NBS TECHNICAL NOTE

470

Edpac:
Utility Programs for Computer-Assisted Editing, Copy-Production, and Data Retrieval

U.S. DEPARTMENT OF COMMERCE
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 three broad program areas and provides central national services in a fourth. 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, and the Center for Radiation Research.

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 the 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 Standard Reference Data and a group of divisions organized by the following areas of science and engineering:

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

THE INSTITUTE FOR MATERIALS RESEARCH conducts materials research leading to methods, standards of measurement, and data needed by industry, commerce, educational institutions, and government. The Institute also provides advisory and research services to other government agencies. The Institute consists of an Office of Standard Reference Materials and a group of divisions organized by the following areas of materials research:

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

THE INSTITUTE FOR APPLIED TECHNOLOGY provides for the creation of appropriate opportunities for the use and application of technology within the Federal Government and within the civilian sector of American industry. The primary functions of the Institute may be broadly classified as programs relating to technological measurements and standards and techniques for the transfer of technology. The Institute consists of a Clearinghouse for Scientific and Technical Information,[3] a Center for Computer Sciences and Technology, and a group of technical divisions and offices organized by the following fields of technology:

Building Research—Electronic Instrumentation — Technical Analysis — Product Evaluation—Invention and Innovation— Weights and Measures — Engineering Standards—Vehicle Systems Research.

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 for Radiation Research consists of the following divisions:

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

---

[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
C. R. Smith, Secretary
NATIONAL BUREAU OF STANDARDS • A. V. Astin, Director

TECHNICAL NOTE 470

ISSUED JANUARY 1969

Edpac: Utility Programs for Computer-Assisted Editing, Copy-Production, and Data Retrieval

Carla G. Messina
and Joseph Hilsenrath

Office of Standard Reference Data
Institute for Basic Standards
National Bureau of Standards
Washington, D.C. 20234

For sale by the Superintendent of Documents, U.S. Government Printing Office, Washington, D.C., 20402
Price 75 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 on a phototypesetting machine at the Government Printing Office from a magnetic tape produced at NBS.

TABLE OF CONTENTS

APPENDIX

ABSTRACT

A description and listings are given of EDPAC, a package of five related utility computer programs: SCRAMBLE, SUBSTITUTE, SEARCH, BLOCKSEARCH, JUSTIFY, and their subroutines. These programs perform transformations on alphanumeric data. The programs have been written in FORTRAN, with care taken to make them as system and machine-independent as possible, permitting their use on many different computers.

SCRAMBLE scans an input file for specified characters, which it replaces by different characters. SUBSTITUTE similarly replaces strings of characters by other strings. SEARCH and BLOCKSEARCH scan for the occurrence of certain strings and list the lines or blocks, respectively, in which they occur. JUSTIFY produces text, for printing on a card-controlled typewriter or on an extended character printer, which has been left and right-justified between specified margins.

Some applications of the EDPAC programs are discussed. Emphasis is placed on computer-assisted text preparation.

Key words: alphanumeric data files, computer-assisted text preparation, data retrieval, FORTRAN programs, free field text files, mechanized text editing

1. Introduction

An important ingredient, even a necessity, for an effective solution to problems of providing critical correlations of standard reference data is the application of computers to as many facets of the problem as possible.

Computers have been applied to a wide variety of computational jobs, to many bibliographic problems, and to automatic printing of data tapes. In each of these areas, however, the existing programs require their own specialized input formats. Most of the time the form of these instructions or data formats are dictated by considerations other than those of simplicity of operation or of universal application. Indeed, the two objectives often seem contradictory.

Such attempts as we have seen in formatting data for computer input have suffered from one or more serious drawbacks. The most serious of these has been the inability of even comprehensive systems to cope with the general case. It is our view that the more profitable line of attack on the problem of compatibility and interchangeability of data cards or data tapes is to provide a series of utility programs which will transform, translate, transpose, and transcribe information from one format to another.

The feasibility of handling free-field input of numerical data was demonstrated a number of years ago in a general-purpose computer program developed at NBS.* In that program, a versatile subroutine for scanning a card was written in machine language. Now the facility for reading free-field numerical data is provided in the compilers for FORTRAN IV. The extension of the free-field capability to the letters and characters employed in English language text poses a variety of problems which cannot be solved in a single scanning package. Here it seems more natural to provide a facility for translation, and transformation of characters or character strings.

To put the matter bluntly, we deem it less profitable to promulgate rigid rules on punching formats for data or text, and more profitable to devise flexible but easily used utility programs for handling mixed input or for recasting the input when other means fail.

This report describes five utility programs: SCRAMBLE, SUBSTITUTION, SEARCH, BLOCKSEARCH, and JUSTIFY. The five perform non-trivial transformations on alphanumeric data punched on cards or stored as records on a magnetic tape.

The programs have been written in a limited subset of FORTRAN IV. Particular care was taken to make them machine independent with respect to internal bit configuration and system independent with respect to input and output, and to minimize the modifications required in conversion to other FORTRAN dialects. Although the programs are all independent and perform very specific tasks, they can be used sequentially to carry out a series of operations in which the results of one code are used as input for the next. Furthermore, under a suitable operating system, it is possible to carry out a series of operations in a single run.

---

*Hilsenrath, J., Ziegler, G.G., Messina, C.G., Walsh, P.J., Herbold, R.J., OMNITAB: A Computer Program for Statistical and Numerical Analysis, National Bureau of Standards Handbook 101, Superintendent of Documents, U. S. Government Printing Office, Washington, D. C. 20402 (March 4, 1966).

An important application of such utility programs arises in the preparation of manuscripts and reports, especially where it is important to produce right-hand justified copy (flush right-hand margins). Furthermore, the programs allow for a variety of precedence characters (used for shifting from lower to upper case of a card-controlled typewriter or with an extended character printer on a computer). The combination provides reproduction copy suitable for publication. Where graphic arts quality is needed, the programs can be used to insert instructions for font changes on a photocomposing machine.

When text is to be prepared for photoreproduction on a card-controlled typewriter, it is useful to provide simple instructions for such things as: case shift, tab stops, and card eject with and without a line-feed. These are achieved by assigning a symbol for each function. As choice of symbols depends somewhat on the text, it is important for the system to allow for their specification in a simple manner with a control card.

Preface to the Second Edition

That the stock of the first printing of this Handbook should have been exhausted almost coincident with the appearance
of the reviews of it, points up the keen interest in user-oriented
systems for problem solving on modern computers. Equally gratifying is the recognition by the designers of more conventional computer languages of the need to free programmers from many of the tedious, annoying and error-generating
rules and restrictions of early versions of FORTRAN - restrictions which are largely absent in OMNITAB.

Preface to the Second Edition

That the stock of the first printing of this Handbook should have been exhausted almost coincident with the appearance of the reviews of it, points up the keen interest in user-oriented systems for problem solving on modern computers. Equally gratifying is the recognition by the designers of more conventional computer languages of the need to free programmers from many of the tedious, annoying, and error-generating rules and restrictions of early versions of FORTRAN - restrictions which are largely absent in OMNITAB.

This edition differs from the first in the following ways a number of typographical errors have

Figure 1a. The upper portion shows the condition of lines of text after editorial changes were made. Each line is punched on a separate card. The lower figure shows the same text after being processed by one of the Edpac programs. The copy was produced on a card controlled typewriter.

2. Characteristics of the Program SCRAMBLE

Numerous situations arise in data manipulation where it is necessary to replace one character by another. A common example is the conversion of data from single precision to double precision format where every E must be changed to a D. An example at the other end of the application spectrum is the encoding of straight text via a substitution cipher. The SCRAMBLE program handles both of these problems with equal facility. The substitution is achieved by having two sets of symbols punched in one-to-one correspondence in corresponding fields of the first two control cards. In this way, the symbols on the second card always replace the symbols in corresponding positions on the first. It is possible, for example, to replace all vowels by a single letter or by a special symbol. If the vowels are replaced by a blank and words are otherwise separated, one has a useful tool for computerized exercises in linguistics. The code will perform as many as 80 single character substitutions in a single pass through the machine.

2.1 The Control Cards for SCRAMBLE.

The first control card contains those characters for which substitutions are to be made in accordance with corresponding instructions punched on the second card.

The second card contains those characters to be substituted for the characters in the corresponding fields of the first control card.

The third card contains four switches in FORMAT (3I2,1I4).

a. Switch one should be a 1 (one) when a special output unit is to be written and a zero if not. The special output unit is specified by switch number 3 (see paragraph c. below).

b. The second switch is the unit from which to read the input file. (If the switch is zero, the program sets the input to unit 5 for the card reader.) The first three control cards are always read from unit 5 but the choice of a unit for the input file is left up to the user.

c. The third switch is the unit, separate from the printer, on which to write the output files. When this switch is zero the unit is set to 3 (the card punch).

d. The fourth switch contains the width (number of characters) of the text of the input file. SCRAMBLE assumes a width of 80 if "width" is less than one or greater to 132.

The program will come to a normal stop if it encounters an input record containing a duplicate of the first 26 columns of the first control card, otherwise it will read input records until it runs out of records to read. The program will terminate at once if the first two control cards are identical in all 80 positions.

```
 XY
 YX
 0 0 0
360     Z(I) =BLK
370     DO 700 I=1,51
375     Z6=I
380      Z7=Y1-(Z6-1.)*Y3
        X8=Z7-Y4
        X7=Z7+Y4
        M1=(51-I)*2
420      IF (Y(M)-X7) 450,430,430
430      M=M+1
        GO TO 420
450      IF (Y(M)-X8) 690,520,520
520      DO 560 I3=1,101
        Z3=I3
        Z4=X1+(Z3-1.)*X3
        Z5=Z4+X4
550      IF (X(M)-Z5) 590,590,560
560     CONTINUE
```

```
360     Z(I) =BLK
370     DO 700 I=1,51
375     Z6=I
380      Z7=X1-(Z6-1.)*X3
        Y8=Z7-X4
        Y7=Z7+X4
        M1=(51-I)*2
420      IF (X(M)-Y7) 450,430,430
430      M=M+1
        GO TO 420
450      IF (X(M)-Y8) 690,520,520
520      DO 560 I3=1,101
        Z3=I3
        Z4=Y1+(Z3-1.)*Y3
        Z5=Z4+Y4
550      IF (Y(M)-Z5) 590,590,560
560     CONTINUE
```

Figure 2a. An application of SCRAMBLE to modification of a computer program wherein every X was replaced by a Y and vice versa. The old program is on the left, the transformed version is on the right. Note the simplicity of the three control cards on the left. The zero designates no special card or tape output. On the computer used at NBS, a blank is read as a zero.

```
+()=')<
<[]>:\)
 1 5 0
000000 IDENTIFICATION DIVISION.
000100 PROGRAM-ID.             ALPHABETIC EMPLOYEE LISTING.
000200 AUTHOR. LAZAR.
000300 REMARKS. SEE PAGE 324 OF SYSTEM BOOK.
000400 DATE-COMPILED.
000500 ENVIRONMENT DIVISION.
000600 CONFIGURATION ...
000700 SOURC...                  PICTURE X[100].
...           ...-LEVEL          PICTURE XXX.
               03  FILLER        PICTURE X[335].
003800 FD   PRINT-FILE
003900      LABEL RECORDS ARE OMITTED
004000      RECORD CONTAINS 132 CHARACTERS
004100      DATA RECORD IS PRINT-REC.
004200 01   PRINT-REC    PICTURE X[132].
004300 WORKING-STORAGE SECTION.
004400 01   PP-DT    PICTURE 9[6].
004500 01   PAGE-CT PICTURE 9[6] VALUE ZEROES.
004600      01  PERSONMAST  PICTURE X[78].
004700      03  ZLAB  PICTURE X[6]  VALUE :1HDR X:.
004800      03  FILLER  PICTURE X[18]  VALUE SPACES.
004900      03  FILLER  PICTURE X[24]  VALUE :MASTER PERSONNEL RECORD :.
```

Figure 2b. In this application to a COBOL program, the six symbols + ( ) = ' >< are changed to < [ ] > : \ ) respectively.

```
PAGE NUMBER:    41

ABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789
ZABCDEFGHIJKLMNOPQRSTUVWXY9012345678
 1 5 8
3.  'S'GDNQDSHBZK 'R'NKHC 'R'SZSD 'O'GXRHBR'O')                          C
'G'. 'K'. 'O'DSDQRNM)                                                    B

 )
      'Q'DRDZQBG NE BTQQDMS HMSDQDRS HM RNKHC RSZSD OGXRHBR HMBKTCDR )
CDUDKNOLDMS ZMC ZOOKHBZSHNM NE MNMKHMDZQ SQZMRONQS SGDNQX, DWSDMRHNMR NE)
SGD DEEDRSHUD EHDKC SGDNQHDR NE L7FMDSHRL, ZMC HMUDRSHFZSHNMR NE RDUDQZK)
ZRODBSR NE ROHM QDKZBSHNM, OZQSHBTKZQKX MDZQ SQZMRHSHNM SDLODQZSTQDR.)
 )
4.  'T'MHSZQX 'L'ZSQHW 'E'HDKC 'S'GDNQX'O')                              C
'D'. 'F'. 'I'NGMRNM' 'I'Q.)                                              B
 )
      'Q'DRDZQBG HR TMCDQVZX SN CDSDQLHMD HE SGHR SGDNQX'4'R MNMKHMDZQ)
DPTZSHNM GZR SGD DKDLDMSZQX OZQSHBKD RODBSQZ, HM OZQSHBTKZQ, SGD )
DKDBSQNM.  'B'TQQDMS RSZSTR NE SGD VNQJ HR SGD EHMCHMF NE Z BNMSHMTTL)
NE '44'FQDD OZQSHBKD'44' RNKTSHNMR, ZLNMF VGHBG ZQD SGD MDTSQHMNR. )
'S'GD QQNAKDL ZS QGDRDMS HR SN RDKDBS EQNL SGHR BNMSHMTTL SGD)
RNKTSHNMR VGHBG ZQD RSZAKD TMCDQ SGD HMSDQZBSHNM NE SGD OZQSHBKD VHSG HSR)
DMUHQNMLDMS. 'S'GD QQHLD DEENQS HMUNKUDC HR SQXHMF SN CN SGD MTLDQHBZK)
ZMZKXRHR NM SGD BNLQTSDQ BNMRHRSZMSKX RN SGZS SGD MTLDQHBZK BGDBJR)
ENQ DHFDMUZKTLRSZQX BNMTSNMR ZQD BNQQDHS. 'Z'MXNMD HMSDQDRSDC HM SGHR)
```

```
PAGE NUMBER:    43

ABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789
BCDEFGHIJKLMNOPQRSTUVWXYZA1234567890
 U 8
4.  'T'HEORETICAL 'S'OLID 'S'TATE 'P'HYSICS'1')                          D
'R'. 'L'. 'P'ETERSON)                                                    C

 )
      'R'ESEARCH OF CURRENT INTEREST IN SOLID STATE PHYSICS INCLUDES )
DEVELOPMENT AND APPLICATION OF NONLINEAR TRANSPORT THEORY, EXTENSIONS OF)
THE EFFECTIVE FIELD THEORIES OF MAGNETISM, AND INVESTIGATIONS OF SEVERAL)
ASPECTS OF SPIN RELAXATION, PARTICULARLY NEAR TRANSITION TEMPERATURES.)
 )
5.  'U'NITARY 'M'ATRIX 'F'IELD 'T'HEORY'1')                              D
'E'. 'G'. 'J'OHNSON' 'J'R.)                                              C
 )
      'R'ESEARCH IS UNDERWAY TO DETERMINE IF THIS THEORY'S'S NONLINEAR)
EQUATION HAS THE ELEMENTARY PARTICLE SPECTRA, IN PARTICULAR, THE )
ELECTRON.  'C'URRENT STATUS OF THE WORK IS THE FINDING OF A CONTINUUM)
OF '55'FREE PARTICLE'55' SOLUTIONS, AMONG WHICH ARE THE NEUTRINOS. )
'T'HE PROBLEM AT PRESENT IS TO SELECT FROM THIS CONTINUUM THE)
SOLUTIONS WHICH ARE STABLE UNDER THE INTERACTION OF THE PARTICLE WITH ITS
ENVIRONMENT.  'T'HE PRIME EFFORT INVOLVED IS TRYING TO DO THE NUMERICAL)
```

Figure 2c. The top portion is the result of a linear substitution of characters. The bottom portion results from a restoration of the original characters. The figures 1, 5, and 8 on the third control card set input to read from the card reader (unit 5) and output to be placed on unit 8. On the lower portion, no cards are punched and the input is from unit 8.

3. Characteristics of the Program SUBSTITUTE

The program SUBSTITUTE is a more versatile version of SCRAMBLE and is therefore more complex. It has a provision for replacing any character string by any other character string regardless of where it occurs in the text.

Among the diverse jobs this program can do are the following:

a. Convert text punched on cards in BCD format (all capital letters) to upper and lower case, such as initial capitalization of the start of each sentence, or names and initials of authors.

b. Replace any arbitrary set of symbols by corresponding instructions for a phototypesetting machine.

c. Recognize typesetting instructions in a text and either delete them or replace them with other codes.

d. Anglicize text written by Americans, and the reverse.

e. Replace journal abbreviations by their five letter CODEN designations or vice versa, or by the full title.

f. Replace citation numbers in the body of a paper by new ones resulting from insertion of new references.

g. Insert complex mathematical expressions when they occur frequently in a text, thereby avoiding needless retyping and subsequent proofreading.

h. Insert typesetting instructions in place of code words for special symbols not available on the input device but available on a phototypesetting device.

i. Screen for and correct automatically inconsistent use of abbreviations or symbols.

j. Change variable names in a computer program to avoid conflicts when incorporating other programmers' work or to recover from the incompatability of various FORTRAN dialects.

SUBSTITUTE is presented in two forms: as a stand-alone main program, and as a subroutine suitable for incorporation into another main program.

3.1 The Control Cards for SUBSTITUTE

The first 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 second card contains three switches in FORMAT (3I2). They serve exactly the same purpose as the first three switches of the third control card for the SCRAMBLE program.

The third and fourth control cards in FORMAT (A1, I3, 2A1, I2), respectively specify the format of the input and output records. The five items on each card perform the following tasks:

a. The first tells which symbol is used to designate a continuation when the line is longer than the designated record length.
b. The second item defines the length (in character) of a record.
c. The next two items designate the characters used for case-shift lock and case-shift unlock. Their use is required only under circumstances described below.
d. The fifth item instructs the program to insert on input and delete on output, the case-shift symbols designated in item c. If this number is set to zero, the option is bypassed, in which case the third and fourth items discussed in c 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 card. See Figure 4a for applications of this feature of program.

Immediately following the fourth control card 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.

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 3d 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 ccl. 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 this program 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.

Since one often replaces a short string by a larger one, some provision must be made for spilling text over to the next record or line. Care is taken not to split words in the process.

3.2 The Subroutine SUBSTITUTE

The substitution capability has been incorporated into a subroutine whose name is SUBST. It can be called from a main program by using the words CALL SUBST (IB, IW, ITYPE). It must be called once to read in the control cards and the substitution table for which purpose ITYPE is set to zero and IB, and IW are ignored. It must be called again for each line to be processed with ITYPE equal to 1. At this time the variable IB denotes a singly dimensioned array on which the program is to operate. IW is the number of characters in the vector IB.

On return from this subroutine, IB contains the altered line which the main program handles in whatever manner is desired. IW contains the length of the altered line. On the first call of SUBST (when ITYPE is zero) the control cards are identical to those described for the program SUBSTITUTE.

In the interest of clarity the control cards for the subroutine SUBST have been kept identical with those for program SUBSTITUTE even though certain parameters required for the program are not needed for the subroutine. They may be left blank without disturbing the operation of the subroutine on those machines that equate a blank to a zero. Otherwise all integer switches that are not needed must be set to zero.

```
='H'. 'H'. 'H'ENKEL,   '55E'QUATION OF 'S'TATE AND THE 'T'HERMAL 'D'EPENDE
='H. 'H. 'HENKEL,   '5'5'EQUATION OF 'STATE AND THE 'THERMAL 'DEPENDENCE
'H'. 'H'. 'H'ENKEL,   '55E'QUATION OF 'S'TATE AND 2'3' 'D'EPENDENCE OF 2
='E'LASTIC 'C'OEFFICIENTS OF 'C'RYSTALLINE 'A'RGON,'55'   'J'. 'C'HEM.
='ELASTIC 'COEFFICIENTS OF 'CRYSTALLINE 'ARGON,'5'5   'J. 'CHEM. 'PHYS
'E'LTIC 'C'OEFFICIENTS OF 'C'RYSTALLINE ALPHARGON,'55'   'J'OURNAL OF
```

Figure 3a. A printout of a problem to test the "lock" and "unlock" features of SUBSTITUTE. The first line shows what was read from the input unit. The second line shows the transformation required in order to recognize strings of characters such as THERMAL. The third line shows the result after substitution and restoration of the shift and lock mode. The final transformation (the 3rd and 6th lines) were achieved via the following substitution table.

```
/'J. 'CHEM/   /'JOURNAL OF 'CHEMISTRY/
/'A/  /ALPHA/
*THE *  *2*
'THERMAL' '3'
:LAST:  :LT:
:=:   ::
```

The last line of the substitute table replaces the = sign by a null string.

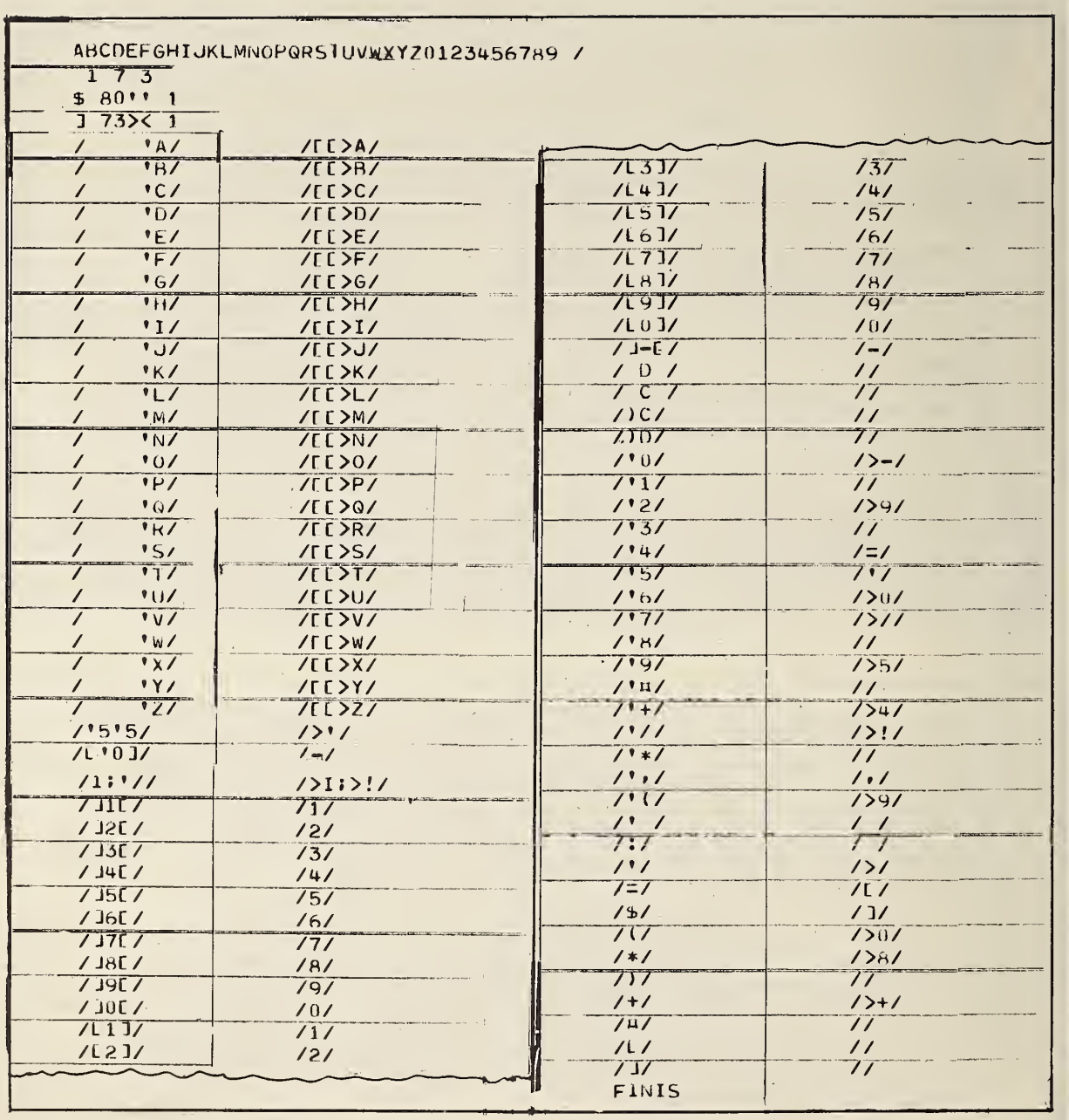
```
POSITIVE AND NEGATIVE IONS ARE PRODUCED IN THIS FASHION DEPENDING
UPON CERTAIN SURFACE AND FREE ATOM PROPERTIES. 'E'XPERIMENTS HAVE
BEEN CARRIED OUT WITH THE SURFACE IONIZATION OF FOREIGN ATOMS
IMPINGING UPON A TRANSITION METAL SURFACE '('I.E.' ALKALI ATOMS
ON TUNGSTEN( AS WELL AS THE SELF'0'] SURFACE IONIZATION OF
SUBSTRATE ATOMS '('I.E.' TUNGSTEN' NIOBIUM' AND RHENIUM(.  'T'HE
APPLICABILITY OF THE EQUILIBRIUM 'S'AHA('0'] 'L'ANGMUIR EQUATION
IS BEING STUDIED FOR A WIDE SPECTRUM OF MATERIALS.
    )
='C';/'HEMICAL 'R'EACTIONS IN THE 'C'RYOGENIC 'R'EGION'1';'/'
='M'. 'D'. 'S'CHEER AND 'R'. 'K'LEIN
    'P'ROCESSES OCCURRING WHEN GAS GENERATED FREE RADICALS REACT
WITH SOLIDS BELOW 100:'K' ARE BEING STUDIED.  'R'EACTIONS OF 'H',
'D', 'T', AND 'O' WITH CONDENSED OLEFINS PERMIT DETAILED
OBSERVATIONS ON DIFFUSION IN THE SOLID' PRECISE RELATIVE RATES OF
COMPETITIVE REACTIONS' QUANTUM MECHANICAL TUNNELING' THE NATURE
OF DISPROPORTIONATION AND COMBINATION REACTIONS' AND THE EFFECT
OF STRUCTURE ON REACTIVITY.
```

