE,IPTAPE,ITYPE,IA              NPRI  11
      COMMON /D/ K,IEND,IBLANK                                          NPRI  12
      COMMON /F/ NCOUT,NWOUT,IWORDS                                     NPRI  13
      COMMON /IIB/ IC                                                   NPRI  14
```

PATCHES TO NPRINT WITH NTRAN

```
      SUBROUTINE NPRINT                                          NTPR  10
      COMMON /A/ ITEST,ITAPE,IOTAPE,IRTAPE,IPTAPE,ITYPE,IA       NTPR  11
      COMMON /D/ K,IEND,IBLANK                                   NTPR  12   w
      COMMON /F/ NCOUT,NWOUT,IWORDS                              NTPR  13
      COMMON /IIB/ IC                                            NTPR  14

)0    CALL PACK                                                  NTPR 380   x
```

PATCHES TO PACK

```
      SUBROUTINE PACK                                            PACK  10
      COMMON /D/ K1,IEND,IBLANK                                  PACK  11
      COMMON /F/ ICHAR,IWOUT,IWORDS                              PACK  12   y
      COMMON /IIB/ ISTRIN                                        PACK  13
```


U. S. GOVERNMENT PRINTING OFFICE : 1970 O - 367-112

NBS TECHNICAL PUBLICATIONS

PERIODICALS

JOURNAL OF RESEARCH reports National Bureau of Standards research and development in physics, mathematics, chemistry, and engineering. 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.

Published in three 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; foreign, $11.75*.

• 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; foreign, $6.25*.

• Engineering and Instrumentation

Reporting results of interest chiefly to the engineer and the applied scientist. This section includes many of the new developments in instrumentation resulting from the Bureau's work in physical measurement, data processing, and development of test methods. It will also cover some of the work in acoustics, applied mechanics, building research, and cryogenic engineering. Issued quarterly. Annual subscription: Domestic, $5.00; foreign, $6.25*.

TECHNICAL NEWS BULLETIN

The best single source of information concerning the Bureau's 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.* Annual subscription: Domestic, $3.00; foreign, $4.00*.

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 quantitive 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.

CLEARINGHOUSE

The Clearinghouse for Federal Scientific and Technical Information, operated by NBS, supplies unclassified information related to Government-generated science and technology in defense, space, atomic energy, and other national programs. For further information on Clearinghouse services, write:

Clearinghouse
U.S. Department of Commerce
Springfield, Virginia 22151

* Difference in price is due to extra cost of foreign mailing.

Order NBS publications from: Superintendent of Documents
Government Printing Office
Washington, D.C. 20402

U.S. DEPARTMENT OF COMMERCE
WASHINGTON, D.C. 20230

OFFICIAL BUSINESS

POSTAGE AND FEES PAID
U.S. DEPARTMENT OF COMMERCE