Figure 3b. This text was punched on cards for a card-controlled typewriter. The first and subsequent odd apostrophies shift and lock the carriage while the second and succeeding even ones unshift the carriage (restores to lower case). The transformation of this text stream to produce the results in Figure 3c was achieved via the SUBSTITUTE program and the control cards in Figure 3d.

```
POSITIVE AND NEGATIVE IONS ARE PRODUCED IN THIS FASHION DEPENDING
UPON CERTAIN SURFACE AND FREE ATOM PROPERTIES. >E<XPERIMENTS HAVE
BEEN CARRIED OUT WITH THE SURFACE IONIZATION OF FOREIGN ATOMS
IMPINGING UPON A TRANSITION METAL SURFACE >9<I.E.' ALKALI ATOMS
ON TUNGSTEN>0< AS WELL AS THE SELF- SURFACE IONIZATION OF
SUBSTRATE ATOMS >9<I.E.' TUNGSTEN' NIOBIUM' AND RHENIUM>0<.  >T<HE
APPLICABILITY OF THE EQUILIBRIUM >S<AHA- >L<ANGMUIR EQUATION
IS BEING STUDIED FOR A WIDE SPECTRUM OF MATERIALS.

[>C;!<HEMICAL >R<EACTIONS IN THE >C<RYOGENIC >R<EGION;>!<
[>M<. >D<. >S<CHEER AND >R<. >K<LEIN
[[>P<ROCESSES OCCURRING WHEN GAS GENERATED FREE RADICALS REACT
WITH SOLIDS BELOW 100 >K< ARE BEING STUDIED.  >R<EACTIONS OF >H<,
>D<, >T<, AND >O< WITH CONDENSED OLEFINS PERMIT DETAILED
OBSERVATIONS ON DIFFUSION IN THE SOLID' PRECISE RELATIVE RATES OF
COMPETITIVE REACTIONS' QUANTUM MECHANICAL TUNNELING' THE NATURE
OF DISPROPORTIONATION AND COMBINATION REACTIONS' AND THE EFFECT
OF STRUCTURE ON REACTIVITY.
```

Figure 3c. This text stream was produced by SUBSTITUTE from the text shown above. The control cards to achieve this transformation are shown in the next figure. The ATS System uses two tabs to achieve an indented line at the start of each paragraph. The tabs are actuated by a square bracket ([). SUBSTITUTE must therefore replace the four blanks by two square brackets. This is achieved in this problem by 26 control cards which change each capital letter of the alphabet that is preceded by four blanks to that same letter preceded by two tab symbols ([[). One control card would have been sufficient here if we were certain that the text did not contain four or more consecutive blanks in the interior of a paragraph.



```
ABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789 /
 1 7 3
$ 80'' 1
] 73>< 1
/     'A/    /[[>A/
/     'B/    /[[>B/
/     'C/    /[[>C/
/     'D/    /[[>D/
/     'E/    /[[>E/
/     'F/    /[[>F/
/     'G/    /[[>G/
/     'H/    /[[>H/
/     'I/    /[[>I/
/     'J/    /[[>J/
/     'K/    /[[>K/
/     'L/    /[[>L/
/     'M/    /[[>M/
/     'N/    /[[>N/
/     'O/    /[[>O/
/     'P/    /[[>P/
/     'Q/    /[[>Q/
/     'R/    /[[>R/
/     'S/    /[[>S/
/     'T/    /[[>T/
/     'U/    /[[>U/
/     'V/    /[[>V/
/     'W/    /[[>W/
/     'X/    /[[>X/
/     'Y/    /[[>Y/
/     'Z/    /[[>Z/
/'5'5/       />'/
/[ '0 ]/     /-/
/1;'//       />I;>!/
/ J1[ /      /1/
/ J2[ /      /2/
/ J3[ /      /3/
/ J4[ /      /4/
/ J5[ /      /5/
/ J6[ /      /6/
/ J7[ /      /7/
/ J8[ /      /8/
/ J9[ /      /9/
/ J0[ /      /0/
/[ 1 ]/      /1/
/[ 2 ]/      /2/
/[ 3 ]/      /3/
/[ 4 ]/      /4/
/[ 5 ]/      /5/
/[ 6 ]/      /6/
/[ 7 ]/      /7/
/[ 8 ]/      /8/
/[ 9 ]/      /9/
/[ 0 ]/      /0/
/ J-[ /      /-/
/  D  /      //
/  C  /      //
/)C/         //
/)D/         //
/'0/         />-/
/'1/         //
/'2/         />9/
/'3/         //
/'4/         /=/
/'5/         /'/
/'6/         />0/
/'7/         />//
/'8/         //
/'9/         />5/
/'[/         //
/'+/         />4/
/'//         />!/
/'*/         //
/',/         /,/
/'(/         />9/
/' /         / /
/:/          / /
/'/          />/
/=/          /[/
/$/          /]/
/(/          />0/
/*/          />8/
/)/          //
/+/          />+/
/[/          //
/L/          //
/]/          //
FINIS
```

Figure 3d. These are the control cards and substitution table required to transform the text in the previous figures for loading on the ATS System. The solidus (/) at the end of the first control card defines the string delimiter to be used in the output. On the next card; the 1 signals the use of a special output unit; the 7 tells the unit from which to read the input text; and the 3 indicates the special output unit. On the third control card: the $ is the continuation symbol used on the input records which contain 80 characters each; next follow the shift-lock and the shift-unlock symbols; and the 1, switches on the "unlock" feature of the program. On the fourth control card the square bracket (]) is the continuation symbol used to extend to the next card a record which is longer than the 73 positions designated; next follow the ATS shift-lock and shift-unlock symbols; and the 1 restores the "lock" feature required by the ATS System.

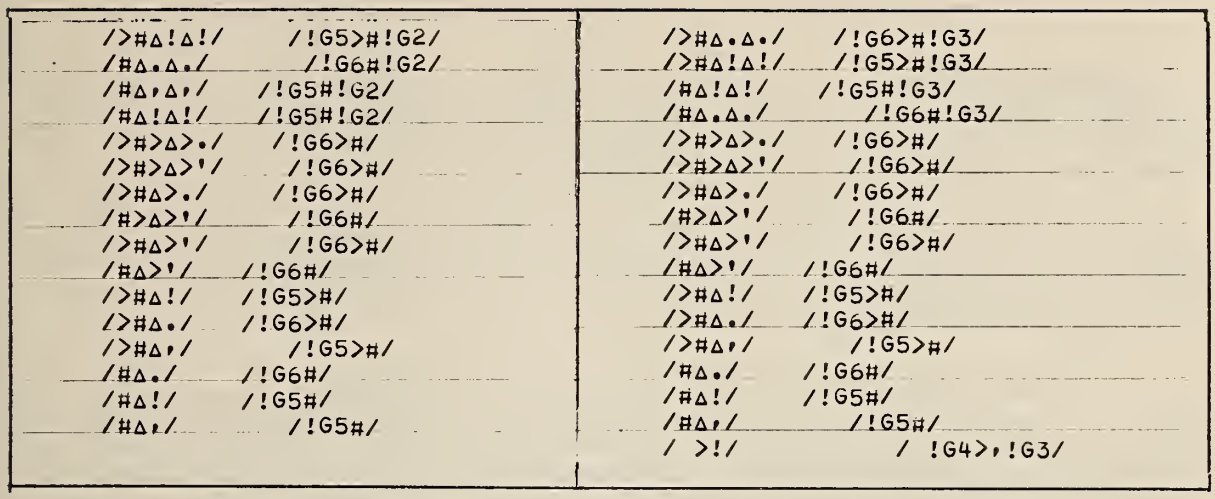


Figure 3e. A partial listing of a substitution table which tranforms the character stream produced on the ATS System to the character stream required by another program for processing the text for a photo-typesetting machine. Although the first string must start in cc 1, the position of the string to be substituted is not fixed on the card.

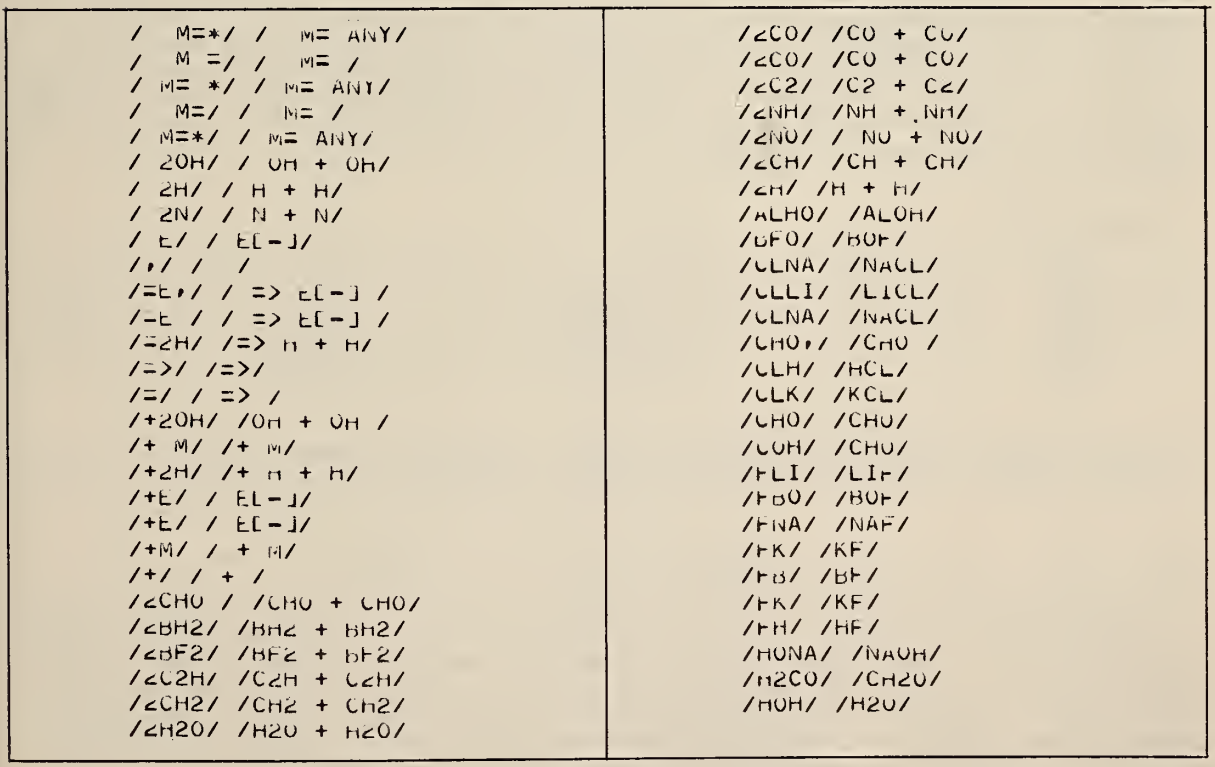


Figure 3f. A portion of a substitute table for editing equations representing chemical reactions.

4. Characteristics of the Programs SEARCH and BLOCKSEARCH

SEARCH and BLOCKSEARCH are useful for data retrieval. The first program searches a card image of a single line of text for the presence of any or all of a group of words or strings or fragments; it prints out the line or punches out a card when such items are located.

BLOCKSEARCH is able to scan an entire block of lines, making it generally more useful in data retrieval. This search is made not on a single line but on a suitably delineated block, such as a paragraph, a page, a full bibliographic citation, or an abstract. On a successful match the entire block is printed or punched.

An important feature of these searching programs is the ability to handle fragments such as prefixes or suffixes or even fragments in the interior of words. Ordinarily the scanning is anchored to the beginning of the word. For example, asking for the word "thermo" would produce lines with thermodynamics, Thermodynamics, thermo-chemistry, and Thermochemistry. If the program is set to the unanchored mode, it will locate the word Aerothermodynamics as well. In this search mode, imbedding a blank (represented here by a °) at the end of a string restricts the search to endings or suffixes Thus, when asked to locate FLEX°, the program will find CELLUFLEX, but not FLEXIBLE. Set to locate all lines containing both of the strings °CEL and LEX°, this program would locate all words beginning with CEL and/or ending with LEX, such as °CELLUFLEX° or COMPLEX CELLS.

A feature of both the SEARCH and BLOCKSEARCH programs is the ability to identify fragments of words as easily as entire words or phrases. Another interesting and important feature of these programs is their ability to recognize word fragments in the interior of words as well as at the ends. Since recognition of interior fragments requires more machine time, a switch has been provided to designate whether the search is to be "anchored" at the beginning of a word or allowed to proceed into the interior of a word. The combination of spaces (°) and anchor switch setting provide for the following search strategies.

ASKING FOR	WITH MODE	GIVES
ION°	anchored	ION
°ION°	unanchored	ION
ION	anchored	IONIZED
ION°	unanchored	ACTION
ION	unanchored	MENTIONED

Note that asking for ION in the unanchored mode will find lines containing prefixes and suffixes as well as interior fragments.

4.1 The Control Cards for SEARCH

The first control card is the same as the first card of the program SUBSTITUTE. It performs the same functions here. Here again cc47 is left blank but cc50 is now used to designate which character is used to terminate the search strings. If the strings are single words, a space will suffice; but if phrases are to be allowed the space cannot be used to terminate strings. In this, a period or any other symbol not appearing in the search text can be used.

The second card contains four switches in FORMAT (4I2).

a. This switch should be a non-zero number if a permanent file of the selected cards is desired either on cards or tape in addition to the normal printed output. In this case the unit on which the copy is made is designated by the fourth switch (see d. below). If this switch is zero, results appear only on the printer.
b. The second switch should be set to zero for searching in the anchored mode and non-zero for an unanchored search.
c. This switch instructs the machine to read the original data file from the specified unit. The unit is set to 5 if this switch is 0.
d. This switch instructs the machine on which unit to write the copy of the abridged file. The unit is set to 3 if this switch is 0.

The third control card carries either the word AND or OR. If the card carries the word OR, a line will be printed out if it contains any one of the designated words or fragments. The use of the word AND imposes the condition that all of the words must be found on a single card image for successful selection. The order in which they appear is immaterial.

The control cards which follow contain the words, phrases or fragments for which the file is to be searched. The search strings must start in ccl and must be terminated by the character designated in cc50 of the first control card. The entire list is terminated on a card carrying the word FINIS, starting in column 1 and terminated by the designated string terminator. This last seemingly unnecessary requirement permits one to include in the search such words as FINISHING which would otherwise terminate the search list.

Here again, when the program encounters any card which matches the first control card (a card with the normal alphabet punched in the first 26 columns), it comes to a normal machine halt.

4.2 The Operation of BLOCKSEARCH

As presently constituted, this program recognizes any fixed block separators (delimiters) starting in the first character of a record. The length of the block delimiter is optimal as is also its content. Thus if one wished to delimit blocks by identifying each block with the word BLOCK followed by a sequence of numbers, the string BLOCK nnn could be the first few characters in a paragraph.

A more natural delimiter for paragraphs would be the three or more spaces a typist normally uses to indent the first line of a paragraph: or a blank line which is often used in text having blocked paragraphs or more generally in separating items in a bibliography etc.

What is more important in this connection is the fact that many systematically structured books, reports or documents contain useable flags or separators which are part of the normal text. In these circumstances it is not necessary to include artifical block separators. This makes it possible to apply this program to many existing data files without special restructuring.

Blocks may be as small as one record (one line) or as large as the space reserved in the program in the dimension statement governing the size of the block. If a block is larger than the number of characters set aside in the program, the search is performed first on the front portion: independent of the remainder, and then independently on the remaining portions. The lines

MEYER		APK		127	253	1865	
ZOCH		APK		128	497	1866	
KUNDT		APK		156	177	1875	WARBURG
WIEDEMANN		APK		157	1	1876	
VANITTERBEEK		APQ		19	88	1944	VANDONINCK 38700
VASILESCO		APQ		20	137	1945	
VASILESCO		APQ		20	292	1945	38700
REGNAULT		ASP		26	1	1862	
MICHELS		ASR		1	94	1948	DEGROOT
MICHELS		ASR		1	103	1948	DEGROOT
MICHELS		ASR		4	52	1954	WASSENAAR 38700
TITANI	J	BCS		4	68	1929	38700
ADZUMI		BCS		12	199	1937	
MACCORMACK		CJC		29	699	1951	SCHNEIDER
WOOLLEY		CJP		31	604	1953	
COOPER		CJR		2	388	1930	MAASS
COOPER		CJR		4	283	1931	MAASS
SUTHERLAND		CJR		6	428	1932	MAASS 38700
TAPP		CJR		9	217	1933	STEACIE MAASS
WINCKLER		CJR		9	613	1933	MAASS
PITT		CJR		12	686	1935	JACKSON
DACEY		CJR	B	17	206	1939	MACINTOSH MAASS
MACINTOSH		CJR	B	17	241	1939	DACEY MAASS
CLARK		CJR		18	39	1940	KATZ
MASON		CJR	B	18	103	1940	NALDRETT MAASS
NALDRETT		CJR	B	18	118	1940	MAASS
KATZ		CJR		27	39	1949	LEVERTON WOODS
MASSON		CPR		44	464	1857	
LEDUC		CPR		113	166	1891	
AMAGAT		CPR		113	446	1891	
AMAGAT		CPR		114	1093	1892	
JAQUEROD		CPR		139	129	1904	PINTZA
GERMANN		CPR		157	926	1913	
GOIG		CPR		189	236	1929	
FORTIER		CPR		203	711	1936	
FORTIER		CPR		208	506	1939	
KASSEL		CRV		18	277	1936	
PIRE		EFQ		27	267	1929	MOLES
GOIG-BOTELLA		EFQ		27	315	1929	
MOLES		EFQ		30	182	1932	SALAZAR
MOLES		EFQ		32	954	1934	SALAZAR
MOLES		EFQ		35	42	1937	TORAL
MOLES		EFQ		35	263	1937	ROQUERO
SACKUR		ETC		20	563	1914	
DAMKOHLER		ETC		48	62	1942	
FRANCK		ETC		55	636	1951	
GOFF		HPC		18	125	1946	GRATCH
MAJUMDAR		IAS		8	171	1938	
SWEIGERT		IEC		38	185	1946	WEBER ALLEN
PORTER		JAC		48	2059	1926	PERRY
DODGE		JAC		49	610	1927	DAVIS

Figure 4a. A portion of a file of references on data of state used to illustrate the operation of the program SEARCH. See the following figures for results.

ABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789 *
1 0 5 7
AND

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

193

ADZUMI	BCS		12	199	1937		
COOPER	CJR		2	388	1930	MAASS	
COOPER	CJR		4	283	1931	MAASS	
SUTHERLAND	CJR		6	428	1932	MAASS	3870
TAPP	CJR		9	217	1933	STEACIE MAASS	
WINCKLER	CJR		9	613	1933	MAASS	
PITT	CJR		12	686	1935	JACKSON	
DACEY	CJR	B	17	206	1939	MACINTOSH MAASS	
MACINTOSH	CJR	B	17	241	1939	DACEY MAASS	
FORTIER	CPR		203	711	1936		
FORTIER	CPR		208	506	1939		
KASSEL	CRV		18	277	1936		
MOLES	EFQ		30	182	1932	SALAZAR	
MOLES	EFQ		32	954	1934	SALAZAR	
MOLES	EFQ		35	42	1937	TORAL	
MOLES	EFQ		35	263	1937	ROQUERO	
MAJUMDAR	IAS		8	171	1938		
BARTLETT	JAC		52	1363	1930	HETHERINGTON	
BARTLETT	JAC		52	1374	1930	HETHERINGTON	
CLAYTON	JAC		54	2610	1932	GIAUQUE	
GIAUQUE	JAC		55	4875	1933	CLAYTON	
ASTON	JAC		57	1642	1935	WILLIHNGANZ	
BENEDICT	JAC		59	2224	1937		
BENEDICT	JAC		59	2233	1937		
WIEBE	JAC		60	2300	1938	GADDY	
HODGE	JCP		5	974	1937		3870
HUBBARD	JCP		5	978	1937	HODGE	
KISTIAKOWSKY	JCP		7	281	1939	RICE	3870
ROSSINI	JRS		9	733	1932		3870
MEYERS	JRS		10	381	1933	VANDUSEN	
VARGAFTIK	JTP		8	189	1938	PARFENOV	
KEESOM	LCS		19	27	1931	VANITTERBEEK NO 216D	
CROMMELIN	LCS		20	10	1931	BIJLEVELD NO217B	
KEESOM	LCS		22	1	1937	BIJL NO 245D	
BOND	NAT		137	1031	1936		3870
VANITTERBEEK	NAT		142	793	1938	CLAES	
ROEBUCK	PAS		64	287	1930		3870

Figure 4b. A portion of the output from the search of the file shown in the previous figure. The second control card instructs the program to read from unit 5 and write the output on unit 7. The zero denotes an "anchored" search. In this case the search is made for 193 as we wished to extract all papers published in the 1930's. While the AND in the third control card seems superfluous, it, or the word OR, is required. In this case either word would do equally well.

at which the blocks were broken in this instance, are printed out preceded by the statement:

BLOCK IS TOO LONG. BLOCK IS CONSIDERED TERMINATED WITH THE FOLLOWING CARD.

In order to operate successfully in the anchored mode, it is necessary to insure the existence of a space (°) between the first character of a line and the line before it. This is handled automatically by the program which inserts a blank space before each line when the block is read in. This provision allows for carrying sequence numbers extending to column 80 of a card or the last character of the record. Thus, if a card or line ends in 1798 and the word "JONES" appears in the next line, the string to be searched reads 1798°JONES rather than 1798JONES. In the former, a search on JONES in the anchored mode would be successful; in the latter case, it would not.

The first control card is identical with the first control card for SEARCH.

The second control card contains 6 switches a, b, c, d, e and f in FORMAT (5I2, I14).

a. This switch, when non-zero, causes a magnetic tape to be written on the unit specified by switch e.
b. This switch should be set to zero for an anchored search. Any non-zero number causes the search to be carried out in the unanchored mode.
c. This switch tells how many characters to expect on the next control card which carries the block separator (flag). If this number is not an integer between 1 and 80, the program sets it to 80 automatically.
d. This switch gives the unit number from which to read the data file to be searched.
e. This switch is the unit on which to write the selected blocks.
f. This switch tells the length of the records which comprise the file to be searched. If this switch is not between 1 and 132 it is automatically set to 80 characters. It is also useful in limiting the reading of cards to 72 or less columns to ignore identification or sequence numbers which are normally punched at the extreme right of a card. In this way when the sequence numbers carry no pertinent information, they are ignored in the search and deleted on the output. If, on the other hand, they carry useful information, this switch permits them to be included both in the search and on the final output.

The third control card contains the block separator starting in cc1 and having a length specified by switch c of the previous card. If c were 5 and this card were blank, the normal paragraph indention would serve to delimit paragraphs in straight text. If c were set to 80 and this card were left entirely blank, a blank line of 80 characters could serve as a block separator.

The control cards which follow contain the AND or OR switch and the search list terminated by the word FINIS exactly as in the program SEARCH (see Section 4.1).

```
ABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789          *
 1 0 7 3
OR

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

JAC
JCP
JRS


      PITT          CJR      12    686    1935    JACKSON
      BARTLETT      JAC      52   1363    1930    HETHERINGTON
      BARTLETT      JAC      52   1374    1930    HETHERINGTON
      CLAYTON       JAC      54   2610    1932    GIAUQUE
      GIAUQUE       JAC      55   4875    1933    CLAYTON
      ASTON         JAC      57   1642    1935    WILLIHNGANZ
      BENEDICT      JAC      59   2224    1937
      BENEDICT      JAC      59   2233    1937
      WIEBE         JAC      60   2300    1938    GADDY
      HODGE         JCP       5    974    1937                    3870
      HUBBARD       JCP       5    978    1937    HODGE
      KISTIAKOWSKY  JCP       7    281    1939    RICE            3870
      ROSSINI       JRS       9    733    1932                    3870
      MEYERS        JRS      10    381    1933    VANDUSEN
ABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789          *
```

Figure 4c. The results of a pass through the abridged file shown in the previous figure to extract all references appearing in the journals whose designations are JAC, JCP, and JRS. Note here that the second control card instructs the program to read from unit 7, which contained the abridged file stored by the previous pass (see Figure 4b). This problem illustrates how sequential applications of the simple SEARCH program can result in more sophisticated search strategies than have been explicitly provided. Note that in the first line the JAC in JACKSON also provides a hit. In a production search this can be avoided by attaching a blank space (JAC°).

```
ABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789
 0 1 3 5 3
CA

THE PROGRAM IS SEARCHING IN LINES OF LENGTH     80 CONTAINING ANY OF THE WO

 RAM *
NITRO*
FINIS*


                                                                  063000E0
 VIBRATIONAL SPECTRA OF GEM-DINITROPARAFFINS. I. 2,2-             063000E0
 DINITROPROPANE.                                                  063000E0
 BUCZKOWSKI, Z                                                    063000E0
 URBANSKI, T                                                      063000E0
 SPECTROCHIM. ACTA 22, 227-33 [1966]                              063000E0
 MICRO                                                            063000E0
 SPAC                                                             063000E0
 1966                                                             063000E0
 IRS                                                              063000E0

 VIB                                                              063000E0
 2,2-DINITROPROPANE                                               063000E0
 CA 64009089HLIT                                                  063000E0
 VIBRATIONAL SPECTRA OF TRIMETHYLAMINE OXIDE DIHYDRATE.           063000E0
 KURODA, Y                                                        063000E0
 KIMURA, M                                                        063000E0
 SPECTROCHIM. ACTA 22, 47-56 [1966]                               063000E0
 MICRO                                                            063000E0
 SPAC                                                             063000E0
 1966                                                             063000E0
 IRS                                                              063000E0
 RAM                                                              063000E0
 VIB                                                              063000E0
 TRIMETHYLENE OXIDE                                               063000E0
ABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789
```

Figure 4d. Results from a run in BLOCKSEARCH for either °RAM° or NITRO. As the search was made in the unanchored mode, we picked up the word GEM-DINITROPARAFFINS, as was intended. Note here that the characters CA° were used as the block delimitor. The last card containing the alphabet is the last card of the data file. It is used instead of an "end of file" mark to signal the end of data because an end of file mark in some machines stops the operation dead.

```
ABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789              *
 0 1 3 5 3
)

THE PROGRAM IS SEARCHING IN LINES OF LENGTH    80 CONTAINING ANY OF THE WORDS GIVE


SPECT*
SPEKT*
FINIS*



)
  01205 'T'ILFORD, 'S'.'G'., 'V'ANDERSLICE, 'J'.'T'. AND 'W'ILKINSON,)     1650101
  01205   'P'.'G'., +'HIGH.RESOLUTION VACUUM ULTRAVIOLET ABSORPTION')     2650101
  01205   'SPECTRUM OF THE E '1 'PI ... X SIGMA + TRANSITION IN CO'+,)     3650101
  01205   'C'AN. 'J'. 'P'HYS. 43, 450(1965'(')                            4650101
)
  01209 'N'AMIOKA, 'T'., +'ABSORPTION SPECTRA OF H2 IN THE VACUUM')        164
  01209   'ULTRAVIOLET REGION'. 'II'. 'THE B'(PRIME'(.X', 'B'+'.X',)      264
  01209   'D.X', 'AND D'(PRIME'(.X BANDS'+, 'J'. 'C'HEM. 'P'HYS. 41,)     364
  01209   2141(1964'(')                                                    464
)
  01212 'B'ORISOV, 'Y'.'A'., 'G'USAROV, 'A'.'V'. AND 'G'OROKHOV,)          1640201
  01212   'L'.'N'., +'MASS SPECTROMETER INVESTIGATION OF THE')             2640201
  01212   'EVAPORATION OF CESIUM PEROXIDE'+, 'H'IGH 'T'EMPERATURE 2,)     3640201
  01212   440(1964'(')                                                     4640201
)
  01217 'H'UBER, 'K'.'P'., +'DIE RYDBERG.SERIEN IM ABSORPTIONSSPEKTRUM')   1610101
  01217   'DES NO.MOLEKULS'+, 'H'ELV. 'P'HYS. 'A'CTA 34, 929(1961'(')     2610101
)
  01218 'G'LADUSHCHAK, 'V'.'I'. AND 'S'HREIDER, 'E'.'Y'A., +'THE')         1640101
  01218   'MEASUREMENT OF ABSOLUTE INTENSITIES IN THE VACUUM REGION')      2640101
  01218   'OF THE SPECTRUM'+, 'O'PTICS AND 'S'PECTROSCOPY 'USSR '17, 75)   3640101
  01218   (1964'(')                                                        4640101
)
CA 64009085DLIT                                    063000E07320  1
VIBRATIONAL SPECTRA OF GEM-DINITROPARAFFINS.  I. 2,2-   063000E07321  1
DINITROPROPANE.                                    063000E07321  2
BUCZKOWSKI, Z                                      063000E07322  1
URBANSKI, T                                        063000E07322  2
SPECTROCHIM. ACTA 22, 227-33 [1966]                063000E07323  1
MICRO                                              063000E07324  1
SPAC                                               063000E07325  1
1966                                               063000E07325  1
IRS                                                063000E07325  1
RAM                                                063000E07325  1
VIB                                                063000E07325  1
2,2-DINITROPROPANE                                 063000E07325  1
)
CA 64009085GLIT                                    063000E07330  1
THE ELECTRONIC EMISSION SPECTRUM AND MOLECULAR CONSTANTS OF   063000E07331  1
IODINE MONOFLUORIDE.                               063000E07331  2
DURIE, R A                                         063000E07332  1
CAN. J. PHYS. 44, 337-52 [1966]                    063000E07333  1
MICRO                                              063000E07334  1
CJPH                                               063000E07335  1
```

Figure 4e. Result of an unanchored block search in a mixed data file. The search was for the fragments SPECT or SPEKT in order to retrieve titles in English and German related to spectroscopy. Note that the only format that the two data files have in common is that each uses the right parenthesis in column 1 as a block separator.

5. Preparation and Editing of Manuscripts

Preparation of manuscripts and other material which requires numerous or periodic revision ordinarily entails repeated retyping and, consequently, needs repeated proofreading. If such material is prepared originally on punched cards, much of the retyping and subsequent proofreading can be avoided. Even without a computer, the use of punched cards represents considerable advantage. Until recently, however, punched card techniques did not permit the production of text with upper and lower case, thus limiting the sphere of application. Now a number of manufacturers offer line-printers with extended character sets. It is therefore now possible to prepare camera-ready copies from a line printer via punched-card input which has upper and lower case letters with subscripts and superscripts and greek letters. Where graphic art quality is desired, the same tape can be processed by an edit-insertion program now available commercially.

The computer programs discussed here have been developed in the course of the last three years to assist in the preparation of a number of documents on an 870 Document Writer (a card-controlled typewriter) for reports not requiring typesetting.

Any editing system which permits the deletion or addition of substantial segments of text must have some provision for rearranging the words into lines of specified length. JUSTIFY is a text formatting program which provides this facility with or without justified right-hand margins. It can center lines of text, indent, and perform other features useful in preparing camera-ready copy. When line justification is called for, the extra spaces are placed first after each period in the line and then between the words, starting from the left in one line and from the right in the next. Because simplicity of use is an important factor in the design of these programs, the rules are kept as conventional as possible. For example, the start of a paragraph is ordinarily signaled by leaving at least one blank space at the beginning of the line -- just as a typist might indent when starting a new paragraph -- or by inserting a blank card which is equivalent to leaving a blank line when the first line of a paragraph is not indented.

An example of the application of JUSTIFY to change the line length of text is afforded by Figures 5j and 5k. In this instance new lines were created with a maximum of 65 characters so as to produce one line per card without continuation cards. In this option the new text was not justified and words are separated by a single space, even though the original text contained extra blanks. This feature of the program permits the reworking of text which had previously been justified by this program.

The reader's attention is directed to the fact that the JUSTIFY program does not have a provision for hyphenating words. Hyphenation becomes important only when one uses a double column spread or a narrow newspaper column width. Figures 5a. et seq. show portions of a number of publications produced from the output of this program. We leave it to the reader to decide whether the lack of hyphenation effects the readability.

It should be noted that where an exceptionally long word falls just short of fitting at the end of a line and the space between words is therefore excessive, it is possible to hyphenate the word at a suitable point and rejustify the paragraph.

5.1 Characteristics of JUSTIFY

Special instructions to the JUSTIFY program can be given on the input control cards. The presence of certain characters in card column 72 of any text card will cause the JUSTIFY program to take special action. Any symbols or characters except those stated below will be treated as ordinary text when appearing in cc 72 of any text card. Since cc 72 is used for sensing control characters and since these characters may normally appear in straight text, it is unwise to punch text beyond cc 71. The last paragraph on any run should be followed by several blank cards.

Text prepared on a card usually has a number of precedence characters for case shift, for card eject (either with or without carriage return), for tabulation, and for half-line platen shift symbols. The JUSTIFY program recognizes specific special control symbols wherever they occur on the text card.

JUSTIFY operates on a "paragraph" at a time. A "paragraph" is considered started when:

a. the first card of input is encountered.
b. or a blank occurs in cc 1 of any text card.
c. or the previous card contained a special control character in cc 72 (the control field).
d. or the previous card contained only a card ejection symbol (the symbol appearing in cc 39 of the first control card).
e. or the previous card was totally blank.

A "paragraph" is considered terminated when:

a. the card immediately following contains a blank in cc 1.
b. or the following text card contains a control character in cc 72. The only exception is when cc 72 carries the ignore character.
c. or the following card contains only a card ejection symbol.
d. or the following card is totally blank.

5.2 The Control Cards for JUSTIFY

All control characters and control symbols are indicated on the first input control card.

The function of the first control card is described in Section 3.1. As with the other programs in this package, card columns 38-80 of this card are reserved for symbols which have meaning specific to the particular program. They perform the following functions:

a. The symbol placed in cc 38 indicates the sentence terminator - normally a period.
b. The symbol placed in cc 39 indicates the card eject symbol. Cards are not read beyond that point.
c. Card columns 40-44 are not used in this program.
d. The symbol placed in cc 45 indicates the continuation flag. It signals that the line of text is continued on another card. Although originally implemented to provide card eject without carriage return or line feed on a card-controlled typewriter, it is also useful in continuing on another card a string of characters which must be produced in the "as is" form rather than in the justified mode.
e. The symbol in cc 46 is a nonprint control character used to indicate a case shift on a typewriter. It is sensed and used by the subroutines, LOCK and UNLOCK.

```
     'A'NY SYMBOLS EXCEPT THOSE STATED ABOVE WILL BE TREATED AS ORDINARY
TEXT WHEN APPEARING IN CARD COLUMN 72.  )
  )
     'S'INCE COLUMN 72 IS USED FOR SENSING THE ABOVE CONTROL CHARACTERS  )
AND SINCE THESE CHARACTERS MAY NORMALLY APPEAR IN STRAIGHT TEXT,  )
IT IS UNWISE TO PUNCH TEXT BEYOND COLUMN 71.  )
  )
                  'TABLE' 2.  )                                          C
  )
'S'PECIFICATIONS OF 'S'PECIAL 'S'YMBOLS.  )                              C
  )
     'T'HE SPECIAL SYMBOLS WILL BE IDENTIFIED BY THEIR POSITION  )
ON THE INPUT CONTROL CARD '('SEE 'F'IGURE 1(.  'O'UR CURRENT CONVENTION )
USES THE SYMBOLS '(', '5', BLANK, '4', AND '0' IN THE LOCATIONS  )
DESIGNATED BELOW. ()
```

Figure 5a. A listing of cards punched for a card controlled typewriter with control symbols in cc72 which are recognized by the JUSTIFY program. The right parenthesis produces a carriage return and line feed on the typewriter, consequently the control symbols in cc72 are not typed on the typewriter copy (See Figure 5c.).

```
  )
     'A'NY SYMBOLS EXCEPT THOSE STATED ABOVE WILL  BE  TREATED)
AS ORDINARY TEXT WHEN APPEARING IN CARD COLUMN 72.)
  )
     'S'INCE COLUMN 72 IS USED FOR SENSING THE  ABOVE  CONTROL)
CHARACTERS AND SINCE THESE CHARACTERS MAY NORMALLY APPEAR IN)
STRAIGHT TEXT, IT IS UNWISE TO PUNCH TEXT BEYOND COLUMN 71.)
  )
                         'TABLE' 2.                                      )C
  )
          'S'PECIFICATIONS OF 'S'PECIAL 'S'YMBOLS.                       )C
  )
     'T'HE  SPECIAL  SYMBOLS  WILL  BE  IDENTIFIED  BY  THEIR)
POSITION  ON  THE  INPUT  CONTROL  CARD  '('SEE 'F'IGURE 1(. 'O'UR)
CURRENT CONVENTION USES THE SYMBOLS '(', '5', BLANK, '4', AND '0' IN)
THE LOCATIONS DESIGNATED BELOW.)
  )
  )
                       'A('39(  IS  (                                    )C
  )
 'A('39( TERMINATES THE READING OF A CARD. 'A'NYTHING PUNCHED IN)
FURTHER FIELDS OF THE CARD WILL BE IGNORED.)
 'I'F IT IS THE FIRST NON-BLANK  CHARACTER  ENCOUNTERED  WHILE)
READING A TEXT CARD, THE PREVIOUS PARAGRAPH IS CONSIDERED AT)
AN END. 'T'HE CARD ITSELF IS REPRODUCED AS-IS IN THE OUTPUT.)
 'I'T IS THE LAST CHARACTER ON ANY TEXT OUTPUT CARD.)
  )
```

Figure 5b. A listing of the cards produced by the program JUSTIFY from the cards listed in Figure 5a. The ragged appearance of the lines results from the varying numbers of shift symbols from line to line.

Any symbols except those stated above will be treated as ordinary text when appearing in card column 72·

Since column 72 is used for sensing the above control characters and since these characters may normally appear in straight text, it is unwise to punch text beyond column 71·

TABLE 2·

Specifications of Special Symbols·

The special symbols will be identified by their position on the input control card (see Figure 1)· Our current convention uses the symbols (, ', blank, =, and _ in the locations designated below·

Figure 5c. Copy produced on a card-controlled typewriter from the cards listed in Figure 5a. The ragged lines results from editorial changes or numerous shift symbols which take up space on the card but do not produce characters on the line.

Any symbols except those stated above will be treated as ordinary text when appearing in card column 72·

Since column 72 is used for sensing the above control characters and since these characters may normally appear in straight text, it is unwise to punch text beyond column 71·

TABLE 2·

Specifications of Special Symbols·

The special symbols will be identified by their position on the input control card (see Figure 1)· Our current convention uses the symbols (, ', blank, =, and _ in the locations designated below·

A(39) is)

A(39) terminates the reading of a card· Anything punched in further fields of the card will be ignored·
If it is the first non-blank character encountered while reading a text card, the previous paragraph is considered at an end· The card itself is reproduced as-is in the output·
It is the last character on any text output card·

Figure 5d. The typewriter output of the justified text produced from the cards listed in Figure 5b.

f. The symbol in cc 47 must be a blank. It indicates that a blank is the character at which lines will be broken. The reason the program removes excess blanks between words is that cc 47 was left blank. If cc 47 contained any other symbol, it would be the multiple contiguous occurrences of that symbol that would be squeezed out.

g. The symbol in cc 48 is a tab indicator. This symbol is treated as a single blank when it is encountered in the input text except that it does not, therefore, start a new paragraph when it happens to fall in cc 1.

h. Card columns 49-59 are not used by this program.

i. The number in cc 60 indicates how many symbols should be considered as having no width in the line justification process. This number is used to clarify how many of the subsequent card columns are to be scanned for the specific "no count" symbols.

j. The symbols to be ignored in counting the characters for line justification are punched into card columns 61 et seq as required.

The second control card contains seven switches in FORMAT (7I3).

a. This switch sets the maximum line length (the number of characters per line). The symbols discussed in item j above are not counted when formatting a line of output. Switch a must be between 20 and 120.

b. Here any non-zero entry produces lines with flush right hand margins (justified).

c. If this switch is non - zero, card images are written on the unit specified by switch f below. The program provides for continuation cards when the line is longer than 80 characters. Even a line shorter than 80 characters will often be carried over to the next card because of the space taken up by the precedence symbols. When a line is broken between two or more cards, care is taken never to split words. The symbol which appears in cc 45 of the first control card is punched as the continuation symbol.

d. This switch, if non-zero, calls the subroutines LOCK and UNLOCK. For all application except cards intended for the 870 Document Writer, this switch should be set to zero.

e. This switch tells from which unit to read the input data.

f. This switch tells on which unit to write the card images or punch the cards.

g. If this switch is zero, the program gives priority to the insertion of a second space at the end of each sentence before distributing the necessary spaces between the words. The extra space is inserted automatically when the lines are not justified.

5.3 The Format Control Symbols

The program has provision for line justification in accord with a specified line width. Practical formatting often requires exempting certain lines from the justification process. Such exemptions are signaled by the characters C, D, I, N, and R when they appear in cc 72.

A letter C in card column 72 causes the information in the previous columns of that card to be centered on the line. The letter D in cc 72 causes the line to be printed exactly as it is punched on the card. The letter N in cc 72 sets a switch to indent the subsequent text lines. The first non-blank characters on the "N" card signals the new width. Another "N" card carrying the original width is required to restore the lines to the normal width. The previously defined tab symbol is automatically inserted at

the beginning of each line of the indented text.

Some data and text files contain lines of information which have a specific use in the present format but must be ignored when the file is reformatted. A letter I in cc 72 causes that line to be ignored completely by this program.

An R in cc 72 generates a box of depth equal to the number which appears on the "R" card. This is used to reserve a space for inserting illustrations. The "R" card is converted to a "D" without further modification, and positioned as the center line of the box. Thus the "R" card can be used to indicate which figure is to be placed in the box.

1. It is hard to overemphasize the importance and usefulness of self-teaching, described in section 4.6. One can learn the basic rules of OMNITAB very quickly, say within two hours, and then proceed on his own. This is very valuable for those of us who have difficulty reading manuals carefully.

Many times little programs have been added (at virtually no cost) at the end of a program to clarify a particular instruction. For example, it is not completely clear whether the command RMS described on page 38 stores

$$\sqrt{\Sigma x_i^2/n}\ ,\ \sqrt{\Sigma(x_i-\bar{x})^2/n}\quad \text{or}\ \sqrt{\Sigma(x_i-\bar{x})^2/(n-1)}\ .$$

However, the question is readily answered by using the following instructions:

```
OMNITAB
GENERATE 1. (1.) 5. STORE 1
RMS 1 STORE 2
PRINT 1 2
STOP
```

(Note, Σx_i^2 for integers is easily obtained from the formula $1^1 + 2^2 + \ldots + n^2 = n(n+1)(2n+1)/6$ so that $\Sigma x_i^2/n = (n+1)(2n+1)/6$.) It would be easy to write many programs like this in a short period of time and very quickly become an ''expert'' user. The ease with which this can be done is sometimes overlooked or unappreciated. The technique of self-teaching has been used effectively in class instruction.

2. An obstacle faced by non-programmers is that with some languages, such as FORTRAN, it is almost necessary to be an expert programmer to do even the simplest calculation. To do anything in FORTRAN, it is necessary to understand some of the most troublesome statements of the language such as FORMAT, DIMENSION and WRITE. An OMNITAB user with no computing experience can being writing programs to perform non-trivial computations with less than two hours' study. By self-teaching he can proceed at his own speed to become an ''expert.''

3. The very nature of OMNITAB is such as to make logical branching unnecessary in most instances. In others it may be possible with a little ingenuity. For example, the following instructions provide for replacing $\log_{10}(a_i/b_i)$ by the constant c whenever $a_i \leq 0$. Capital letters A, B, D, and F represent column numbers; corresponding lower case letters represent elements in the columns. We assume $b_i > 0$.

Figure 5e. A page of a mechanized text produced on a modified card-controlled typewriter. See the next figure for a listing of the cards that produced this page.

```
 1.:::'I'T IS HARD TO OVEREMPHASIZE THE IMPORTANCE AND USEFULNESS OF)
 SELF['0']TEACHING, DESCRIBED IN SECTION 4.6.  'O'NE CAN LEARN THE $)
 BASIC)
 RULES OF 'OMNITAB' VERY QUICKLY, SAY WITHIN TWO HOURS, AND THEN)
 PROCEED ON HIS OWN.  'T'HIS IS VERY VALUABLE FOR THOSE OF US WHO)
 HAVE DIFFICULTY READING MANUALS CAREFULLY.)
  )
      'M'ANY TIMES LITTLE PROGRAMS HAVE BEEN ADDED '('AT VIRTUALLY NO)
 COST(  AT  THE END OF A PROGRAM TO CLARIFY A PARTICULAR)
 INSTRUCTION.  'F'OR EXAMPLE, IT IS NOT COMPLETELY CLEAR WHETHER THE)
 COMMAND 'RMS' DESCRIBED ON PAGE 38 STORES)
  )
     ΔX]I[[2]/N ,  Δ'('X]I[['0']X([2]/N  OR  Δ'('X]I[['0']X([2]/'('N$)  D
 [']1(.) I                                                              D
   )
 'H'OWEVER, THE QUESTION IS READILY ANSWERED BY USING THE FOLLOWING)
 INSTRUCTIONS'1')
  )
      'OMNITAB')
      'GENERATE' 1. '('1.( 5. 'STORE' 1)
      'RMS' 1 'STORE' 2)
      'PRINT' 1 2)
      'STOP')
  )
 '(N'OTE, :ΔX]I[[2] FOR INTEGERS IS EASILY OBTAINED FROM THE FORMULA.)
  [)                                                                    I
 1[1] + 2[2]:+ ... +  N[2]:'4'  N'('N+1('('2N+1(/6  SO  THAT  $)
 :ΔX]I[[2]/N:'4')
  [)                                                                    I
 '('N+1('('2N+1(/6.( 'I'T WOULD BE EASY TO WRITE MANY PROGRAMS LIKE $)
 THIS)
 IN A SHORT PERIOD OF TIME AND VERY QUICKLY BECOME AN '55'EXPERT'55')
 USER.  'T'HE EASE WITH WHICH THIS CAN BE DONE IS SOMETIMES)
 OVERLOOKED OR UNAPPRECIATED.  'T'HE TECHNIQUE OF SELF['0']TEACHING $)
 HAS)
 BEEN USED EFFECTIVELY IN CLASS INSTRUCTION.)
  )
 2.:::'A'N OBSTACLE FACED BY NON['0']PROGRAMMERS IS THAT WITH $)
 SOME)
 LANGUAGES, SUCH AS 'FORTRAN', IT IS ALMOST NECESSARY TO BE AN)
 EXPERT PROGRAMMER TO DO EVEN THE SIMPLEST CALCULATION.  'T'O DO)
 ANYTHING IN 'FORTRAN', IT IS NECESSARY TO UNDERSTAND SOME OF THE)
 MOST TROUBLESOME STATEMENTS OF THE LANGUAGE SUCH AS 'FORMAT',)
 'DIMENSION' AND 'WRITE'.  'A'N 'OMNITAB' USER WITH NO COMPUTING $)
 EXPERIENCE)
 CAN BEING WRITING PROGRAMS TO PERFORM NON['0']TRIVIAL COMPUTATIONS)
 WITH LESS THAN TWO HOURS'5' STUDY.  'B'Y SELF['0']TEACHING HE CAN PROCEED)
 AT HIS OWN SPEED TO BECOME AN '55'EXPERT.'55')
  )
 3.:::'T'HE VERY NATURE OF 'OMNITAB' IS SUCH AS TO MAKE LOGICAL)
 BRANCHING UNNECESSARY IN MOST INSTANCES.  'I'N OTHERS IT MAY BE)
 POSSIBLE WITH A LITTLE INGENUITY.  'F'OR EXAMPLE, THE FOLLOWING)
 INSTRUCTIONS PROVIDE FOR REPLACING LOG]10[:'('A]I[/B]I[( BY THE $)
 CONSTANT)
  [)                                                                    I
 C WHENEVER A]I[:'8'Δ:0.  'C'APITAL LETTERS 'A', 'B', 'D', AND $)
 'F' REPRESENT)
```

Figure 5f. A listing of cards produced by JUSTIFY to prepare the text in the previous figure. See Section 5.4 for a discussion of this application.

5.4 Two Applications of JUSTIFY

In this section we describe briefly two applications which motivated the development and influenced the design of the programs JUSTIFY and SUBSTITUTE. Both of these resulted in publications produced from camera-ready copy on a modified 870 Document Writer (a card-controlled typewriter). Figure 5e shows a page from NBS Handbook 101 (loc. cit.). The subscripts and superscripts were obtained by automatic platen rotation signaled by two special multipunches. As with shift symbols, the program recognizes them as being characters of zero width for line justification. The program has room for as many as nine such special symbols.

The listing in Figure 5f shows the character stream produced by JUSTIFY to generate the contents of Figure 5e on a modified card-controlled typewriter. Among the features to be noted in Figure 5e are these:

a. the colon (:) designates a special blank which is treated like any other character to retain the spacing in the first line of each of the numbered paragraphs.
b. the square brackets cause fractional platen rotation, are ignored in the character count, and do not appear in the typed copy.
c. the delta is a stop code to permit the insertion of characters not present on the typewriter.
d. when the line contains a large number of shift symbols, it is continued on the next card and the $ sign signals the typewriter to eject the card without returning the carriage. Note that the JUSTIFY program does not break words.
e. the "D" in cc72 signifies a line that was reproduced as is - without justification.
f. the lines with a "I" in the control field (cc72) were inserted by hand to produce a half-line space before lines with superscripts and after lines with subscripts. Because of the "I" in cc72, these lines will be ignored on any subsequent pass through JUSTIFY.

An important motivation in the development of the general purpose programs discussed here has been our desire to be able to reformat and recast data and text files at will. The typset page from the NBS Postdoctoral Research Associateships announcement booklet for 1968-1969 shown in Figure 5g affords a good example of the utility of the EDPAC programs. The 94 page booklet of which Figure 5g is an sample was typset automatically from cards punched in previous years to drive a card-controlled typewriter. In order to achieve the typsetting without rekeyboarding, it was necessary to transform the information to conform to the punching convention used by the Administrative Terminal System (ATS) on a 1440 computer, from which system the material went forward for typsetting. Figures 5j, 5k, 3b and 3c show how existing cards from the earlier publication were transformed to produce the typset version in Figure 5g. A page from this publication for the previous year produced on a card-controlled typewriter with the help of JUSTIFY is shown in Figure 5h. Prior to the development of JUSTIFY the same material appeared with ragged right hand margins as can be seen from Figure 5i.

The circled characters in Figure 5k were inserted by an ad hoc modification of JUSTIFY as they were required by the typsetting system on which the 1968-1969 version of the Postdoctorial booklet was produced. The = sign instructs the ATS to produce a new line positioned at the left margin (quad left). The sequence ;/ was inserted after the first character and the last character of a title (a card with a "D" in cc72) in order to set the title in boldface type. Since the ATS has its own justification program which does not remove extra blanks, the chore of justification was delegated to the ATS. JUSTIFY in Figure 5k produced unjustified lines by removing the extra spaces which were present in the existing cards.

Hydraulics and Hydrodynamics

G. Kulin

Research is currently centered on water surface waves (with emphasis on various phenomena affecting wave damping) and internal waves in density-stratified water (with emphasis on wave generation by motion of submerged objects). There is opportunity for theoretical as well as experimental work in these areas. Facilities are also available for open- channel flow and sediment-transport research.

Dynamic Measurement of Properties of Solids and Liquids at Very High Temperatures

C. W. Beckett and A. Cezairliyan

Current research includes the investigation of thermodynamic and related properties of solids and liquids at high temperatures by dynamic experimental techniques, such as, pulse calorimetry, exploding wires, electrical discharges, etc. Advanced measurement techniques for obtaining both microscopic and macroscopic properties are being explored. The equipment includes pulse calorimeters of millisecond and microsecond time resolution, high-speed photoelectric pyrometers, and ultra high-speed framing camera, a high-speed digital recording system and other auxiliary dynamic and steady-state measurement instruments.

Combustion and Reaction Calorimetry

G. T. Armstrong

Relationships between binding energy and structure are being investigated among organic and inorganic compounds. Recent emphasis has been on fluorine compounds, nitro compounds, and compounds of biological interest. A rotating bomb calorimeter, facilities for bomb or flow colorimetry with fluorine, and other calorimetric facilities are available for high precision measurements. The calorimetric process, reference materials for reaction calorimetry, and new forms for correlating calorimetric measurments may also be investigated.

Production and Measurement of Very High Temperatures

John B. Shumaker

A program of research in the measurement of temperatures above 10,000°C is being conducted in the High Temperature Measurements Laboratory. Current interests include the detailed investigation of stable high current density arcs and plasma jets and the measurement of their temperatures and related physical parameters by a variety of spectroscopic techniques.

Figure 5g. A typical page from the NBS Postdoctoral Research Associateships announcement booklet which was typset from cards transformed by JUSTIFY and SUBSTITUTE. See Figure 5h for the same section for the year 1967-1968.

Hydraulics and Hydrodynamics:
G. Kulin

Research is currently centered on water surface waves (with emphasis on various phenomena affecting wave damping) and internal waves in density-stratified water (with emphasis on wave generation by motion of submerged objects). There is opportunity for theoretical as well as experimental work in these areas. Facilities are also available for open- channel flow and sediment-transport research.

Dynamic Measurement of Properties of Gases, Liquids, and Solids at Very High Temperatures:
C. W. Beckett

Current research includes the investigation of thermodynamic and related properties of gases, liquids, and solids at high temperatures by dynamic experimental techniques, such as shock tubes, exploding wires, and electrical discharges. Advanced measurement techniques for obtaining both microscopic and macroscopic properties are being explored. The equipment includes an ultra- high speed framing camera, a time-resolved spectrometer, and high-speed photoelectric pyrometers.

Molecular Spectra and Energy Levels:
A. M. Bass

Studies of energy levels and structures of diatomic and small polyatomic molecules as determined from the analysis of spectroscopic data. Facilities include spectroscopic instrumentation permitting observations from 500 Å in the vacuum ultraviolet to 53 microns in the infrared. A 21-foot focal length vacuum spectrograph provides the capability for obtaining high-resolution spectra. Also available are various sources for exciting spectra, including electric discharges, flames, flash photolysis and flash heating. The program also includes studies of transient species stabilized by condensation in low-temperature inert matrices.

Microwave Spectra of Gaseous Radicals:
H. E. Radford

Experiments are performed to determine the hyperfine structure, Zeeman effect and lambdatype doubling of simple diatomic radicals produced in gas mixtures by electric discharges, chemical reactions, triggered explosions, and photolysis. The paramagnetic resonance technique is used mainly, and spectrometers of various types are available for work at 3 Gc, 9 Gc and 24 Gc in magnetic field strengths up to 24 Kg. Optical and radio-frequency apparatus is also available for radio-optical resonance experiments on diatomic molecules, and for measurements of radio spectra in molecular excited states. Recent successful experiments in these two fields of study have been performed on the paramagnetic resonance spectra of OH and SH, and on the rotational perturbation microwave spectrum of excited CN. Other experiments are in progress on the kinetics of radical reactions, using paramagnetic resonance detection methods.

Production and Measurement of Very High Temperatures:
John B. Shumaker

A program of research in the measurement of temperatures above 10,000°C is being conducted in the High Temperature Measurements Laboratory. current interests include the detailed investigation of stable high current density arcs and plasma jets and the measurement of their temperatures and related physical parameters by a variety of spectroscopic techniques.

Figure 5h. A page from the NBS Postdoctoral Research Associateships announcement booklet produced via the card-controlled typewriter from cards processed by JUSTIFY. The cards that produced this page were rearranged and edited where necessary prior to the transformation described in Section 5.4.

Dynamic Measurement of Properties of Gases at Very High Temperatures: C. W. Beckett

Current research includes the investigation of thermodynamic and related properties of gases at high temperatures by dynamic experimental techniques, such as shock tubes, exploding wires, and electrical discharges. Advanced measurement techniques for obtaining both microscopic and macroscopic properties are being explored. The equipment includes an ultra high speed framing camera and a time-resolved spectrometer.

Molecular Energy Levels and Intensities: A. M. Bass

Studies of energy levels of diatomic and small polyatomic molecules as determined from the analysis of spectroscopic data. Facilities include spectroscopic instrumentation permitting observations from 500 A in the vacuum ultraviolet to 50 microns in the infrared. Also available are various sources for exciting spectra, including electric discharges, flames, flash photolysis and flash heating. The observation of radicals and transient molecules produced in low-temperature matrices and the investigation of matrix effects on molecular energy levels is also of interest.

Microwave Spectra of Gaseous Radicals: H. E. Radford

Experiments are performed to determine the hyperfine structure, Zeeman effect and lambdatype doubling of simple diatomic radicals produced in gas mixtures by electric discharges, chemical reactions, triggered explosions, and photolysis. The paramagnetic resonance technique is used mainly, and spectrometers of various types are available for work at 3 Gc, 9 Gc and 24 Gc in magnetic field strengths up to 24 Kg. Optical and radio-frequency apparatus is also available for radio-optical resonance experiments on diatomic molecules, and for measurements of radio spectra in molecular excited states. Recent successful experiments in these two fields of study have been performed on the paramagnetic resonance spectra of OH and SH, and on the rotational perturbation microwave spectrum of excited CN. Other experiments are in progress on the kinetics of radical reactions, using paramagnetic resonance detection methods.

Plasma Physics: C. K. McLane

An experimental and theoretical study of transport processes in the plasma state has as its object a fundamental understanding of the contribution of collective interactions. Facilities available include a magnetically confined arc discharge apparatus for study of the steady state magneto-plasma in fields up to 7000 gauss, and the usual electronic and optical spectroscopic equipment for plasma diagnostics. Equipment for the study of the transient plasma by capacitor discharge techniques is also available.

Production and Measurement of Very High Temperatures: John B. Shumaker

A program of research in the measurement of temperatures above 10,000°C is being conducted in the High Temperature Measurements Laboratory. Current interests include the detailed investigation of stable high current density arcs and plasma jets and the measurement of their temperatures and related physical parameters by a variety of spectroscopic techniques.

Figure 5i. A page from an earlier issue of the NBS Postdoctoral Research Associateships announcement booklet produced prior to the development of JUSTIFY. Note the ragged right hand margins.

```
'D'YNAMIC 'M'EASUREMENT OF 'P'ROPERTIES OF 'S'OLIDS AND 'L'IQUIDS AT ) D
'V'ERY 'H'IGH  'T'EMPERATURES)
'C'. 'W'. 'B'ECKETT AND 'A'. 'C'EZAIRLIYAN)                                C
     'C'URRENT RESEARCH INCLUDES THE INVESTIGATION OF THERMODYNAMIC)
AND RELATED PROPERTIES OF SOLIDS AND LIQUIDS AT HIGH TEMPERATURES BY)
DYNAMIC EXPERIMENTAL TECHNIQUES, SUCH AS, PULSE CALORIMETRY, )
EXPLODING WIRES, ELECTRICAL DISCHARGES, ETC.  'A'DVANCED MEASUREMENT)
TECHNIQUES FOR OBTAINING BOTH MICROSCOPIC AND MACROSCOPIC PROPERTIES)
ARE BEING EXPLORED.  'T'HE EQUIPMENT INCLUDES PULSE CALORIMETERS OF)
MILLISECOND AND MICROSECOND TIME RESOLUTION, HIGH['0']SPEED)
PHOTOELECTRIC PYROMETERS, AND ULTRA HIGH['0']SPEED FRAMING CAMERA, A)
HIGH['0']SPEED DIGITAL RECORDING SYSTEM AND OTHER AUXILIARY DYNAMIC)
MEASURMENT INSTRUMENTS.)
 )
'C'OMBUSTION AND 'R'EACTION 'C'ALORIMETRY)                                  D
'G'. 'T'. 'A'RMSTRONG)                                                     C
 )
     'R'ELATIONSHIPS BETWEEN BINDING ENERGY AND STRUCTURE ARE BEING)
INVESTIGATED AMONG ORGANIC AND INORGANIC COMPOUNDS.  'R'ECENT EMPHASIS)
HAS BEEN ON FLUORINE COMPOUNDS, NITRO COMPOUNDS, AND COMPOUNDS OF)
BIOLOGICAL INTEREST.  'A' ROTATING BOMB CALORIMETER, FACILITIES FOR)
BOMB OR FLOW COLORIMETRY WITH FLUORINE, AND OTHER CALORIMETRIC)
FACILITIES ARE AVAILABLE FOR HIGH PRECISION MEASUREMENTS.  'T'HE)
CALORIMETRIC PROCESS, REFERENCE MATERIALS FOR REACTION CALORIMETRY,)
AND NEW FORMS FOR CORRELATING CALORIMETRIC MEASURMENTS MAY ALSO)
BE INVESTIGATED.)
 )
'P'RODUCTION AND 'M'EASUREMENT OF 'V'ERY 'H'IGH 'T'EMPERATURES'1' )     D
                                 'J'OHN 'B'. 'S'HUMAKER )                 )C
   )
    'A' PROGRAM OF RESEARCH IN THE MEASUREMENT OF TEMPERATURES ABOVE  $
10,000'/C'  IS)
BEING CONDUCTED IN THE 'H'IGH 'T'EMPERATURE 'M'EASUREMENTS  $
'L'ABORATORY.  CURRENT)
INTERESTS INCLUDE THE DETAILED INVESTIGATION OF STABLE HIGH CURRENT $
DENSITY ARCS)
AND PLASMA JETS AND THE MEASUREMENT OF THEIR TEMPERATURES AND  RELATED  $
PHYSICAL)
PARAMETERS BY A VARIETY OF SPECTROSCOPIC TECHNIQUES.) RES AND  RELATE   $
 )
```

Figure 5j. The text shown above represents editorial changes in existing cards prior to conversion via JUSTIFY to produce the character stream shown in the next figure.

```
ABCREFGHIJKLMNOPQDSTUVWXYZ0123456789+.)$*/,($' =          1'          %);
 65  0  1  1  5  7  0
THE WIDTH OF OUTPUT TEXT IS  65


(=)D(/)'YNAMIC 'M'EASUREMENT OF 'P'ROPERTIES OF 'S'OLIDS AND 'L'IQUIDS AT(;'/)
'V'ERY 'H'IGH 'T'EMPERATURES
(=)'C'. 'W'. 'B'ECKETT AND 'A'. 'C'EZAIRLIYAN
     'C'URRENT RESEARCH INCLUDES THE INVESTIGATION OF
THERMODYNAMIC AND RELATED PROPERTIES OF SOLIDS AND LIQUIDS AT
HIGH TEMPERATURES BY DYNAMIC EXPERIMENTAL TECHNIQUES, SUCH AS,
PULSE CALORIMETRY, EXPLODING WIRES, ELECTRICAL DISCHARGES, ETC.
'A'DVANCED MEASUREMENT TECHNIQUES FOR OBTAINING BOTH MICROSCOPIC
AND MACROSCOPIC PROPERTIES ARE BEING EXPLORED.  'T'HE EQUIPMENT
INCLUDES PULSE CALORIMETERS OF MILLISECOND AND MICROSECOND TIME
RESOLUTION, HIGH['0']SPEED PHOTOELECTRIC PYROMETERS, AND ULTRA
HIGH['0']SPEED FRAMING CAMERA, A HIGH['0']SPEED DIGITAL RECORDING
SYSTEM AND OTHER AUXILIARY DYNAMIC MEASURMENT INSTRUMENTS.
)
(=)'C(;/)'OMBUSTION AND 'R'EACTION 'C'ALORIMETRY(;'/)
(=)'G'. 'T'. 'A'RMSTRONG
)
     'R'ELATIONSHIPS BETWEEN BINDING ENERGY AND STRUCTURE ARE
BEING INVESTIGATED AMONG ORGANIC AND INORGANIC COMPOUNDS.
'R'ECENT EMPHASIS HAS BEEN ON FLUORINE COMPOUNDS, NITRO
COMPOUNDS, AND COMPOUNDS OF BIOLOGICAL INTEREST.  'A' ROTATING
BOMB CALORIMETER, FACILITIES FOR BOMB OR FLOW COLORIMETRY WITH
FLUORINE, AND OTHER CALORIMETRIC FACILITIES ARE AVAILABLE FOR
HIGH PRECISION MEASUREMENTS.  'T'HE CALORIMETRIC PROCESS,
REFERENCE MATERIALS FOR REACTION CALORIMETRY, AND NEW FORMS FOR
CORRELATING CALORIMETRIC MEASURMENTS MAY ALSO BE INVESTIGATED.
)
(=)P(;/)'RODUCTION AND 'M'EASUREMENT OF 'V'ERY 'H'IGH 'T'EMPERATURES'1'(;'/)
(=)'J'OHN 'B'. 'S'HUMAKER
)
     'A' PROGRAM OF RESEARCH IN THE MEASUREMENT OF TEMPERATURES
ABOVE 10,000'/C' IS BEING CONDUCTED IN THE 'H'IGH 'T'EMPERATURE
'M'EASUREMENTS 'L'ABORATORY.  CURRENT INTERESTS INCLUDE THE
DETAILED INVESTIGATION OF STABLE HIGH CURRENT DENSITY ARCS AND
PLASMA JETS AND THE MEASUREMENT OF THEIR TEMPERATURES AND RELATED
PHYSICAL PARAMETERS BY A VARIETY OF SPECTROSCOPIC TECHNIQUES.
)
(=)L(;/)'OW 'T'EMPERATURE 'C'ALORIMETRY'1'(;'/)
(=)'G'. 'T'. 'F'URUKAWA
)
     'C'URRENT RESEARCH INTERESTS INCLUDE THE INVESTIGATIONS OF
HEAT CAPACITY,VAPOR PRESSURE, HEATS OF FUSION AND TRANSITION,
HEAT OF VAPORIZATION, CHEMICAL PURITY, AND PHASE EQUILIBRIA OF
SELECTED COMPOUNDS OF FLUORINE, HYDROGEN AND OTHER LIGHT
```

Figure 5k. Results from a pass through a slightly modified ad hoc version of JUSTIFY preparatory to the production of the typset page shown in Figure 5g. See Section 5.4 for a discussion of the circled characters.

APPENDIX

PROGRAM LISTINGS

The program listings which follow were produced on a photocomposing machine at the Government Printing Office from a magnetic tape produced at NBS by a Fortran program called TYPSET. That program accepts a symbolic program deck as input data and produces a magnetic tape formatted to be processed by an Autoset Composition Program or the Master Typography Program at the Government Printing Office.

The listings were reduced 10% from paper positives set in 10 point type with 12 point leading in the Clarinda typeface.

A magnetic tape containing card images of the programs in this appendix will be prepared for sale by the Clearinghouse for Federal Scientific and Technical Information if the demand warrants it. A self-addressed card is provided for this purpose at the back of this report.

```
C                                SCRAMBLE                                    SCRA0010
C                                                                            SCRA0020
C  ITAPE - THE NORMAL SYSTEM INPUT TAPE.                                     SCRA0030
C IOTAPE - THE NORMAL SYSTEM OUTPUT TAPE.                                    SCRA0040
C IRTAPE - A SPECIAL INPUT TAPE. IF ONE INPUT TAPE WILL DO, SET IRTAPE       SCRA0050
C          EQUAL TO ITAPE                                                    SCRA0060
C IPTAPE - A SPECIAL OUTPUT TAPE. IT CAN BE USED AS AN INPUT TO OTHER        SCRA0070
C          PROGRAMS, BUT CANNOT BE EQUATED WITH IOTAPE AS IOTAPE, IN         SCRA0080
C          ADDITION TO NORMAL OUTPUT, CONTAINS OTHER PROGRAM MESSAGES        SCRA0090
C          AND ERROR REMARKS.                                                SCRA0100
C THE FIRST CARD CONTAINS THOSE CHARACTERS FOR WHICH SUBSTITUTIONS ARE       SCRA0110
C TO BE MADE. THE SECOND CARD CONTAINS THE CHARACTERS TO BE SUBSTITUTED      SCRA0120
C FOR THE CHARACTERS IN THE CORRESPONDING FIELDS OF FIRST CONTROL CARD.      SCRA0130
C THE THIRD CONTROL CARD CONTAINS IN I2 FORMAT THE PUNCH SWITCH ITEST        SCRA0140
C WHICH IS NONZERO TO PUNCH AND IRTAPE AND IPTAPE                            SCRA0150
C                                                                            SCRA0155
C  CODE BY MRS CARLA G. MESSINA       NSRDS - NBS       1967                 SCRAM160
C                                                                            SCRA0165
      DIMENSION ICOL(140),IA(80),IB(80),IS(80)                               SCRAM170
      ITAPE=5                                                                SCRAM180
      IOTAPE=6                                                               SCRAM190
10    FORMAT (132A1)                                                         SCRAM200
      READ (ITAPE,10) (IA(J),J=1,80)                                         SCRAM210
      WRITE (IOTAPE,75) (IA(J),J=1,80)                                       SCRAM220
      READ (ITAPE,10) (IB(J),J=1,80)                                         SCRAM230
      WRITE (IOTAPE,75) (IB(J),J=1,80)                                       SCRAM240
      READ (ITAPE,20) ITEST,IRTAPE,IPTAPE,IWIDE1                             SCRAM250
20    FORMAT (3I2,1I4)                                                       SCRAM260
      IF (IWIDE1)      21,21,22                                              SCRAM261
21    IWIDE1=80                                                              SCRAM262
22    IF (IWIDE1-132)   23,23,21                                             SCRAM263
23    IF (IRTAPE) 11,11,12                                                   SCRAM270
11    IRTAPE = 5                                                             SCRAM280
      GO TO 14                                                               SCRAM290
12    IF (IRTAPE -6) 14,14,13                                                SCRAM300
13    REWIND IRTAPE                                                          SCRAM310
14    IF (IPTAPE) 15,15,16                                                   SCRAM320
15    IPTAPE = 3                                                             SCRAM330
      GO TO 18                                                               SCRAM340
16    IF (IPTAPE - 6) 18,18,17                                               SCRA0350
17    REWIND IPTAPE                                                          SCRA0360
18    WRITE (IOTAPE,39) ITEST,IRTAPE,IPTAPE                                  SCRA0370
39    FORMAT (1X,3I2)                                                        SCRA0380
      DO 40 I=1,80                                                           SCRA0390
      IF (IA(I) - IB(I)) 25,30,25                                            SCRA0400
25    IS(I)=1                                                                SCRA0410
       GO TO 40                                                              SCRA0420
30    IS(I)=0                                                                SCRA0430
40    CONTINUE                                                               SCRA0440
      DO 60 I=1,80                                                           SCRA0450
          M=I                                                                SCRA0460
      IF (IS(I)) 90,60,90                                                    SCRA0470
60    CONTINUE                                                               SCRA0480
      IF (M-80) 90,70,70                                                     SCRA0490
```

```
70    WRITE (IOTAPE,75) (IA(J),J=1,80)                                  SCRA0500
      WRITE (IOTAPE,75) (IB(J),J=1,80)                                  SCRA0510
75     FORMAT( 1X,131A1)                                                SCRA0520
      WRITE (IOTAPE,8)                                                  SCRA0530
8     FORMAT(72HOTHE ABOVE CARDS ARE IDENTICAL AND THEREFORE NO SUBSTITUSCRA0540
     1TION IS NEEDED.        )                                          SCRA0550
1111   STOP                                                             SCRA0560
90    DO 100 I=1,80                                                     SCRA0570
      N=81-I                                                            SCRA0580
      IF (IS(N)) 50,100,50                                              SCRA0590
100     CONTINUE                                                        SCRA0600
50    READ (IRTAPE,10) (ICOL(J),J=1,IWIDE1)                             SCRA0610
      CALL CHECK(IA,ICOL,K,ITEST,IPTAPE,IOTAPE)                         SCRA0620
       IF (K) 1111,110 ,1111                                            SCRA0630
110   DO 150 I=1,IWIDE1                                                 SCRA0640
      DO 130 J=M,N                                                      SCRA0650
      IF (IS(J)) 120,130,120                                            SCRA0660
120   IF (ICOL(I) - IA(J)) 130,140,130                                  SCRA0670
130    CONTINUE                                                         SCRA0680
      GO TO 150                                                         SCRA0690
140   ICOL(I) = IB(J)                                                   SCRA0700
150   CONTINUE                                                          SCRA0710
      WRITE (IOTAPE,75) (ICOL(J),J=1,IWIDE1)                            SCRA0720
      IF (ITEST) 160,50,160                                             SCRA0730
160    WRITE (IPTAPE,10) (ICOL(J),J=1,IWIDE1)                           SCRA0740
      GO TO 50                                                          SCRA0750
       END                                                              SCRA0760
C
      SUBROUTINE CHECK(IA,II,K,ITEST,IPTAPE,IOTAPE)                     SCCK0010
      DIMENSION IA(80),II(140)                                          SCCK0020
      DO 10 I=1,26                                                      SCCK0030
      IF (IA(I) - II(I)) 50,10,50                                       SCCK0040
10    CONTINUE                                                          SCCK0050
      K=1                                                               SCCK0060
      WRITE (IOTAPE,19) (IA(I),I=1,80)                                  SCCK0070
      IF (ITEST) 20,60,20                                               SCCK0080
20    WRITE (IPTAPE,29) (IA(I),I=1,80)                                  SCCK0090
      IF (IPTAPE -6) 60,60,30                                           SCCK0100
30    END FILE IPTAPE                                                   SCCK0110
      GO TO 60                                                          SCCK0120
50    K=0                                                               SCCK0130
60    RETURN                                                            SCCK0140
19    FORMAT (1X,80A1)                                                  SCCK0150
29    FORMAT (80A1)                                                     SCCK0160
      END                                                               SCCK0170


C                          SUBSTITUTE                                   SUB  10
C                                                                       SUB  20
C     TEXTUAL SUBSTITUTION PROGRAM WRITTEN BY C. MESSINA FOR IBM 7094   SUB  30
C     RECOMPILED NOV 1966 FOR USE ON CDC 3100, MAY 1967 FOR UNIVAC 1108 SUB  40
C     THE PROGRAM SUBSTITUTES AN ITEM OF TEXT WHENEVER IT OCCURS IN     SUB  50
C     THE SET OF RECORDS IN THE LIBRARY, PUTTING IN A NEW PRESCRIBED    SUB  60
C     ITEM.  THE PROGRAM IS DESIGNED TO HANDLE, BUT IS NOT LIMITED      SUB  70
C     TO, IBM 870 DOCUMENT-WRITER DRIVER RECORDS.                       SUB  80
```

```
C     THE INPUT DECK AT OBJECT TIME IS THE FOLLOWING SET OF CARDS       SUB  90
C     THE FIRST CARD IS A DICTIONARY OF THE ALPHABET STARTING WITH THE  SUB 100
C     LETTER A IN CARD COL ONE, A LETTER B IN COL 2 AND SO FORTH. THE NOSUB 110
C     FOLLOW THE ALPHABET STARTING WITH ZERO.  COL 38 CONTAINS THE PRINTSUB 120
C     OUT STRING DELIMITER.  COL 47 CONTAINS A BLANK.                   SUB 130
C     THE SECOND CARD HAS A ZERO IN COL TWO IF NO CARDS ARE TO BE PUNCHESUB 140
C     1 IF THE PUNCH TAPE IS TO BE WRITTEN. THE NEXT 2I2 FIELDS ON THIS SUB 150
C     CARD IF NON BLANK CONTAIN THE IRTAPE NUMBER AND THE IPTAPE NO.    SUB 160
C     IF IRTAPE OR IPTAPE ARE GREATER THAN 6 THEY WILL REWIND AT START  SUB 170
C     OF RUN. UNLESS CHANGED ON THE SECOND CARD IRTAPE=5 AND IPTAPE=3   SUB 180
C     THE THIRD AND FOURTH CARDS, BOTH IN A1,I3,2A1,I2 FORMATS CONTROL ISUB 190
C     AND OUTPUT RECORD  FORMATS RESPECTIVELY.  THE FIRST ITEM IS THE   SUB 200
C     CONTINUATION CARD SYMBOL, THE SECOND IS THE RECORD LENGTH, THE    SUB 210
C     THIRD IS THE SHIFT TO UPPER CASE SYMBOL, THE FOURTH IS THE SHIFT  SUB 220
C     TO LOWER CASE SYMBOL,  AND THE FIFTH IS THE SHIFT AND LOCK SWITCH SUB 230
C     THAT IS 0 IF THE MODE IS  NOT SHIFT AND LOCK AND 1 IF IT IS.      SUB 240
C     CARDS FIVE ET SEQ CONTAIN THE LIST OF STRINGS TO BE EXCHANGED.    SUB 250
C     ON EACH CARD THE OLD RECORD OR STRING APPEARS ON THE LEFT SIDE ANDSUB 260
C     THE NEW STRING ON THE RIGHT.  THE CHARACTER WHICH APPEARS IN CARD SUB 270
C     COLUMN 1 IS THE STRING DELIMITER WHICH REMAINS IN FORCE FOR THAT  SUB 280
C     CARD.  IT MAY, HOWEVER, CHANGE FROM CARD TO CARD.                 SUB 290
C     THE  FORMAT IS PRESCRIBED.  A CHARACTER IN COL 1 DEFINES THE STARTSUB 300
C     OF A STRING.  THE SAME CHARACTER MUST APPEAR AFTER THE END        SUB 310
C     OF THE STRING.  THE THIRD APPEARANCE OF THE COLUMN 1 CHARACTER ON SUB 320
C     THE CARD STARTS THE 2ND STRING AND THE FOURTH APPEARANCE ENDS IT. SUB 330
C                                                                       SUB 340
C        EXAMPLE        /REAL/    /TRUE/                              SUB 350
C                                                                       SUB 360
C     AFTER THE SUBSTITUTION LIST MUST COME A CARD WITH THE WORD  FINIS SUB 370
C     STARTING IN CARD COLUMN ONE.                                      SUB 380
C     AT THIS POINT THE PROGRAM STARTS TO READ THE LIBRARY RECORDS FROM  SUB 390
C     THE UNIT CALLED IRTAPE.  THE OTHER UNITS ARE          INPUT DECK  SUB 400
C     AND SEARCH LIST FROM UNIT ITAPE, PRINTER OUTPUT TO UNIT IOTAPE ANDSUB 410
C     PUNCH TO UNIT IPTAPE.  IPTAPE CONTAINS THE SAME INFORMATION AS    SUB 420
C     UNIT IOTAPE EXCLUDING PROGRAM MESSAGES, I.E. ONLY TEXT.           SUB 430
C     INPUT AND OUTPUT RECORD LENGTHS MUST BE AT LEAST 1 CHARACTER LONG SUB 440
C     NO MORE THAN 132 CHARACTERS LONG.                                 SUB 450
C                                                                       SUB 460
      DIMENSION IA(86),N(400),IC(8000),IB(361)                          SUB 470
      COMMON ITAPE,IOTAPE,IRTAPE, IPTAPE,IWIDE1,IWIDE2,IA,IW,N,IC,IB    SUB 480
      ITAPE=5                                                           SUB 490
      IOTAPE=6                                                          SUB 500
      IEND=0                                                            SUB 510
      READ (ITAPE,1060) (IA(J),J=1,80)                                  SUB 520
      WRITE (IOTAPE,1110) (IA(J),J=1,80)                                SUB 530
      READ (ITAPE,1070) ITEST,IRTAPE,IPTAPE                             SUB 540
      IF (IRTAPE) 20,20,30                                              SUB 550
20    IRTAPE=5                                                          SUB 560
      GO TO 50                                                          SUB 570
30    IF (IRTAPE-6) 50,50,40                                            SUB 580
40    REWIND IRTAPE                                                     SUB 590
50    IF (IPTAPE) 60,60,70                                              SUB 600
60    IPTAPE=3                                                          SUB 610
      GO TO 90                                                          SUB 620
70    IF (IPTAPE-6) 90,90,80                                            SUB 630
```

```
80    REWIND IPTAPE                                                    SUB 640
90    WRITE (IOTAPE,1150) ITEST,IRTAPE,IPTAPE                          SUB 650
      READ (ITAPE,1080) IA(81),IWIDE1,IA(83),IA(85),LOCK1              SUB 660
      WRITE (IOTAPE,1140) IA(81),IWIDE1,IA(83),IA(85),LOCK1            SUB 670
      READ (ITAPE,1080) IA(82),IWIDE2,IA(84),IA(86),LOCK2              SUB 680
      WRITE (IOTAPE,1140) IA(82),IWIDE2,IA(84),IA(86),LOCK2            SUB 690
      IF (IWIDE1-1) 100,110,110                                        SUB 700
100   WRITE (IOTAPE,1090)                                              SUB 710
      GO TO 1050                                                       SUB 720
110   IF (IWIDE1-132) 120,120,100                                      SUB 730
120   IF (IWIDE2-1) 100,130,130                                        SUB 740
130   IF (IWIDE2-132) 140,140,100                                      SUB 750
140   N1=0                                                             SUB 760
      N3=1                                                             SUB 770
C        START OF READING IN SUBSTITUTE LISTS                          SUB 780
150   READ (ITAPE,1060) (IB(J),J=1,80)                                 SUB 790
      N2=0                                                             SUB 800
      N22=0                                                            SUB 810
      IF (IB(1)-IA(6)) 200,160,200                                     SUB 820
160   IF (IB(2)-IA(9)) 200,170,200                                     SUB 830
170   IF (IB(3)-IA(14)) 200,180,200                                    SUB 840
180   IF (IB(4)-IA(9)) 200,190,200                                     SUB 850
190   IF (IB(5)-IA(19)) 200,430,200                                    SUB 860
200   DO 220 I=2,78                                                    SUB 870
      IF (IB(I)-IB(1)) 220,210,220                                     SUB 880
210   IF (N2) 150,150,230                                              SUB 890
220   N2=I-1                                                           SUB 900
230   J=N2+3                                                           SUB 910
      IF (J-79) 250,240,240                                            SUB 920
240   WRITE (IOTAPE,1120) IB(1),(IB(I),I=1,80)                         SUB 930
      GO TO 1050                                                       SUB 940
250   K=J+1                                                            SUB 950
      DO 260 I=J,79                                                    SUB 960
      IF (IB(I)-IB(1)) 260,270,260                                     SUB 970
260   K=I+2                                                            SUB 980
      GO TO 240                                                        SUB 990
270   DO 280 I=K,80                                                    SUB1000
      IF (IB(I)-IB(1)) 280,290,280                                     SUB1010
280   N22=I-K+1                                                        SUB1020
      GO TO 240                                                        SUB1030
290   N1=N1+2                                                          SUB1040
      N(N1-1)=N2                                                       SUB1050
      N(N1)=N22                                                        SUB1060
      N4=N3+N2-1                                                       SUB1070
      IF (N4-7920) 310,310,300                                         SUB1080
300   WRITE (IOTAPE,1100) N4,N1                                        SUB1090
      GO TO 1050                                                       SUB1100
310   IF (N1-400) 320,320,300                                          SUB1110
320   J=2                                                              SUB1120
      DO 330 I=N3,N4                                                   SUB1130
      IC(I)=IB(J)                                                      SUB1140
330   J=J+1                                                            SUB1150
      N3=N3+N2                                                         SUB1160
      IF (N22) 360,360,340                                             SUB1170
340   N4=N3+N22-1                                                      SUB1180
```

```
      J=K                                                          SUB1190
      DO 350 I=N3,N4                                               SUB1200
      IC(I)=IB(J)                                                  SUB1210
350   J=J+1                                                        SUB1220
360   IF (N2-38) 370,370,380                                       SUB1230
370   K=42                                                         SUB1240
380   K1=K+N22-1                                                   SUB1250
      J=N2+3                                                       SUB1260
      DO 390 L=J,80                                                SUB1270
390   IB(L)=IA(47)                                                 SUB1280
      IB(1)=IA(38)                                                 SUB1290
      IB(N2+2)=IA(38)                                              SUB1300
      IB(K-1)=IA(38)                                               SUB1310
      IB(K1+1)=IA(38)                                              SUB1320
      IF (N22) 420,420,400                                         SUB1330
400   DO 410 I=N3,N4                                               SUB1340
      IB(K)=IC(I)                                                  SUB1350
410   K=K+1                                                        SUB1360
      N3=N3+N22                                                    SUB1370
420   WRITE (IOTAPE,1110) (IB(J),J=1,80)                           SUB1380
      GO TO 150                                                    SUB1390
430   N44=N4                                                       SUB1400
      IF (N1-4) 540,440,440                                        SUB1410
440   N7=N1+2                                                      SUB1420
450   N3=1                                                         SUB1430
      K1=0                                                         SUB1440
      N7=N7-2                                                      SUB1450
      IF (N7-4) 460,470,470                                        SUB1460
460   N7=N7+2                                                      SUB1470
470   DO 530 I=4,N7,2                                              SUB1480
      N2=N(I-3)+N(I-2)                                             SUB1490
      N22=N(I-1)+N(I)                                              SUB1500
      IF (N(I-3)-N(I-1)) 490,480,480                               SUB1510
480   N3=N3+N2                                                     SUB1520
      GO TO 530                                                    SUB1530
490   N4=N(I-3)                                                    SUB1540
      N(I-3)=N(I-1)                                                SUB1550
      N(I-1)=N4                                                    SUB1560
      N4=N(I)                                                      SUB1570
      N(I)=N(I-2)                                                  SUB1580
      N(I-2)=N4                                                    SUB1590
      K1=K1+1                                                      SUB1600
      N4=N3+N2-1                                                   SUB1610
      K=0                                                          SUB1620
      DO 500 J=N3,N4                                               SUB1630
      K=K+1                                                        SUB1640
500   IB(K)=IC(J)                                                  SUB1650
      DO 510 J=1,N22                                               SUB1660
      K=N3+J-1                                                     SUB1670
      N6=N4+J                                                      SUB1680
510   IC(K)=IC(N6)                                                 SUB1690
      N3=N3+N22                                                    SUB1700
      DO 520 J=1,N2                                                SUB1710
      K=N3+J-1                                                     SUB1720
520   IC(K)=IB(J)                                                  SUB1730
```

```
530   CONTINUE                                                          SUB1740
      IF (K1) 540,540,450                                               SUB1750
540   WRITE (IOTAPE,1110) IA(6),IA(9),IA(14),IA(9),IA(19)               SUB1760
C        START OF SUBSTITUTION                                          SUB1770
550   IF (IEND) 570,560,570                                             SUB1780
560   IB(IWIDE1+1)=IA(47)                                               SUB1790
      READ (IRTAPE,1060) (IB(J),J=1,IWIDE1)                             SUB1800
570   CALL CHECK2 (IEND,ITEST,1)                                        SUB1810
      IF (IEND) 580,590,580                                             SUB1820
580   STOP                                                              SUB1830
590   N3=1                                                              SUB1840
      DO 600 I=1,IWIDE1                                                 SUB1850
      IF (IA(81)-IB(I)) 600,630,600                                     SUB1860
600   CONTINUE                                                          SUB1870
      I=IWIDE1                                                          SUB1880
      IF (IB(IWIDE1)-IA(47)) 700,610,700                                SUB1890
610   DO 620 J=1,IWIDE1                                                 SUB1900
      IF (IB(I)-IA(47)) 690,620,690                                     SUB1910
620   I=I-1                                                             SUB1920
      GO TO 690                                                         SUB1930
630   J=I+IWIDE1-1                                                      SUB1940
      IF (J-360) 640,640,700                                            SUB1950
640   K=I                                                               SUB1960
      READ (IRTAPE,1060) (IB(I),I=K,J)                                  SUB1970
      IEND=2                                                            SUB1980
      CALL CHECK2 (IEND,ITEST,K)                                        SUB1990
      IF (IEND) 650,660,650                                             SUB2000
650   I=K-1                                                             SUB2010
      GO TO 700                                                         SUB2020
660   DO 670 I=K,J                                                      SUB2030
      IF (IA(81)-IB(I)) 670,630,670                                     SUB2040
670   CONTINUE                                                          SUB2050
      I=J                                                               SUB2060
      DO 680 L=K,J                                                      SUB2070
      IF (IB(I)-IA(47)) 700,680,700                                     SUB2080
680   I=I-1                                                             SUB2090
690   I=I+1                                                             SUB2100
700   IW=I                                                              SUB2110
      IB(IW+1)=IA(47)                                                   SUB2120
      IF (LOCK1) 710,720,710                                            SUB2130
710   CALL SUNLOK                                                       SUB2140
720   K2=1                                                              SUB2150
      ILK=0                                                             SUB2160
730   N6=0                                                              SUB2170
      N7=0                                                              SUB2180
      DO 760 K=K2,IW                                                    SUB2190
      N3=1                                                              SUB2200
      DO 750 I=2,N1,2                                                   SUB2210
      N2=N(I-1)                                                         SUB2220
      N22=N(I)                                                          SUB2230
      K1=K                                                              SUB2240
      N4=N3+N2-1                                                        SUB2250
      DO 740 J=N3,N4                                                    SUB2260
      IF (IB(K1)-IC(J)) 750,740,750                                     SUB2270
740   K1=K1+1                                                           SUB2280
```

```
      K1=K1-1                                                          SUB2290
      N7=I                                                             SUB2300
      N6=N3                                                            SUB2310
      GO TO 770                                                        SUB2320
750   N3=N3+N2+N22                                                     SUB2330
760   CONTINUE                                                         SUB2340
      GO TO 910                                                        SUB2350
770   IF (K1-IW) 790,790,780                                           SUB2360
780   IW=K1                                                            SUB2370
      ILK=1                                                            SUB2380
      IF (IW-360) 790,790,900                                          SUB2390
790   K1=K                                                             SUB2400
      N2=N(N7-1)                                                       SUB2410
      N22=N(N7)                                                        SUB2420
      N3=N6+N2                                                         SUB2430
      N4=N3+N22-1                                                      SUB2440
      N5=N22-N2                                                        SUB2450
      IF (N5) 870,800,840                                              SUB2460
800   DO 810 J=N3,N4                                                   SUB2470
      IB(K1)=IC(J)                                                     SUB2480
810   K1=K1+1                                                          SUB2490
820   IF (ILK) 830,830,910                                             SUB2500
830   K2=K1                                                            SUB2510
      GO TO 730                                                        SUB2520
840   IF (IW+N5-360) 850,850,900                                       SUB2530
850   IW=IW+N5                                                         SUB2540
      K2=IW                                                            SUB2550
      DO 860 J=K1,IW                                                   SUB2560
      K9=K2-N5                                                         SUB2570
      IB(K2)=IB(K9)                                                    SUB2580
860   K2=K2-1                                                          SUB2590
      GO TO 800                                                        SUB2600
870   DO 880 J=K1,IW                                                   SUB2610
      K9=J-N5                                                          SUB2620
880   IB(J)=IB(K9)                                                     SUB2630
      K9=IW+N5+1                                                       SUB2640
      DO 890 J=K9,IW                                                   SUB2650
890   IB(J)=IA(47)                                                     SUB2660
      IW=IW+N5                                                         SUB2670
      IF (N22) 820,820,800                                             SUB2680
900   WRITE (IOTAPE,1130)                                              SUB2690
910   IF (LOCK2) 920,930,920                                           SUB2700
920   CALL SLOCK                                                       SUB2710
C        PRINT OUT INSTRUCTIONS WITH THE CONDITION THAT WHEN A LINE IS  SUB2720
C     BROKEN INTO TWO OR MORE CARDS  THE NEW CARD WILL ALWAYS START WITHSUB2730
C     THE BEGINNING OF A WORD.                                         SUB2740
930   K=1                                                              SUB2750
      K1=IWIDE2-1                                                      SUB2760
      IF (IW-K1-1) 940,940,970                                         SUB2770
940   WRITE (IOTAPE,1110).(IB(J),J=K,IW)                               SUB2780
      IF (ITEST) 950,550,950                                           SUB2790
950   WRITE (IPTAPE,1060) (IB(J),J=K,IW)                               SUB2800
      GO TO 550                                                        SUB2810
960   K=K1+1                                                           SUB2820
      K1=K1+IWIDE2-1                                                   SUB2830
```

```
      IF (IW-K1-1) 940,940,970                                          SUB2840
970   J=K1                                                              SUB2850
      DO 980 I=K,K1                                                     SUB2860
      IF (IB(J)-IA(47)) 980,1010,980                                    SUB2870
980   J=J-1                                                             SUB2880
990   WRITE (IOTAPE,1110) (IB(J),J=K,K1),IA(82)                         SUB2890
      IF (ITEST) 1000,960,1000                                          SUB2900
1000  WRITE (IPTAPE,1060) (IB(J),J=K,K1),IA(82)                         SUB2910
      GO TO 960                                                         SUB2920
1010  K1=J                                                              SUB2930
      IF (IB(K1+1)-IA(47)) 990,1020,990                                 SUB2940
1020  DO 1030 I=K,K1                                                    SUB2950
      IF (IB(J)-IA(47)) 1040,1030,1040                                  SUB2960
1030  J=J-1                                                             SUB2970
      GO TO 990                                                         SUB2980
1040  K1=J                                                              SUB2990
      GO TO 970                                                         SUB3000
1050  STOP                                                              SUB3010
C                                                                       SUB3020
1060  FORMAT (132A1)                                                    SUB3030
1070  FORMAT (40I2)                                                     SUB3040
1080  FORMAT (1A1,1I3,2A1,1I2)                                          SUB3050
1090  FORMAT (70H INPUT OR OUTPUT IMAGE WIDTH CANNOT BE LESS THAT 1 OR GSUB3060
     1REATER THAN 132/6H STOP.)                                         SUB3070
1100  FORMAT (33H LIST OF REPLACEMENTS IS TOO LONG/67H MAXIMUM CHARACTERSUB3080
     1 LENGTH IS 8000, MAXIMUM NUMBER OF PHRASES IS 400/20H CURRENT VALUSUB3090
     2ES ARE ,2I6,6H STOP.)                                             SUB3100
1110  FORMAT (1X,131A1)                                                 SUB3110
1120  FORMAT (16H THE CHARACTER  ,1A1,48H DID NOT APPEAR 4 TIMES ON THE SUB3120
     1CARD BELOW. STOP./1X,80A1)                                        SUB3130
1130  FORMAT (86HOTHE LINE FOLLOWING WOULD HAVE EXCEEDED 360 CHARACTERS SUB3140
     1IF SUBSTITUTION HAD CONTINUED. )                                  SUB3150
1140  FORMAT (1X,1A1,1I3,2A1,1I2)                                       SUB3160
1150  FORMAT (1X,50I2)                                                  SUB3170
      END                                                               SUB3180-
C
C
      SUBROUTINE SUNLOK                                                 SULK 10
      DIMENSION IA(86), N(400), IC(8000), IB(361)                       SULK 20
      COMMON ITAPE,IOTAPE,IRTAPE,IPTAPE,IWIDE1,IWIDE2,IA,IW,N,IC,IB     SULK 30
      L=0                                                               SULK 40
      J=0                                                               SULK 50
      K=0                                                               SULK 60
      DO 60 I=1,IW                                                      SULK 70
      IF (IB(I)-IA(83)) 40,20,40                                        SULK 80
20    L=L+1                                                             SULK 90
      IF (L-J-1) 30,60,30                                               SULK100
30    K=1                                                               SULK110
      GO TO 60                                                          SULK120
40    IF (IB(I)-IA(85)) 60,50,60                                        SULK130
50    J=J+1                                                             SULK140
      IF (L-J) 30,60,30                                                 SULK150
60    CONTINUE                                                          SULK160
      IF (L-J) 80,70,90                                                 SULK170
70    IF (K) 80,120,80                                                  SULK180
```

```
80    WRITE (IOTAPE,280)                                                    SULK190
      GO TO 150                                                             SULK200
90    IF (IA(83)-IA(85)) 80,100,80                                          SULK210
100   K=2*(L/2)-L                                                           SULK220
      IF (K) 110,120,110                                                    SULK230
110   IW=IW+1                                                               SULK240
      IB(IW)=IA(85)                                                         SULK250
120   J=1                                                                   SULK260
130   IF (IB(J)-IA(83)) 140,160,140                                         SULK270
140   J=J+1                                                                 SULK280
      IF (J-(IW+1)) 130,150,150                                             SULK290
150   RETURN                                                                SULK300
160   IF (IB(J+1)-IA(85)) 190,170,190                                       SULK310
170   J=J+2                                                                 SULK320
      DO 180 I=J,IW                                                         SULK330
180   IB(I-1)=IB(I)                                                         SULK340
      GO TO 220                                                             SULK350
190   IF (IB(J+2)-IA(85)) 230,200,230                                       SULK360
200   J=J+3                                                                 SULK370
      DO 210 I=J,IW                                                         SULK380
210   IB(I-1)=IB(I)                                                         SULK390
220   IW=IW-1                                                               SULK400
      IB(IW+1)=IA(47)                                                       SULK410
      J=J-1                                                                 SULK420
      GO TO 130                                                             SULK430
230   IF ((IW+1)-360) 250,250,240                                           SULK440
240   WRITE (IOTAPE,270)                                                    SULK450
      GO TO 150                                                             SULK460
250   IW=IW+1                                                               SULK470
      J=J+3                                                                 SULK480
      K=IW                                                                  SULK490
      DO 260 L=J,IW                                                         SULK500
      IB(K)=IB(K-1)                                                         SULK510
260   K=K-1                                                                 SULK520
      J=J-1                                                                 SULK530
       B(R)= A( 3)                                                          SULK540
      GO TO 160                                                             SULK550
C                                                                           SULK560
270   FORMAT (116HOTHE WORK ON THE FOLLOWING LINE WAS HALTED JUST BEFORESULK570
     1 THE 360 CHARACTER LINE LIMIT WAS EXCEEDED IN SHIFT AND UNLOCK)      SULK580
280   FORMAT (69HOTHE FOLLOWING LINE DID NOT CONTAIN A BALANCED SET OF SSULK590
     1HIFT SYMBOLS. )                                                      SULK600
      END                                                                   SULK610-
C
      SUBROUTINE SLOCK                                                      SUUC 10
      DIMENSION IA(86), N(400), IC(8000), IB(361)                           SUUC 20
      COMMON ITAPE,IOTAPE,IRTAPE,IPTAPE,IWIDE1,IWIDE2,IA,IW,N,IC,IB         SUUC 30
      J=1                                                                   SUUC 40
20    IF (IB(J)-IA(84)) 30,60,30                                            SUUC 50
30    J=J+1                                                                 SUUC 60
      IF (J-(IW+1)) 20,20,50                                                SUUC 70
40    WRITE (IOTAPE,140)                                                    SUUC 80
50    RETURN                                                                SUUC 90
60    J=J+2                                                                SUUC100
70    IF (IB(J)-IA(84)) 110,80,110                                         SUUC110
```

```
80    IW=IW-1                                                          SUUC120
      DO 90 K=J,IW                                                     SUUC130
90    IB(K)=IB(K+1)                                                    SUUC140
      IB(IW+1)=IA(47)                                                  SUUC150
      J=J+1                                                            SUUC160
      IF (J-IW) 70,70,100                                              SUUC170
100   IW=IW+1                                                          SUUC180
      IB(IW)=IA(86)                                                    SUUC190
      GO TO 50                                                         SUUC200
110   IF (IW-360) 120,100,40                                           SUUC210
120   IW=IW+1                                                          SUUC220
      J=J+1                                                            SUUC230
      K=IW                                                             SUUC240
      DO 130 L=J,IW                                                    SUUC250
      IB(K)=IB(K-1)                                                    SUUC260
130   K=K-1                                                            SUUC270
      IB(J-1)=IA(86)                                                   SUUC280
      GO TO 20                                                         SUUC290
C                                                                      SUUC300
140   FORMAT (116HOTHE WORK ON THE FOLLOWING LINE WAS HALTED JUST BEFORESUUC310
     1 THE 360 CHARACTER LINE LIMIT WAS EXCEEDED IN SHIFT AND LOCK  )  SUUC320
      END                                                              SUUC330-
C
      SUBROUTINE CHECK2 (K,ITEST,J)                                    SUCK 10
      DIMENSION IA(86), N(400), IC(8000), IB(361)                      SUCK 20
      COMMON ITAPE,IOTAPE,IRTAPE,IPTAPE,IWIDE1,IWIDE2,IA,IW,N,IC,IB    SUCK 30
      IF (K-1) 20,40,20                                                SUCK 40
20    L=J-1                                                            SUCK 50
      DO 30 I=1,26                                                     SUCK 60
      L=L+1                                                            SUCK 70
      IF (IA(I)-IB(L)) 70,30,70                                        SUCK 80
30    CONTINUE                                                         SUCK 90
      IF (K-2) 40,90,90                                                SUCK100
40    K=1                                                              SUCK110
      WRITE (IOTAPE,100) (IA(I),I=1,80)                                SUCK120
      IF (ITEST) 50,80,50                                              SUCK130
50    WRITE (IPTAPE,110) (IA(I),I=1,80)                                SUCK140
      IF (IPTAPE-6) 80,80,60                                           SUCK150
60    END FILE IPTAPE                                                  SUCK160
      GO TO 80                                                         SUCK170
70    K=0                                                              SUCK180
80    RETURN                                                           SUCK190
90    K=1                                                              SUCK200
      GO TO 80                                                         SUCK210
C                                                                      SUCK220
100   FORMAT (1X,80A1)                                                 SUCK230
110   FORMAT (80A1)                                                    SUCK240
      END                                                              SUCK250
C
      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
      N2=N(I-1)                                               SUBS187
      N22=N(I)                                                SUBS188
      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+N2+N22                                            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   IF (ILK) 720,720,800                                    SUBS215
720   K2=K1                                                   SUBS216
      GO TO 620                                               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
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 (40HOTHE 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-
C
C
      SUBROUTINE SUNLK(IA,IB,IW,IOTAPE)
      DIMENSION IA(86),IB(999)
      MAXIW=998
      L=0
      J=0
      K=0
      DO 60 I=1,IW
      IF (IB(I)-IA(83)) 40,20,40
20    L=L+1
      IF (L-J-1) 30,60,30
30    K=1
      GO TO 60
40    IF (IB(I)-IA(85)) 60,50,60
50    J=J+1
      IF (L-J) 30,60,30
60    CONTINUE
      IF (L-J) 80,70,90
70    IF (K) 80,120,80
80    WRITE (IOTAPE,280)
      GO TO 150
90    IF (IA(83)-IA(85)) 80,100,80
100   K=2*(L/2)-L
      IF (K) 110,120,110
110   IW=IW+1
      IB(IW)=IA(85)
120   J=1
130   IF (IB(J)-IA(83)) 140,160,140
140   J=J+1
      IF (J-(IW+1)) 130,150,150
150   RETURN
160   IF (IB(J+1)-IA(85)) 190,170,190
170   J=J+2
      DO 180 I=J,IWA
180   IB(I-1)=IB(I)
      GO TO 220
```

```
190   IF (IB(J+2)-IA(85)) 230,200,230
200   J=J+3
      DO 210 I=J,IW
210   IB(I-1)=IB(I)
220   IW=IW-1
      IB(IW+1)=IA(47)
      J=J-1
      GO TO 130
230   IF ((IW+1)-MAXIW) 250,250,240
240   WRITE (IOTAPE,270)
      GO TO 150
250   IW=IW+1
      J=J+3
      K=IW
      DO 260 L=J,IW
      IB(K)=IB(K-1)
260   K=K-1
      J=J-1
      IB(J)=IA(83)
      GO TO 160
270   FORMAT (116HOTHE WORK ON THE FOLLOWING LINE WAS HALTED JUST BEFORE
     1 THE MAXIMUM CHARACTER LINE LIMIT WAS EXCEEDED IN  SUNLK      )
280   FORMAT (69HOTHE FOLLOWING LINE DID NOT CONTAIN A BALANCED SET OF S
     1HIFT SYMBOLS. )
      END
C
C
      SUBROUTINE SULOCK(IA,IB,IW,IOTAPE)
      DIMENSION IA(86),IB(999)
      MAXIW=998
      J=1
20    IF (IB(J)-IA(84)) 30,60,30
30    J=J+1
      IF (J-(IW+1)) 20,20,50
40    WRITE (IOTAPE,140)
50    RETURN
60    J=J+2
70    IF (IB(J)-IA(84)) 110,80,110
80    IW=IW-1
      DO 90 K=J,IW
90    IB(K)=IB(K+1)
      IB(IW+1)=IA(47)
      J=J+1
      IF (J-IW) 70,70,100
100   IW=IW+1
      IB(IW)=IA(86)
      GO TO 50
110   IF (IW-MAXIW) 120,100,40
120   IW=IW+1
      J=J+1
      K=IW
      DO 130 L=J,IW
      IB(K)=IB(K-1)
130   K=K-1
      IB(J-1)=IA(86)
```

```
      GO TO 20
C
140   FORMAT (116HOTHE WORK ON THE FOLLOWING LINE WAS HALTED JUST BEFORV
     1 THE MAXIMUM CHARACTER LINE LIMIT WAS EXCEEDED IN  SULOCK     )
      END
C
C
      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
C
C
C
C                                SEARCH                                 SEAR0010
C                                                                       SEAR0020
C THIS CODE WAS WRITTEN IN FOUR SECTIONS DESIGNED TO WORK INDEPENDENTLY SEAR0030
C                     INPUT                                             SEAR0040
C THE FIRST CARD FOLLOWING $DATA CONTAINS THE  SPECIAL SEARCH CHARACTER SEAR0050
C     IA(47)   CONTAINS THE THE CHARACTER PRECEDING THE START OF A WORD SEAR0060
C             'NORMALLY A BLANK                                         SEAR0070
C     IA(50)   CONTAINS THE CHARACTER SIGNALING THE END OF A SEARCH WORDSEAR0080
C              OR PHRASE , NORMALLY A BLANK OR A PERIOD.                SEAR0090
C THE SUGGESTED FORMAT FOR THE FIRST CARD IS GIVEN BELOW WHERE CARD     SEAR0100
C COLUMN ONE IS FOUND IN 11,2 IN 12 , AND SO FORTH                      SEAR0110
C         ABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789+.)$*/,(=' -              SEAR0120
C THE SECOND CARD CONTAINS FOUR SWITCHES IN I2 FORMAT                   SEAR0130
C      ITEST   IS ONE  WHEN THE CHOSEN LINES ARE TO BE WRITTEN AS CARD  SEAR0140
C              IMAGES ON IPTAPE. ITEST = 0 WHEN NO CARDS ARE TO BE MADE.SEAR0150
C      INSIDE  IS 0 IN THE ANCHORED MODE WHEN THE PROGRAM IS LOOKING    SEAR0160
C              AT THE START OF ANY WORD IN THE CARD IMAGE.  INSIDE IS   SEAR0170
C              1 IN THE UNANCHORED MODE WHEN THE SEARCH PHRASE IS TO    SEAR0180
C              BE LOCATED ANYWHERE ON THE CARD IMAGE.                   SEAR0190
C IRTAPE - A SPECIAL INPUT TAPE. IF ONE INPUT TAPE WILL DO,SET IRTAPE   SEAR0200
C          EQUAL TO ITAPE                                               SEAR0210
C IPTAPE - A SPECIAL OUTPUT TAPE. IT CAN BE USED AS AN INPUT TO OTHER   SEAR0220
```

```
C          PROGRAMS, BUT CANNOT BE EQUATED WITH IOTAPE AS IOTAPE, IN    SEAR0230
C          ADDITION TO NORMAL OUTPUT, CONTAINS OTHER PROGRAM MESSAGES   SEAR0240
C          AND ERROR REMARKS.                                           SEAR0250
C                                                                       SEAR0260
C THE  THIRD CARD CONTAINS EITHER THE WORD 'AND' OR THE WORD 'OR'       SEAR0270
C STARTING IN CARD COLUMN ONE.   'AND' BEING EQUAL TO 1.                SEAR0280
C              THE 'AND' DEMANDS THAT ALL SEARCH WORDS OR PHRASES MUST  SEAR0290
C              BE FOUND IN A CARD IMAGE IN ORDER TO BE CHOSEN.          SEAR0300
C              THE 'OR' REQUIRES ONLY THAT ONE OF THE SEARCH WORDS OR   SEAR0310
C              PHRASES BE FOUND IN ORDER TO BE CHOSEN.                  SEAR0320
C NOW FOLLOW THE SEARCH WORDS OR PHRASES . EACH STARTING IN CARD COLUMN SEAR0330
C              ONE AND ENDING WITH THE SYMBOL STORED IN   IA(50)        SEAR0340
C THE LAST CARD IS TO CONTAIN THE WORD   FINIS   STARTING IN CARD COLUMNSEAR0350
C              ONE AND ENDING WITH THE SYMBOL  IA(50)                   SEAR0360
C                                                                       SEAR0370
C TAPES ARE LABELED ACCORDING TO THEIR USE.                             SEAR0380
C  ITAPE - THE NORMAL SYSTEM INPUT TAPE.                                SEAR0390
C IOTAPE - THE NORMAL SYSTEM OUTPUT TAPE.                               SEAR0400
C                                                                       SEAR0404
C     CODE WRITTEN BY MRS CARLA G. MESSINA  NSRDS NBS 1966              SEAR0406
C                                                                       SEAR0410
      DIMENSION IA(81),IB(81),ICOL(82),N(200),IC(8000)                  SEAR0420
      ITAPE=5                                                           SEAR0430
      IOTAPE=6                                                          SEAR0440
      READ (ITAPE,1) (IA(J),J=1,80)                                     SEAR0450
1     FORMAT(80A1)                                                      SEAR0460
      WRITE (IOTAPE,2) (IA(J),J=1,80)                                   SEAR0470
2     FORMAT(1X,80A1)                                                   SEAR0480
      READ (ITAPE,200) ITEST,INSIDE,IRTAPE,IPTAPE                       SEAR0490
200   FORMAT(40I2)                                                      SEAR0500
201   FORMAT (1X,40I2)                                                  SEAR0510
      IF (IRTAPE) 61,61,62                                              SEAR0520
61    IRTAPE=5                                                          SEAR0530
      GO TO 64                                                          SEAR0540
62    IF (IRTAPE- 6) 64,64,63                                           SEAR0550
63    REWIND IRTAPE                                                     SEAR0560
64    IF (IPTAPE) 65,65,66                                              SEAR0570
65    IPTAPE = 3                                                        SEAR0580
      GO TO 68                                                          SEAR0590
66    IF (IPTAPE - 6) 68,68,67                                          SEAR0600
67    REWIND IPTAPE                                                     SEAR0610
68    WRITE (IOTAPE,201) ITEST,INSIDE,IRTAPE,IPTAPE                     SEAR0620
      READ (ITAPE,1) (IB(J),J=1,80)                                     SEAR0630
      WRITE (IOTAPE,2) (IB(J),J=1,80)                                   SEAR0640
      IF (IB(1) - IA(1)) 4,69,4                                         SEAR0650
69    IF (IB(2) - IA(14)) 4,70,4                                        SEAR0660
70    IF (IB(3) - IA( 4))  4,71,4                                       SEAR0670
71    IAND = 1                                                          SEAR0680
      WRITE (IOTAPE,3) IA(1),IA(12),IA(12)                              SEAR0690
3     FORMAT (47HOTHE PROGRAM IS SEARCHING FOR LINES CONTAINING  ,3A1,26SEAR0700
     1H OF THE WORDS GIVEN BELOW.   //)                                 SEAR0710
      GO TO 5                                                           SEAR0720
4     IAND = 0                                                          SEAR0730
      WRITE (IOTAPE,3) IA(1),IA(14),IA(25)                              SEAR0740
5     N1=0                                                              SEAR0750
```

```
      N3=1                                                              SEAR0760
6     READ (ITAPE,1) (IB(J),J=1,80)                                     SEAR0770
7     N2=0                                                              SEAR0780
      J=1                                                               SEAR0790
      IF (IB(1) - IA( 6)) 8,72,8                                        SEAR0800
72    IF (IB(2) - IA(9)) 8,73,8                                         SEAR0810
73    IF (IB(3) - IA(14)) 8,74,8                                        SEAR0820
74    IF (IB(4) - IA( 9)) 8,75,8                                        SEAR0830
75    IF (IB(5) - IA(19)) 8,76,8                                        SEAR0840
76    IF (IB(6) - IA(50)) 8,17,8                                        SEAR0850
8     DO 9 I=1,80                                                       SEAR0860
      IF (IB(I) - IA(50)) 9,10,9                                        SEAR0870
9     N2=I                                                              SEAR0880
10    IF (N2) 6,6,11                                                    SEAR0890
11    N1=N1+1                                                           SEAR0900
      N(N1)=N2                                                          SEAR0910
      N4=N3+N2-1                                                        SEAR0920
      IF (N4-8000) 14,14,12                                             SEAR0930
12    WRITE (IOTAPE,13) N4,N1                                           SEAR0940
13    FORMAT (47HOLIST OF SEARCH WORDS TOO LONG. PLEASE SHORTEN.  ,/68HOSEAR0950
     1MAXIMUM CHARACTER LENGTH IS 8000  MAXIMUM NUMBER OF PHRASES IS 200SEAR0960
     2 /21H  CURRENT VALUES ARE  ,2I6)                                  SEAR0970
999   STOP                                                              SEAR0980
14    IF (N1-200) 15,15,12                                              SEAR0990
15    J=1                                                               SEAR1000
      DO 16 I=N3,N4                                                     SEAR1010
      IC(I) = IB(J)                                                     SEAR1020
16    J=J+1                                                             SEAR1030
      N3=N3+N2                                                          SEAR1040
      WRITE (IOTAPE,2) (IB(J),J=1,N2)                                   SEAR1050
      GO TO 6                                                           SEAR1060
17    ICOL(1) =IA(47)                                                   SEAR1070
      WRITE (IOTAPE,2) IA(47)                                           SEAR1080
      WRITE (IOTAPE,2) IA(47)                                           SEAR1090
      WRITE (IOTAPE,2) IA(47)                                           SEAR1100
      IF (IAND) 40,18,40                                                SEAR1110
18    IF (INSIDE) 19,30,19                                              SEAR1120
19    READ (IRTAPE,1) (ICOL(J),J=2,81)                                  SEAR1130
      CALL CHECK1(IA,ICOL,K,ITEST,IPTAPE,IOTAPE)                        SEAR1140
      IF (K) 999,190,999                                                SEAR1150
190   N3=1                                                              SEAR1160
      DO 24 K1=1,N1                                                     SEAR1170
      N2=N(K1)                                                          SEAR1180
      N4=N3+N2-1                                                        SEAR1190
      I2=82-N2                                                          SEAR1200
      DO 22 I=2,I2                                                      SEAR1210
       J1=1                                                             SEAR1220
      DO 20 J=N3,N4                                                     SEAR1230
      K=I+J1-1                                                          SEAR1240
      IF (ICOL(K) - IC(J)) 22,20,22                                     SEAR1250
20    J1=J1+1                                                           SEAR1260
      GO TO 26                                                          SEAR1270
22    CONTINUE                                                          SEAR1280
24    N3=N3+N2                                                          SEAR1290
      GO TO 19                                                          SEAR1300
```

```
26    WRITE (IOTAPE,2) (ICOL(J),J=2,81)                          SEAR1310
      IF (ITEST) 28,19,28                                        SEAR1320
28    WRITE (IPTAPE,1) (ICOL(J),J=2,81)                          SEAR1330
      GO TO 19                                                   SEAR1340
30    READ  (IRTAPE,1) (ICOL(J),J=2,81)                          SEAR1350
      CALL CHECK1(IA,ICOL,K,ITEST,IPTAPE,IOTAPE)                 SEAR1360
      IF (K) 999,300,999                                         SEAR1370
300   N3=1                                                       SEAR1380
      DO 36 K1=1,N1                                              SEAR1390
      N2=N(K1)                                                   SEAR1400
      N4=N3+N2-1                                                 SEAR1410
      I2=82-N2                                                   SEAR1420
      DO 34 I=2,I2                                               SEAR1430
      IF (ICOL(I) - IA(47)) 77,34,77                             SEAR1440
77    IF (ICOL(I-1) - IA(47)) 34,78,34                           SEAR1450
78     J1=1                                                      SEAR1460
      DO 32 J=N3,N4                                              SEAR1470
      K=I+J1-1                                                   SEAR1480
      IF (ICOL(K) - IC(J)) 34,32,34                              SEAR1490
32    J1=J1+1                                                    SEAR1500
      GO TO 38                                                   SEAR1510
34    CONTINUE                                                   SEAR1520
36    N3=N3+N2                                                   SEAR1530
      GO TO 30                                                   SEAR1540
38    WRITE (IOTAPE,2) (ICOL(J),J=2,81)                          SEAR1550
      IF (ITEST) 39,30,39                                        SEAR1560
39    WRITE (IPTAPE,1) (ICOL(J),J=2,81)                          SEAR1570
      GO TO 30                                                   SEAR1580
40    IF (INSIDE) 42,50,42                                       SEAR1590
42    READ (IRTAPE,1) (ICOL(J),J=2,81)                           SEAR1600
      CALL CHECK1(IA,ICOL,K,ITEST,IPTAPE,IOTAPE)                 SEAR1610
      IF (K) 999,420,999                                         SEAR1620
420   N3=1                                                       SEAR1630
      DO 48 K1=1,N1                                              SEAR1640
      N2=N(K1)                                                   SEAR1650
      N4=N3+N2-1                                                 SEAR1660
      I2=82-N2                                                   SEAR1670
      DO 46 I=2,I2                                               SEAR1680
       J1=1                                                      SEAR1690
      DO 44 J=N3,N4                                              SEAR1700
      K=I+J1-1                                                   SEAR1710
      IF (ICOL(K) - IC(J)) 46,44,46                              SEAR1720
44    J1=J1+1                                                    SEAR1730
      GO TO 48                                                   SEAR1740
46    CONTINUE                                                   SEAR1750
      GO TO 42                                                   SEAR1760
48    N3=N3+N2                                                   SEAR1770
      WRITE (IOTAPE,2) (ICOL(J),J=2,81)                          SEAR1780
      IF (ITEST) 49,42,49                                        SEAR1790
49    WRITE (IPTAPE,1) (ICOL(J),J=2,81)                          SEAR1800
      GO TO 42                                                   SEAR1810
50    READ (IRTAPE,1) (ICOL(J),J=2,81)                           SEAR1820
      CALL CHECK1(IA,ICOL,K,ITEST,IPTAPE,IOTAPE)                 SEAR1830
      IF (K) 999,500,999                                         SEAR1840
500   N3=1                                                       SEAR1850
```

```
      DO 58 K1=1,N1                                                    SEAR1860
      N2=N(K1)                                                         SEAR1870
      N4=N3+N2-1                                                       SEAR1880
      I2=82-N2                                                         SEAR1890
      DO 56 I=2,I2                                                     SEAR1900
      IF (ICOL(I) - IA(47))  79,56,79                                  SEAR1910
79    IF (ICOL(I-1) - IA(47)) 56,80,56                                 SEAR1920
80     J1=1                                                            SEAR1930
      DO 54 J=N3,N4                                                    SEAR1940
      K=I+J1-1                                                         SEAR1950
      IF (ICOL(K) - IC(J)) 56,54,56                                    SEAR1960
54    J1=J1+1                                                          SEAR1970
      GO TO 58                                                         SEAR1980
56    CONTINUE                                                         SEAR1990
      GO TO 50                                                         SEAR2000
58    N3=N3+N2                                                         SEAR2010
      WRITE (IOTAPE,2) (ICOL(J),J=2,81)                                SEAR2020
      IF (ITEST) 59,50,59                                              SEAR2030
59    WRITE (IPTAPE,1) (ICOL(J),J=2,81)                                SEAR2040
      GO TO 50                                                         SEAR2050
       END                                                             SEAR2060
C
      SUBROUTINE CHECK1(IA,II,K,ITEST,IPTAPE,IOTAPE)                   SECK0010
      DIMENSION IA(81),II(82),IJ(82)                                   SECK0020
      DO 40 I = 2, 81                                                  SECK0030
40    IJ(I-1) =II(I)                                                   SECK0040
      DO 10 I=1,26                                                     SECK0050
      IF (IA(I)-IJ(I)) 50,10,50                                        SECK0060
10    CONTINUE                                                         SECK0070
      K=1                                                              SECK0080
      WRITE (IOTAPE,19) (IA(I),I=1,80)                                 SECK0090
      IF (ITEST) 20,60,20                                              SECK0100
20    WRITE (IPTAPE,29) (IA(I),I=1,80)                                 SECK0110
      IF (IPTAPE -6) 60,60,30                                          SECK0120
30    END FILE IPTAPE                                                  SECK0130
      GO TO 60                                                         SECK0140
50    K=0                                                              SECK0150
60    RETURN                                                           SECK0160
19    FORMAT (1X,80A1)                                                 SECK0170
29    FORMAT (80A1)                                                    SECK0180
      END                                                              SECK0190
C
C
C                        BLOCK    SEARCH                               BSEA0001
C                                                                      BSEA0002
C     THIS PROGRAM PERFORMS A SEARCH IN A FILE MADE UP OF VARIABLE     BSEA0004
C LENGTH BLOCKS, SUITABLY DELIMITED, FOR THE OCCURANCE OF ONE OR MORE  BSEA0006
C SPECIFIED STRINGS OF CHARACTERS.  THE BLOCK IS SELECTED FOR OUTPUT   BSEA0008
C IF THE DESIRED STRING OR STRINGS ARE FOUND WITHIN IT.  THE BLOCK     BSEA0010
C DELIMITER, SEARCH STRATEGIES AND DESIRED STRINGS ARE INPUT PARAMETERS.BSEA0012
C                                                                      BSEA0020
C THE FIRST CARD FOLLOWING THE XQT CONTAINS THE SPECIAL SEARCH CHARACTERBSEA0030
C      A(47)   CONTAINS THE THE CHARACTER PRECEDING THE START OF A WORD BSEA0040
C              NORMALLY A BLANK                                        BSEA0050
C      A(50)   CONTAINS THE CHARACTER SIGNALING THE END OF A SEARCH WORDBSEA0060
```

```
C              OR PHRASE , NORMALLY A BLANK OR A PERIOD.                BSEA0070
C THE SUGGESTED FORMAT FOR THE FIRST CARD IS GIVEN BELOW WHERE CARD     BSEA0080
C COLUMN 1 IS IN LOCATION 11, 2 IS IN 12 ETC.                          BSEA0090
C         ABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789+.)$*/,(-' =             BSEA0100
C THE SECOND CARD CONTAINS FIVE SWITCHES IN (I2) FORMAT                BSEA0110
C      ITEST   IS NONZERO WHEN THE CHOSEN ONES ARE TO BE WRITTEN AS CARDBSEA0120
C              IMAGES ON LOGICAL UNIT IPTAPE. ITEST =0 WHEN NO CARDS   BSEA0130
C              ARE TO BE MADE.                                         BSEA0140
C      INSIDE  IS 0  WHEN THE CODE IS TO LOOK ONLY AT THE START OF ANY BSEA0150
C              WORD IN THE CARD IMAGE. INSIDE = 1. WHEN THE SEARCH     BSEA0160
C              STRING IS SOUGHT ANYWHERE ON THE CARD IMAGE.            BSEA0170
C      IFLAG   IS THE LENGTH OF THE BLOCK FLAG ON THE FOLLOWING CARD.  BSEA0180
C              IF IFLAG IS LEFT BLANK,THE ENTIRE CARD WILL BE CONDIDEREDBSEA0190
C              THE FLAG.                                               BSEA0200
C      IRTAPE  IS THE LOGICAL UNIT TO READ THE INPUT RECORDS FROM. IF  BSEA0210
C              ZERO, IT IS SET TO 5(UNIVAC 1108 CARD READER)           BSEA0220
C      IPTAPE  IS THE LOGICAL UNIT TO WRITE THE OUTPUT RECORDS ON. IF  BSEA0230
C              ZERO, IT IS SET TO 3(UNIVAC 1108 CARD PUNCH)            BSEA0240
C      IWIDE1  IS THE WIDTH OF THE LIBRARY IMAGES ON UNIT IRTAPE.  IF  BSEA0244
C              IWIDE1 IS NOT A VALUE BETWEEN 0 AND 132 IT IS SET TO 80 BSEA0246
C THE THIRD  CARD CONTAINS THE BLOCK FLAG STARTING IN CARD COLUMN ONE ANBSEA0250
C              CONTINUING UP TO CARD COLUMN IFLAG OR 80                BSEA0260
C THE FOURTH CARD CONTAINS EITHER THE WORD 'AND' OR THE WORD 'OR'       BSEA0270
C STARTING IN CARD COLUMN ONE.   'AND' BEING EQUAL TO 1.               BSEA0280
C              THE 'AND' DEMANDS THAT ALL SEARCH WORDS OR PHRASES MUST BSEA0290
C              BE FOUND IN A CARD IMAGE IN ORDER TO BE CHOSEN.         BSEA0300
C              THE 'OR' REQUIRES ONLY THAT ONE OF THE SEARCH WORDS OR  BSEA0310
C              PHRASES BE FOUND IN ORDER TO BE CHOSEN.                 BSEA0320
C NOW FOLLOW THE SEARCH WORDS OR PHRASES . EACH STARTING IN CARD COLUMN BSEA0330
C              ONE AND ENDING WITH THE SYMBOL STORED IN A(50)          BSEA0340
C THE LAST CARD IS TO CONTAIN THE WORD  FINIS  STARTING IN CARD COLUMN  BSEA0350
C              ONE.  AND ENDING WITH THE SYMBOL A(50).                 BSEA0360
C THE CODE NOW STARTS READING THE LIBRARY IMAGES FROM UNIT IRTAPE.      BSEA0370
C                                                                      BSEA0375
C     CODE WRITTEN BY MRS CARLA G. MESSINA  NSRDS NBS 1966             BSEA0380
C                                                                      BSEA0390
      DIMENSION IA(81),IB(81),IC(8080),ID(140),IFLAGS(80),N(200)       BSEA0400
      DIMENSION ICOL(8300)                                             BSEA0410
      COMMON ITEST,ITAPE,IOTAPE,IRTAPE,IPTAPE,IA,IB,IC,ID,IFLAGS,N,ICOL BSEA0420
      ITAPE=5                                                          BSEA0430
      IOTAPE=6                                                         BSEA0440
      IEND=1                                                           BSEA0450
      READ (ITAPE,9) (IA(J),J=1,80)                                    BSEA0460
      WRITE (IOTAPE,19) (IA(I),I=1,80)                                 BSEA0470
      READ (ITAPE,29) ITEST,INSIDE,IFLAG,IRTAPE,IPTAPE,J,IWIDE1        BSEA0480
      IF (IRTAPE) 10,10,20                                             BSEA0490
10    IRTAPE=5                                                         BSEA0500
      GO TO 30                                                         BSEA0510
20    IF (IRTAPE - 6) 30,30,25                                         BSEA0520
25    REWIND IRTAPE                                                    BSEA0530
30    IF (IPTAPE) 32,32,34                                             BSEA0540
32    IPTAPE=3                                                         BSEA0550
      IWIDE3=81                                                        BSEA0560
      GO TO 36                                                         BSEA0570
34    IF (IPTAPE - 6 ) 32,32,35                                        BSEA0580
```

```
35    REWIND IPTAPE                                                    BSEA0590
36    IF (IFLAG)50,50,40                                               BSEA0600
40    IF (IFLAG-80) 60,60,50                                           BSEA0610
50    IFLAG =80                                                        BSEA0620
60    WRITE (IOTAPE,39) ITEST,INSIDE,IFLAG,IRTAPE,IPTAPE               BSEA0630
      IF (J-1)  63,61,62                                               BSEA0640
61    IWIDE1= IWIDE1 + 100                                             BSEA0650
      IF (IWIDE1 - 132) 63,63,62                                       BSEA0660
62    J=0                                                              BSEA0670
      IWIDE1 = 80                                                      BSEA0680
      GO TO 64                                                         BSEA0690
63    IF (IWIDE1) 62,62,64                                             BSEA0700
64    READ (ITAPE,9) (IFLAGS(J),J=1,IFLAG)                             BSEA0710
      WRITE (IOTAPE,19) (IFLAGS(J),J=1,IFLAG)                          BSEA0720
      IF (IPTAPE-6) 66,66,65                                           BSEA0730
65    IWIDE3=IWIDE1+1                                                  BSEA0740
66    READ (ITAPE,9) (IB(J),J=1,80)                                    BSEA0750
      IF (IB(1) - IA(1)) 90,70,90                                      BSEA0760
70    IF (IB(2) - IA(14)) 90,80,90                                     BSEA0770
80    IF (IB(3) - IA(4))  90,85,90                                     BSEA0780
85    IAND =1                                                          BSEA0790
      WRITE (IOTAPE,49) IWIDE1,IA(1),IA(12),IA(12)                     BSEA0800
      GO TO 100                                                        BSEA0810
90    IAND = 0                                                         BSEA0820
      WRITE (IOTAPE,49) IWIDE1,IA(1),IA(14),IA(25)                     BSEA0830
100   N1=0                                                             BSEA0840
      N3=1                                                             BSEA0850
      L4=0                                                             BSEA0860
110   READ (ITAPE,9) (IB(J),J=1,80)                                    BSEA0870
      WRITE (IOTAPE,19) (IB(J),J=1,80)                                 BSEA0880
      N2=0                                                             BSEA0890
      IF (IB(1) - IA(6)) 170,120,170                                   BSEA0900
120   IF (IB(2) - IA(9)) 170,130,170                                   BSEA0910
130   IF (IB(3) - IA(14))170,140,170                                   BSEA0920
140   IF (IB(4) - IA( 9))170,150,170                                   BSEA0930
150   IF (IB(5) - IA(19))170,160,170                                   BSEA0940
160   IF (IB(6) - IA(50))170,260,170                                   BSEA0950
170   DO 180 I=1,80                                                    BSEA0960
      IF (IB(I) - IA(50)) 180,190,180                                  BSEA0970
180   N2=I                                                             BSEA0980
190   IF (N2) 110,110,200                                              BSEA0990
200   N1=N1+1                                                          BSEA1000
      N(N1)=N2                                                         BSEA1010
      N4=N3+N2-1                                                       BSEA1020
      IF (N4-8000) 220,220,210                                         BSEA1030
210   WRITE (IOTAPE,59) N4,N1                                          BSEA1040
9999  STOP                                                             BSEA1050
220   IF (N1-200) 230,230,210                                          BSEA1060
230   J=1                                                              BSEA1070
      DO 240 I=N3,N4                                                   BSEA1080
      IC(I)=IB(J)                                                      BSEA1090
240   J=J+1                                                            BSEA1100
      N3=N3+N2                                                         BSEA1110
      GO TO 110                                                        BSEA1120
260   WRITE (IOTAPE,19) IA(47)                                         BSEA1130
```

```
      WRITE (IOTAPE,19) IA(47)                                          BSEA1140
      WRITE (IOTAPE,19) IA(47)                                          BSEA1150
      IWIDE2= IWIDE1+1                                                  BSEA1160
      DO 270 I=1,8250,IWIDE2                                            BSEA1170
270   ICOL(I)=IA(47)                                                    BSEA1180
280   L1=-IWIDE1+1                                                      BSEA1190
      L2=0                                                              BSEA1200
      L3=0                                                              BSEA1210
290   IF (L2-8181) 310,300,300                                          BSEA1220
300   WRITE (IOTAPE,69) (ID(J),J=1,IWIDE1)                              BSEA1230
      L4=0                                                              BSEA1240
      GO TO 380                                                         BSEA1250
310   READ (IRTAPE,9)  (ID(J),J=1,IWIDE1)                               BSEA1260
      CALL CHECK4(1,IEND)                                               BSEA1270
      IF (IEND) 305,325,305                                             BSEA1280
305   DO 320 I=1,IFLAG                                                  BSEA1290
      IF (IFLAGS(I) - ID(I)) 350,320,350                                BSEA1300
320   CONTINUE                                                          BSEA1310
C  THIS CARD SIGNALS THE END OF A BLOCK OF CARDS, IF L3 IS ZERO THE LASTBSEA1320
C    CARD WAS ALSO THE END OF A BLOCK.                                  BSEA1330
325   IF (L3) 330,330,370                                               BSEA1340
330   L4=0                                                              BSEA1350
      L3=1                                                              BSEA1360
      L2=IWIDE2                                                         BSEA1370
      L1=2                                                              BSEA1380
      J=1                                                               BSEA1390
      DO 340 I=L1,L2                                                    BSEA1400
      ICOL(I) =ID(J)                                                    BSEA1410
340   J=J+1                                                             BSEA1420
      GO TO 380                                                         BSEA1430
350   L3=L3+1                                                           BSEA1440
      L2=L2+IWIDE2                                                      BSEA1450
      L1=L1+IWIDE2                                                      BSEA1460
      J=1                                                               BSEA1470
      DO 360 I=L1,L2                                                    BSEA1480
      ICOL(I) =ID(J)                                                    BSEA1490
360   J=J+1                                                             BSEA1500
      GO TO 290                                                         BSEA1510
370   L4=1                                                              BSEA1520
380   IF (IAND),490,390,490                                             BSEA1530
390   IF (INSIDE) 400,440,400                                           BSEA1540
400   N3=1                                                              BSEA1550
      DO 430 K1=1,N1                                                    BSEA1560
      N2=N(K1)                                                          BSEA1570
      N4=N3+N2-1                                                        BSEA1580
      I2=L2-N2+1                                                        BSEA1590
      DO 420 I=2,I2                                                     BSEA1600
      J1=1                                                              BSEA1610
      DO 410 J=N3,N4                                                    BSEA1620
      K=I+J1-1                                                          BSEA1630
      IF (ICOL(K) - IC(J)) 420,410,420                                  BSEA1640
410   J1=J1+1                                                           BSEA1650
      GO TO 590                                                         BSEA1660
420   CONTINUE                                                          BSEA1670
430   N3=N3+N2                                                          BSEA1680
```

```
      GO TO 630                                                    BSEA1690
440   N3=1                                                         BSEA1700
      DO 480 K1=1,N1                                               BSEA1710
      N2=N(K1)                                                     BSEA1720
      N4=N3+N2-1                                                   BSEA1730
      I2=L2-N2+1                                                   BSEA1740
      DO 470  I=2,I2                                               BSEA1750
      IF (ICOL(I) - IA(47)) 450,470,450                            BSEA1760
450   IF (ICOL(I-1) - IA(47)) 470,455,470                          BSEA1770
455   J1=1                                                         BSEA1780
      DO 460 J=N3,N4                                               BSEA1790
      K=I+J1-1                                                     BSEA1800
      IF (ICOL(K) - IC(J))  470,460,470                            BSEA1810
460   J1=J1+1                                                      BSEA1820
      GO TO 590                                                    BSEA1830
470   CONTINUE                                                     BSEA1840
480   N3=N3+N2                                                     BSEA1850
      GO TO 630                                                    BSEA1860
490   IF (INSIDE) 500,540,500                                      BSEA1870
500   N3=1                                                         BSEA1880
      DO 530 K1=1,N1                                               BSEA1890
      N2=N(K1)                                                     BSEA1900
      N4=N3+N2-1                                                   BSEA1910
      I2=L2-N2+1                                                   BSEA1920
      DO 520 I=2,I2                                                BSEA1930
      J1=1                                                         BSEA1940
      DO 510 J=N3,N4                                               BSEA1950
      K=I+J1-1                                                     BSEA1960
      IF (ICOL(K) - IC(J)) 520,510,520                             BSEA1970
510   J1=J1+1                                                      BSEA1980
      GO TO 530                                                    BSEA1990
520   CONTINUE                                                     BSEA2000
      GO TO 630                                                    BSEA2010
530   N3=N3+N2                                                     BSEA2020
      GO TO 590                                                    BSEA2030
540   N3=1                                                         BSEA2040
      DO 580 K1=1,N1                                               BSEA2050
      N2=N(K1)                                                     BSEA2060
      N4=N3+N2-1                                                   BSEA2070
      I2=L2-N2+1                                                   BSEA2080
      DO 560 I=2,I2                                                BSEA2090
      IF (ICOL(I) - IA(47)) 545,570,545                            BSEA2100
545   IF (ICOL(I-1) - IA(47)) 570,550,570                          BSEA2110
550   J1=1                                                         BSEA2120
      DO 560 J=N3,N4                                               BSEA2130
      K=I+J1-1                                                     BSEA2140
      IF (ICOL(K) - IC(J)) 570,560,570                             BSEA2150
560   J1=J1+1                                                      BSEA2160
      GO TO 580                                                    BSEA2170
570   CONTINUE                                                     BSEA2180
      GO TO 630                                                    BSEA2190
580   N3=N3+N2                                                     BSEA2200
590   DO 600 I=1,L2,IWIDE2                                         BSEA2210
      L5=I+IWIDE1                                                  BSEA2220
600   WRITE (IOTAPE,19) (ICOL(J),J=I,L5)                           BSEA2230
```

```
      IF (ITEST) 610,630,610                                          BSEA2240
610   DO 620 I=1,L2,IWIDE3                                            BSEA2250
      L5=I+IWIDE3-1                                                   BSEA2260
620   WRITE (IPTAPE,9) (ICOL(J),J=L4,L5)                              BSEA2270
630   CALL CHECK4(0,IEND)                                             BSEA2280
      IF (IEND)  640,9999,640                                         BSEA2290
640      IF (L4) 280,280,650                                          BSEA2300
650   L4=0                                                            BSEA2310
      L3=1                                                            BSEA2320
      L2=IWIDE1+1                                                     BSEA2330
      L1=2                                                            BSEA2340
      J=1                                                             BSEA2350
      DO 660 I=L1,L2                                                  BSEA2360
      ICOL(I) =ID(J)                                                  BSEA2370
660   J=J+1                                                           BSEA2380
      GO TO 310                                                       BSEA2390
9     FORMAT (132A1)                                                  BSEA2400
19    FORMAT (1X,131A1)                                               BSEA2410
29    FORMAT (40I2)                                                   BSEA2420
39    FORMAT (1X,40I2)                                                BSEA2430
49    FORMAT (46HOTHE PROGRAM IS SEARCHING IN LINES OF LENGTH  ,1I4,  BSEA2440
     112H CONTAINING  ,3A1,26H OF THE WORDS GIVEN BELOW.   //)        BSEA2450
59    FORMAT (47HOLIST OF SEARCH WORDS TOO LONG. PLEASE SHORTEN.  ,/68HOBSEA2460
     1MAXIMUM CHARACTER LENGTH IS 8000. MAXIMUM NUMBER OF PHRASES IS 200BSEA2470
     2 /21H  CURRENT VALUES ARE   ,2I6)                               BSEA2480
69    FORMAT (74HOBLOCK IS TOO LONG. BLOCK IS CONSIDERED TERMINATED WITHBSEA2490
     1 THE FOLLOWING CARD    ,//1X,131A1//)                          BSEA2500
      END                                                             BSEA2510
C
      SUBROUTINE CHECK4(IT,K)                                         BSCK0010
      DIMENSION IA(81),IB(81),IC(8080),ID(140),IFLAGS(80),N(200)      BSCK0020
      DIMENSION ICOL(8300)                                            BSCK0030
      COMMON ITEST,ITAPE,IOTAPE,IRTAPE,IPTAPE,IA,IB,IC,ID,IFLAGS,N,ICOL BSCK0040
      IF (IT)   10,60,10                                              BSCK0050
10    DO 20 I=1,26                                                    BSCK0060
      IF (IA(I)-ID(I)) 40,20,40                                       BSCK0070
20    CONTINUE                                                        BSCK0080
      K=0                                                             BSCK0090
      GO TO 50                                                        BSCK0100
30    ENDFILE IPTAPE                                                  BSCK0110
      GO TO 50                                                        BSCK0120
40    K=1                                                             BSCK0130
50    RETURN                                                          BSCK0140
60    IF (K) 50,70,50                                                 BSCK0150
70    WRITE (IOTAPE,9) (IA(J),J=1,80)                                 BSCK0160
      IF (ITEST) 80,50,80                                             BSCK0170
80    WRITE (IPTAPE,19)(IA(J),J=1,80)                                 BSCK0180
      IF (IPTAPE-6)  50,50,30                                         BSCK0190
9     FORMAT (1X,131A1)                                               BSCK0200
19    FORMAT (132A1)                                                  BSCK0210
       END                                                            BSCK0220
C
C
C                          JUSTIFY                                    JUAT  1
C                                                                     JUAT  2
```

```
C     JUSTIFY PRODUCES RIGHT-HAND JUSTIFIED TEXT WHILE RECOGNIZING      JUAT  3
C     THE START OF A NEW PARAGRAPH BY LEADING BLANKS AND RECOGNIZING    JUAT  4
C     SYMBOLS IN CONTROL FIELD 72 OF THE CARD FOR INDENTING (N), FOR    JUAT  5
C     CENTERING LINES (C), FOR IGNORING LINES (I), FOR LEAVING LINES    JUAT  6
C     UNALTERED (D), AND FOR CREATING RUNAROUNDS (R).                   JUAT  7
C     JUSTIFY REFORMATS PARAGRAPHS BY FIRST REMOVING ALL EXTRANEOUS     JUAT  8
C     BLANKS AND THEN REMAKES THE PARAGRAPH INTO LINES OF THE SPECIFIED JUAT  9
C     WIDTH WITHOUT HYPENATING WORDS.   TO MAKE RIGHT HAND JUSTIFIED LINJUAT 10
C     EXTRA SPACES ARE REINSERTED AS NECESSARY.                         JUAT 11
C                                                                       JUAT 12
C     THE FIRST INPUT DATA CARD, STORED IN THE DIMENSIONED ARRAY A(80), JUAT 13
C     CONTAINS CONTROL LETTERS AND SYMBOLS.  THE CARD FIELDS ARE TREATEDJUAT 14
C     FOLLOWS.                                                          JUAT 15
C                                                                       JUAT 16
C     A(1) TO A(26)          CONTAINS THE ALPHABET                      JUAT 17
C                   THEREFORE A(3) IS ASSUMED TO BE C, A(4) TO BE D, ETC JUAT 18
C     A(27) TO A(36)         CONTAINS THE INTEGERS STARTING WITH ZERO   JUAT 19
C     A(38)                  CONTAINS A PERIOD OR END OF SENTENCE SYMBOL.JUAT 20
C     A(39)                  TERMINATES THE READING OF A CARD.  ANYTHING JUAT 21
C                   FOUND IN FURTHER FIELDS OF THE CARD WILL BE IGNORED  JUAT 22
C                   EXCEPT CONTROL LETTERS IN CARD FIELD 72.            JUAT 23
C     A(45)                  IS USED TO INDICATE THAT THE OUTPUT LINE IS JUAT 24
C                   CONTINUED ONTO THE NEXT CARD.                       JUAT 25
C     A(46)                  SHIFT SYMBOL USED IN LOCK AND UNLOCK       JUAT 26
C     A(47)                  NORMALLY A BLANK, IS THE CHARACTER WHICH   JUAT 27
C                   SEPARATES WORDS.  IT IS, THEREFORE, THE ONLY VALID   JUAT 28
C                   CHARACTER AT WHICH TO BREAK A LINE.  NOTE THAT THIS  JUAT 29
C                   PROGRAM REMOVES ALL EXTRA SPACES BETWEEN WORDS.  IF  JUAT 30
C                   RESERVED SPACES ARE DESIRED A SPECIAL CHARACTER OTHERJUAT 31
C                   THAN A(47) MUST BE USED.                            JUAT 32
C     A(48)                  IS A TAB INDICATOR.  THIS SYMBOL IS TREATED JUAT 33
C                   AN A(47) WHEN ENCOUNTERED IN THE INPUT TEXT EXCEPT   JUAT 34
C                   THAT IT WILL NOT START A NEW PARAGRAPH IF ENCOUNTEREDJUAT 35
C                   IN CARD COLUMN ONE.                                 JUAT 36
C     A(60)                  CONTAINS AN INTEGER COUNT OF THE NUMBER OF  JUAT 37
C                   FIGURING THE LENGTH OF THE OUTPUT LINE,  ALL OTHER   JUAT 39
C                   CHARACTERS AND SYMBOLS HAVE A WIDTH OF ONE. THESE NO JUAT 40
C                   COUNT SYMBOLS ARE PUNCHED STARTING IN A(61).        JUAT 41
C                                                                       JUAT 42
C     THE SECOND INPUT CARD CONTAINS CONTROL SWITCHES IN (I3) FORMAT.   JUAT 43
C                                                                       JUAT 44
C     FIELD                          USE                              JUAT 45
C                                                                       JUAT 46
C       1          WIDTH OF OUTPUT RECORDS                            JUAT 47
C       2          NONZERO INSERTS SPACES TO RIGHT JUSTIFY LINES.     JUAT 48
C                  A ZERO FORMATS LINES WITHOUT EXTRA SPACES INSERTED. JUAT 49
C       3          NONZERO WRITES A PUNCH TAPE, ZERO SUPRESSES PUNCH. JUAT 50
C       4          NONZERO CALLS SUBROUTINES LOCK AND UNLOCK.         JUAT 51
C       5          UNIT NO. OF READ TAPE(SET TO 5 IS ZERO)            JUAT 52
C       6          UNIT NO. OF PUNCH TAPE (SET TO 3 IF ZERO)          JUAT 53
C       7          ZERO INSERTS ONE EXTRA SPACE AFTER A PERIOD A(38)  JUAT 54
C                  WHEN POSSIBLE.                                     JUAT 55
C                                                                       JUAT 56
C     A ''PARAGRAPH'' IS CONSIDERED ENDED BY ONE OR MORE BLANKS AT THE  JUAT 57
C     BEGINNING OF A NEW LINE, A CARD CONTAINING ALL BLANKS, A CARD     JUAT 58
```

```
C     CONTAINING ONLY BLANKS AND  A(39)'S, OR A CARD CONTAINING THE SPECJUAT 59
C     LETTERS, A(3), A(4), A(14), OR A(18), IN CARD COLUMN 72.          JUAT 60
C                                                                       JUAT 61
C     CODE WRITTEN BY MRS. CARLA G. MESSINA OF NSRDS-NBS   1965         JUAT 62
C                                                                       JUAT 63
      COMMON ITAPE,IRTAPE,IOTAPE,IPTAPE                                 JUAT 64
      COMMON SWITCH,N,N1,N2,NOPAR,IW2,ITEST,IJUST,IW,IA,ICOL,ICP,ICT,IC JUAT 65
C     CODE BY MRS CARLA G. MESSINA      NSRDS  - NBS     1967           JUAT 66
       DIMENSION ICOL(80),ICP(360),ICT(360),IC(8002),IA(80),SWITCH(100) JUAT 67
      DIMENSION ISWIT(100)                                              JUAT 68
      ITAPE=5                                                           JUAT 69
      IOTAPE=6                                                          JUAT 70
      N=0                                                               JUAT 71
      N1=0                                                              JUAT 72
      NOPAR=0                                                           JUAT 73
      READ (ITAPE,670) (IA(J),J=1,80)                                   JUAT 74
      WRITE (IOTAPE,680) (IA(J),J=1,80)                                 JUAT 75
      READ (ITAPE,690) IW,IJUST,ITEST,(ISWIT(J),J=1,21)                 JUAT 76
      IRTAPE=ISWIT(2)                                                   JUAT 77
      IPTAPE=ISWIT(3)                                                   JUAT 78
      IF (IRTAPE) 20,20,30                                              JUAT 79
20    IRTAPE=5                                                          JUAT 80
      GO TO 50                                                          JUAT 81
30    IF (IRTAPE-6) 50,50,40                                            JUAT 82
40    REWIND IRTAPE                                                     JUAT 83
50    IF (IPTAPE) 60,60,70                                              JUAT 84
60    IPTAPE=3                                                          JUAT 85
      GO TO 90                                                          JUAT 86
70    IF (IPTAPE-6) 90,90,80                                            JUAT 87
80    REWIND IPTAPE                                                     JUAT 88
90    DO 100 J=1,100                                                    JUAT 89
100   SWITCH(J)=0.0                                                     JUAT 90
      SWITCH(1)=1.                                                      JUAT 91
      K=1                                                               JUAT 92
      DO 120 J=2,21                                                     JUAT 93
      IF (ISWIT(J)) 120,120,110                                         JUAT 94
110   K=J                                                               JUAT 95
      SWITCH(J)=ISWIT(J)                                                JUAT 96
120   CONTINUE                                                          JUAT 97
      WRITE (IOTAPE,700) IW,IJUST,ITEST,(ISWIT(J),J=1,K)                JUAT 98
      IF (ISWIT(1)) 130,140,130                                         JUAT 99
130   SWITCH(3)=ISWIT(1)                                                JUAT100
140   WRITE (IOTAPE,730) IW                                             JUAT101
      DO 150 I=1,8002                                                   JUAT102
150   IC(I)=IA(47)                                                      JUAT103
      IF (IW-20) 480,470,470                                            JUAT104
160   READ (IRTAPE,670) (ICOL(J),J=1,80)                                JUAT105
      CALL CHECK3 (K)                                                   JUAT106
      IF (K) 170,180,170                                                JUAT107
170   STOP                                                              JUAT108
180   IF (ICOL(72)-IA(4)) 500,590,500                                   JUAT109
190   IF (NOPAR) 200,210,200                                            JUAT110
200   NOPAR=0                                                           JUAT111
      CALL OUTPUT                                                       JUAT112
210   GO TO (540,550,560,570), IS                                       JUAT113
```

```
220   N2=0                                                    JUAT114
      IF (ICOL(1)-IA(47)) 230,260,230                         JUAT115
230   IF (ICOL(1)-IA(39)) 250,590,250                         JUAT116
240   N1=0                                                    JUAT117
      N=0                                                     JUAT118
      NOPAR=1                                                 JUAT119
      GO TO 330                                               JUAT120
250   IF (NOPAR) 330,240,330                                  JUAT121
260   IF (NOPAR) 270,280,270                                  JUAT122
270   CALL OUTPUT                                             JUAT123
      NOPAR=0                                                 JUAT124
280   DO 290 J=1,80                                           JUAT125
      N1=J                                                    JUAT126
      IF (ICOL(J)-IA(47)) 300,290,300                         JUAT127
290   CONTINUE                                                JUAT128
      GO TO 590                                               JUAT129
300   IF (ICOL(N1)-IA(39)) 310,590,310                        JUAT130
310   N2=N1                                                   JUAT131
      NOPAR=1                                                 JUAT132
      N=N1                                                    JUAT133
      N1=N1-1                                                 JUAT134
      DO 320 J=1,N                                            JUAT135
320   IC(J)=ICOL(J)                                           JUAT136
330   N=N+1                                                   JUAT137
      IF (N-7999) 350,340,350                                 JUAT138
340   WRITE (IOTAPE,710)                                      JUAT139
      GO TO 260                                               JUAT140
350   ISP=0                                                   JUAT141
      N2=N2+1                                                 JUAT142
      IF (N2-80) 360,360,400                                  JUAT143
360   IF (ICOL(N2)-IA(47)) 370,430,370                        JUAT144
370   IF (ICOL(N2)-IA(48)) 380,620,380                        JUAT145
380   IF (ICOL(N2)-IA(39)) 390,400,390                        JUAT146
390   IF (ICOL(N2)-IA(45)) 420,400,420                        JUAT147
400   IF (ISP) 490,410,490                                    JUAT148
410   IC(N)=IA(47)                                            JUAT149
      GO TO 160                                               JUAT150
420   ISP=0                                                   JUAT151
      IC(N)=ICOL(N2)                                          JUAT152
      GO TO 450                                               JUAT153
430   IF (ISP) 460,440,460                                    JUAT154
440   IC(N)=IA(47)                                            JUAT155
      ISP=1                                                   JUAT156
450   N=N+1                                                   JUAT157
460   N2=N2+1                                                 JUAT158
      IF (N2-80) 360,360,400                                  JUAT159
470   IF (IW-120) 640,640,480                                 JUAT160
480   WRITE (IOTAPE,720) IW                                   JUAT161
      STOP                                                    JUAT162
490   N=N-1                                                   JUAT163
      GO TO 160                                               JUAT164
500   IF (ICOL(72)-IA(9)) 510,160,510                         JUAT165
510   IF (ICOL(72)-IA(18)) 520,600,520                        JUAT166
520   IF (ICOL(72)-IA(3)) 530,610,530                         JUAT167
530   IF (ICOL(72)-IA(14)) 220,580,220                        JUAT168
```

```
540   CALL IOCENT                                                    JUAT169
      GO TO 160                                                      JUAT170
550   CALL RABOUT                                                    JUAT171
      GO TO 160                                                      JUAT172
560   CALL IOSAME                                                    JUAT173
      GO TO 160                                                      JUAT174
570   CALL INDENT                                                    JUAT175
      WRITE (IOTAPE,730) IW                                          JUAT176
      GO TO 160                                                      JUAT177
580   IS=4                                                           JUAT178
      GO TO 190                                                      JUAT179
590   IS=3                                                           JUAT180
      GO TO 190                                                      JUAT181
600   IS=2                                                           JUAT182
      GO TO 190                                                      JUAT183
610   IS=1                                                           JUAT184
      GO TO 190                                                      JUAT185
620   IF (N2-1) 480,630,430                                          JUAT186
630   ISP=1                                                          JUAT187
      GO TO 430                                                      JUAT188
640   DO 650 I=28,36                                                 JUAT189
      IF (IA(60)-IA(I)) 650,660,650                                  JUAT190
650   CONTINUE                                                       JUAT191
      SWITCH(2)=0.0                                                  JUAT192
      IA(61)=IA(46)                                                  JUAT193
      GO TO 160                                                      JUAT194
660   SWITCH(2)=I-27                                                 JUAT195
      GO TO 160                                                      JUAT196
C                                                                    JUAT197
670   FORMAT (80A1)                                                  JUAT198
680   FORMAT (1X,80A1)                                               JUAT199
690   FORMAT (24I3)                                                  JUAT200
700   FORMAT (1X,24I3)                                               JUAT201
710   FORMAT (62H0ONLY 8000 CHARACTERS OF A PARAGRAPH ARE JUSTIFIED AT AJUAT202
     1  TIME.)                                                        JUAT203
720   FORMAT (50H1THE WIDTH CANNOT BE LESS THAN 20 OR MORE THAN 120,4H IJUAT204
     1W=,I3)                                                        JUAT205
730   FORMAT (29H THE WIDTH OF OUTPUT TEXT IS ,I3///)                JUAT206
      END                                                            JUAT207
C
C
      SUBROUTINE OUTPUT                                              JOAT  1
      COMMON ITAPE,IRTAPE,IOTAPE,IPTAPE                              JOAT  2
      COMMON SWITCH,N,N1,N2,NOPAR,IW2,ITEST,IJUST,IW,IA,ICOL,ICP,ICT,IC JOAT  3
       DIMENSION ICOL(80),ICP(360),ICT(360),IC(8002),IA(80),SWITCH(10C) JOAT  4
      IF (SWITCH(3)) 20,30,20                                        JOAT  5
20    CALL UNLOCK                                                    JOAT  6
      CALL LOCK                                                      JOAT  7
30    N=N-1                                                          JOAT  8
      IW1=0                                                          JOAT  9
      DO 50 I=1,N                                                    JOAT 10
      CALL COWNIT (IC(I),IT)                                         JOAT 11
      IF (IT) 40,50,40                                               JOAT 12
40    IW1=IW1+1                                                      JOAT 13
50    CONTINUE                                                       JOAT 14
```

```
      IF (IW-IW1) 90,60,60                                    JOAT 15
60    IW2=N                                                   JOAT 16
      DO 70 I=1,IW2                                           JOAT 17
70    ICP(I)=IC(I)                                            JOAT 18
      CALL PRINTP                                             JOAT 19
80    RETURN                                                  JOAT 20
90    J=0                                                     JOAT 21
      I=0                                                     JOAT 22
      I1=1                                                    JOAT 23
      K=0                                                     JOAT 24
      K1=0                                                    JOAT 25
      K2=0                                                    JOAT 26
100   CALL COWNIT (IC(I1),IT)                                 JOAT 27
      IF (IT) 110,120,110                                     JOAT 28
110   I=I+1                                                   JOAT 29
      GO TO 130                                               JOAT 30
120   J=J+1                                                   JOAT 31
130   I1=I1+1                                                 JOAT 32
      IF (I1-(N+1)) 140,240,240                               JOAT 33
140   IF (I-IW) 100,150,150                                   JOAT 34
150   K=K1+1                                                  JOAT 35
      CALL COWNIT (IC(I1),IT)                                 JOAT 36
      IF (IT) 170,160,170                                     JOAT 37
160   J=J+1                                                   JOAT 38
      I=I-1                                                   JOAT 39
170   IF (IC(I1)-IA(47)) 180,250,180                          JOAT 40
180   I1=I1-1                                                 JOAT 41
      CALL COWNIT (IC(I1+1),IT)                               JOAT 42
      IF (IT) 190,200,190                                     JOAT 43
190   I=I-1                                                   JOAT 44
      GO TO 210                                               JOAT 45
200   J=J-1                                                   JOAT 46
210   IF (I1) 220,220,170                                     JOAT 47
220   WRITE (IOTAPE,880)                                      JOAT 48
230   STOP                                                    JOAT 49
240   K=K1+1                                                  JOAT 50
250   M1=1                                                    JOAT 51
      K1=I1                                                   JOAT 52
      DO 260 I2=K,K1                                          JOAT 53
      ICP(M1)=IC(I2)                                          JOAT 54
260   M1=M1+1                                                 JOAT 55
      M=K1-1                                                  JOAT 56
      IW2=M-K+1                                               JOAT 57
      IF (I1-N) 270,620,620                                   JOAT 58
270   IF (IJUST) 290,620,290                                  JOAT 59
280   CALL PRINTP                                             JOAT 60
      J=0                                                     JOAT 61
      I=0                                                     JOAT 62
      I1=I1+1                                                 JOAT 63
      IF (I1-(N+1)) 100,80,80                                 JOAT 64
290   IF (IW-I) 300,280,310                                   JOAT 65
300   WRITE (IOTAPE,890)                                      JOAT 66
      GO TO 230                                               JOAT 67
310   I2=IW-I                                                 JOAT 68
      K2=K2+1                                                 JOAT 69
```

```
      K4=0                                                          JOAT 70
      IT=2*(K2/2)-K2                                                JOAT 71
      IF (IT) 320,350,320                                           JOAT 72
320   M1=IW2                                                        JOAT 73
      DO 330 I3=1,IW2                                               JOAT 74
      ICT(I3)=ICP(M1)                                               JOAT 75
330   M1=M1-1                                                       JOAT 76
      DO 340 I3=1,IW2                                               JOAT 77
340   ICP(I3)=ICT(I3)                                               JOAT 78
350   IF (K2-1) 300,370,360                                         JOAT 79
360   K3=1                                                          JOAT 80
      GO TO 380                                                     JOAT 81
370   K3=N1+1                                                       JOAT 82
380   I3=1                                                          JOAT 83
390   IF (ICP(I3)-IA(47)) 410,400,410                               JOAT 84
400   K4=K4+1                                                       JOAT 85
410   IF (I3-IW2+K3-1) 420,430,430                                  JOAT 86
420   I3=I3+1                                                       JOAT 87
      GO TO 390                                                     JOAT 88
430   M=I2                                                          JOAT 89
      IT=K4*(I2/K4)-I2                                              JOAT 90
      IF (IT) 450,440,450                                           JOAT 91
440   K5=I2/K4                                                      JOAT 92
      GO TO 460                                                     JOAT 93
450   K5=I2/K4+1                                                    JOAT 94
      IF (K5-1) 710,710,460                                         JOAT 95
460   I4=0                                                          JOAT 96
      I3=1                                                          JOAT 97
470   M1=I3+I4                                                      JOAT 98
      ICT(M1)=ICP(I3)                                               JOAT 99
      IF (ICP(I3)-IA(47)) 510,480,510                               JOAT100
480   DO 490 I5=1,K5                                                JOAT101
      I4=I4+1                                                       JOAT102
      M1=I3+I4                                                      JOAT103
490   ICT(M1)=IA(47)                                                JOAT104
      M=M-1                                                         JOAT105
      IT=K4*(M/K4)-M                                                JOAT106
      IF (IT) 510,500,510                                           JOAT107
500   K5=I2/K4                                                      JOAT108
510   IF (I4-I2) 520,540,540                                        JOAT109
520   IF (I3-IW2) 530,560,560                                       JOAT110
530   I3=I3+1                                                       JOAT111
      GO TO 470                                                     JOAT112
540   I6=I3+1                                                       JOAT113
      DO 550 I13=I6,IW2                                             JOAT114
      M1=I13+I4                                                     JOAT115
550   ICT(M1)=ICP(I13)                                              JOAT116
560   IW2=IW2+I4                                                    JOAT117
570   IT=2*(K2/2)-K2                                                JOAT118
      IF (IT) 600,580,600                                           JOAT119
580   DO 590 I3=1,IW2                                               JOAT120
590   ICP(I3)=ICT(I3)                                               JOAT121
      GO TO 280                                                     JOAT122
600   DO 610 I3=1,IW2                                               JOAT123
      M1=IW2-I3+1                                                   JOAT124
```

```
610   ICP(I3)=ICT(M1)                                                    JOAT125
      GO TO 280                                                          JOAT126
C     IF SWITCH(4) IS ZERO AN EXTRA BLANK IS INSERTED AFTER A PERIOD(38)JOAT127
620   IF (SWITCH(4)) 280,630,280                                         JOAT128
630   IF (IW2-2) 280,280,640                                             JOAT129
640   I3=2                                                               JOAT130
650   IF (I-IW) 660,280,280                                              JOAT131
660   DO 680 M1=I3,IW2                                                   JOAT132
      IF (ICP(M1-1)-IA(38)) 680,670,680                                  JOAT133
670   IF (ICP(M1)-IA(47)) 680,690,680                                    JOAT134
680   CONTINUE                                                           JOAT135
      GO TO 280                                                          JOAT136
690   I3=M1+1                                                            JOAT137
      IW2=IW2+1                                                          JOAT138
      I=I+1                                                              JOAT139
      I13=IW2                                                            JOAT140
      DO 700 M1=I3,IW2                                                   JOAT141
      ICP(I13)=ICP(I13-1)                                                JOAT142
700   I13=I13-1                                                          JOAT143
      GO TO 650                                                          JOAT144
710   DO 720 I3=1,IW2                                                    JOAT145
720   ICT(I3)=ICP(I3)                                                    JOAT146
      IT=IW2-1                                                           JOAT147
      I13=2                                                              JOAT148
730   K7=1                                                               JOAT149
      DO 760 I3=I13,IT                                                   JOAT150
      K9=I3                                                              JOAT151
      IF (ICT(I3)-IA(38)) 760,740,760                                    JOAT152
740   IF (ICT(I3-1)-IA(47)) 750,850,750                                  JOAT153
750   IF (ICT(I3+1)-IA(47)) 760,840,760                                  JOAT154
760   CONTINUE                                                           JOAT155
      IF (I13-2) 460,460,770                                             JOAT156
770   IF (I-IW) 780,570,570                                              JOAT157
780   DO 810 M1=K7,IT                                                    JOAT158
      K9=M1                                                              JOAT159
      IF (ICT(M1)-IA(47)) 810,790,810                                    JOAT160
790   IF (ICT(M1+1)-IA(47)) 800,810,800                                  JOAT161
800   IF (ICT(M1-1)-IA(47)) 820,810,820                                  JOAT162
810   CONTINUE                                                           JOAT163
      GO TO 570                                                          JOAT164
820   K7=IW2                                                             JOAT165
      DO 830 I3=K9,IW2                                                   JOAT166
      ICT(K7+1)=ICT(K7)                                                  JOAT167
830   K7=K7-1                                                            JOAT168
      IW2=IW2+1                                                          JOAT169
      IT=IT+1                                                            JOAT170
      I=I+1                                                              JOAT171
      K7=K9+2                                                            JOAT172
      GO TO 770                                                          JOAT173
840   K8=K9+1                                                            JOAT174
      I13=K9+1                                                           JOAT175
      GO TO 860                                                          JOAT176
850   K8=K9-1                                                            JOAT177
860   K7=IW2                                                             JOAT178
      I13=K9+2                                                           JOAT179
```

```
      DO 870 I3=K8,IW2                                                  JOAT180
      ICT(K7+1)=ICT(K7)                                                 JOAT181
870   K7=K7-1                                                           JOAT182
      IW2=IW2+1                                                         JOAT183
      IT=IT+1                                                           JOAT184
      I=I+1                                                             JOAT185
      IF (I-IW) 730,570,570                                             JOAT186
C                                                                       JOAT187
880   FORMAT (21H1K1 HAS BECOME ZERO. )                                 JOAT188
890   FORMAT (26H1IW CANNOT BE LESS THAN I )                            JOAT189
      END                                                               JOAT190-
C
C
      SUBROUTINE COWNIT(ICHAR,IT)                                       JCIT0010
      COMMON ITAPE,IRTAPE,IOTAPE,IPTAPE                                 JCIT0020
      COMMON SWITCH,N,N1,N2,NOPAR,IW2,ITEST,IJUST,IW,IA,ICOL,ICP,ICT,IC JCIT0030
       DIMENSION ICOL(80),ICP(360),ICT(360),IC(8002),IA(80),SWITCH(100) JCIT0040
      IF (SWITCH(2) -.9999)  3,3,1                                      JCIT0050
1     N3 = SWITCH(2) + 1.E-4                                            JCIT0060
      DO 2 I=1,N3                                                       JCIT0070
      IF (ICHAR - IA(I+60)) 2,4,2                                       JCIT0080
2     CONTINUE                                                          JCIT0090
3     IT = 1                                                            JCIT0100
      GO TO 5                                                           JCIT0110
4     IT = 0                                                            JCIT0120
5     RETURN                                                            JCIT0130
       END                                                              JCIT0140
C
       SUBROUTINE IOSAME                                                JSAM0010
      COMMON ITAPE,IRTAPE,IOTAPE,IPTAPE                                 JSAM0020
      COMMON SWITCH,N,N1,N2,NOPAR,IW2,ITEST,IJUST,IW,IA,ICOL,ICP,ICT,IC JSAM0030
       DIMENSION ICOL(80),ICP(360),ICT(360),IC(8002),IA(80),SWITCH(100) JSAM0040
      N=0                                                               JSAM0050
      N1=0                                                              JSAM0060
      N2=0                                                              JSAM0070
      NOPAR=0                                                           JSAM0080
      IF (ITEST) 2,3,2                                                  JSAM0090
1     FORMAT (80A1)                                                     JSAM0100
2     WRITE (IPTAPE,1) (ICOL(J),J=1,80)                                 JSAM0110
3     WRITE (IOTAPE,4) (ICOL(J),J=1,80)                                 JSAM0120
4     FORMAT (1X,80A1)                                                  JSAM0130
      RETURN                                                            JSAM0140
       END                                                              JSAM0150
C
      SUBROUTINE IOCENT                                                 JCEN0010
       COMMON ITAPE,IRTAPE,IPTAPE,IOTAPE                                JCEN0020
      COMMON SWITCH,N,N1,N2,NOPAR,IW2,ITEST,IJUST,IW,IA,ICOL,ICP,ICT,IC JCEN0030
       DIMENSION ICOL(80),ICP(360),ICT(360),IC(8002),IA(80),SWITCH(100) JCEN0040
      DO 1 I=1,70                                                       JCEN0050
1     ICP(I)=IA(47)                                                     JCEN0060
      DO 2 I=1,70                                                       JCEN0070
      IF (ICOL(I) - IA(47)) 5,2,5                                       JCEN0080
2     CONTINUE                                                          JCEN0090
3     CALL IOSAME                                                       JCEN0100
4     RETURN                                                            JCEN0110
```

```
5     M1=I                                                                  JCEN0120
      DO 7 I=1,70                                                           JCEN0130
      M2=71-I                                                               JCEN0140
      IF (ICOL(M2) - IA(47)) 6,7,6                                          JCEN0150
6     IF (ICOL(M2) - IA(39)) 8,7,8                                          JCEN0160
7     CONTINUE                                                              JCEN0170
      GO TO 3                                                               JCEN0180
8     M= M2-M1+1                                                            JCEN0190
      I2=(IW-M)/2                                                           JCEN0200
      IF (I2) 9,9,10                                                        JCEN0210
9     M3=1                                                                  JCEN0220
      GO TO 12                                                              JCEN0230
10    IF (70-I2-M) 11,11,16                                                 JCEN0240
11    M3=71-M                                                               JCEN0250
12    DO 13 I=M1,M2                                                         JCEN0260
      ICP(M3)=ICOL(I)                                                       JCEN0270
13    M3=M3+1                                                               JCEN0280
       ICP(M3)=IA(39)                                                       JCEN0290
      ICP(71)=IA(39)                                                        JCEN0300
      ICP(72)=IA(3)                                                         JCEN0310
14    DO 15 I=1,72                                                          JCEN0320
15    ICOL(I)=ICP(I)                                                        JCEN0330
      GO TO 3                                                               JCEN0340
16     J1=0                                                                 JCEN0350
       DO 18 I=M1,M2                                                        JCEN0360
      CALL COWNIT(ICOL(I),J)                                                JCEN0370
      IF (J) 18,17,18                                                       JCEN0380
17    J1=J1+1                                                               JCEN0390
18     CONTINUE                                                             JCEN0410
       IF (J1) 19,19,20                                                     JCEN0420
19     M3=I2+1                                                              JCEN0430
      GO TO 12                                                              JCEN0440
20    M3=I2+(J1/2) +1                                                       JCEN0450
      IF (70-M3-M) 11,11,12                                                 JCEN0460
       END                                                                  JCEN0470
C
       SUBROUTINE INDENT                                                    JIND0010
      COMMON ITAPE,IRTAPE,IOTAPE,IPTAPE                                     JIND0020
      COMMON SWITCH,N,N1,N2,NOPAR,IW2,ITEST,IJUST,IW,IA,ICOL,ICP,ICT,IC     JIND0030
       DIMENSION ICOL(80),ICP(360),ICT(360),IC(8002),IA(80),SWITCH(100)     JIND0040
      SWITCH(1) = -1.*SWITCH(1)                                             JIND0050
      DO 1 M1=1,71                                                          JIND0060
      IF (ICOL(M1) - IA(47)) 6,1,6                                          JIND0070
1     CONTINUE                                                              JIND0080
3     IW=J                                                                  JIND0090
16    N=0                                                                   JIND0100
      N1=0                                                                  JIND0110
      N2=0                                                                  JIND0120
      NOPAR=0                                                               JIND0130
      RETURN                                                                JIND0140
4     WRITE (IOTAPE,5) (ICOL(I),I=1,80),(ICOL(I),I=M1,M2)                   JIND0150
5     FORMAT (1HO,80A1,/53H THE ABOVE CARD CONTAINS AN INDENT OF ILLEGALJIND0160
     1 SIZE =    ,80A1//      )                                            JIND0170
      SWITCH(1) = -1.*SWITCH(1)                                             JIND0180
      GO TO 16                                                              JIND0190
```

```
6     M2=M1+2                                                          JIND0200
      J=0                                                              JIND0210
      IF (ICOL(M2) - IA(47)) 7,9,7                                     JIND0220
7     IF (ICOL(M1) - IA(28)) 4,8,4                                     JIND0230
8     J=100                                                            JIND0240
      M1=M1+1                                                          JIND0250
      GO TO 10                                                         JIND0260
9     M2=M1+1                                                          JIND0270
10    DO 11 I=27,36                                                    JIND0280
      IF (ICOL(M2) - IA(I)) 11,12,11                                   JIND0290
11    CONTINUE                                                         JIND0300
      GO TO 4                                                          JIND0310
12    J = J + I - 27                                                   JIND0320
      DO 13 I=27,36                                                    JIND0330
      IF (ICOL(M1) - IA(I)) 13,14,13                                   JIND0340
13    CONTINUE                                                         JIND0350
      GO TO 4                                                          JIND0360
14    J = 10*(I-27) + J                                                JIND0370
      IF (J-120) 15,15,4                                               JIND0380
15    IF (J-20) 4,3,3                                                  JIND0390
       END                                                             JIND0400
C
      SUBROUTINE RABOUT                                                JRAB0010
      COMMON ITAPE,IRTAPE,IOTAPE, IPTAPE                               JRAB0020
      COMMON SWITCH,N,N1,N2,NOPAR,IW2,ITEST,IJUST,IW,IA,ICOL,ICP,ICT,IC JRAB0030
       DIMENSION ICOL(80),ICP(360),ICT(360),IC(8002),IA(80),SWITCH(100) JRAB0040
      DO 1 M1=1,71                                                     JRAB0050
      IF (ICOL(M1) - IA(47)) 4,1,4                                     JRAB0060
1     CONTINUE                                                         JRAB0070
2     CALL IOSAME                                                      JRAB0080
3     N=0                                                              JRAB0090
      N1=0                                                             JRAB0100
      N2=0                                                             JRAB0110
       NOPAR=0                                                         JRAB0120
      RETURN                                                           JRAB0130
8     WRITE (IOTAPE,17) (ICOL(I),I=1,80),(ICOL(I),I=M1,M2)             JRAB0140
9     FORMAT (1H0,80A1,/67H JUSTIFY WILL NOT ALLOW A RUN AROUND FROM THEJRAB0150
     1 ABOVE CARD OF SIZE =     ,80A1//)                               JRAB0155
      GO TO 3                                                          JRAB0160
4     M2= M1+1                                                         JRAB0170
      M=2                                                              JRAB0180
      IF (ICOL(M2) - IA(47)) 11,5,11                                   JRAB0190
5     M=1                                                              JRAB0200
       M2=M1                                                           JRAB0210
11    DO 12 I=27,36                                                    JRAB0220
      IF (ICOL(M2) - IA(I)) 12,13,12                                   JRAB0230
12    CONTINUE                                                         JRAB0240
      GO TO 8                                                          JRAB0250
13     J = I-27                                                        JRAB0260
      IF (M-1) 3,2,14                                                  JRAB0270
14    DO 15 I=27,36                                                    JRAB0280
      IF (ICOL(M1) - IA(I)) 15,16,15                                   JRAB0290
15    CONTINUE                                                         JRAB0300
      GO TO 8                                                          JRAB0310
16    J=10*(I-27) + J                                                  JRAB0320
```

```
20    WRITE (IOTAPE,17) IA(39)                                                JRAB0330
      WRITE (IOTAPE,17) IA(39)                                                JRAB0340
17    FORMAT (1X,80A1)                                                        JRAB0350
      WRITE (IOTAPE,18)                                                       JRAB0360
18     FORMAT(51H * * * * * * * * * * * * * * * * * * * * * * *               JRAB0370
      IF(J-2) 8,22,22                                                         JRAB0380
22     M=J/2                                                                  JRAB0390
      DO 19 I=1,M                                                             JRAB0400
19    WRITE (IOTAPE,17)  IA(39)                                               JRAB0410
      ICOL(72) =IA(4)                                                         JRAB0420
      ICOL(71)=IA(39)                                                         JRAB0430
      WRITE (IOTAPE,17) (ICOL(I),I=1,72)                                      JRAB0440
       M=M+2                                                                  JRAB0450
      DO 21 I=M,J                                                             JRAB0460
21    WRITE (IOTAPE,17) IA(39)                                                JRAB0470
      WRITE (IOTAPE,18)                                                       JRAB0480
      WRITE (IOTAPE,17) IA(39)                                                JRAB0490
      WRITE (IOTAPE,17)  IA(39)                                               JRAB0500
      IF (ITEST) 25,3,25                                                      JRAB0510
25    M=M-2                                                                   JRAB0520
      WRITE (IPTAPE,17) IA(39)                                                JRAB0530
      WRITE (IPTAPE,17) IA(39)                                                JRAB0540
      WRITE (IPTAPE,18)                                                       JRAB0550
      DO 26 I=1,M                                                             JRAB0560
26    WRITE (IPTAPE,17)  IA(39)                                               JRAB0570
      WRITE (IPTAPE,30) (ICOL(I),I=1,72)                                      JRAB0580
30        FORMAT(80A1)                                                        JRAB0590
      M=M+2                                                                   JRAB0600
      DO 27 I=M,J                                                             JRAB0610
27    WRITE (IPTAPE,17) IA(39)                                                JRAB0620
      WRITE (IPTAPE,18)                                                       JRAB0630
      WRITE (IPTAPE,17) IA(39)                                                JRAB0640
      WRITE (IPTAPE,17) IA(39)                                                JRAB0650
      GO TO 3                                                                 JRAB0660
       END                                                                    JRAB0670
C
C
       SUBROUTINE UNLOCK                                                      JULK0010
      COMMON ITAPE,IRTAPE,IOTAPE,IPTAPE                                       JULK0020
      COMMON SWITCH,N,N1,N2,NOPAR,IW2,ITEST,IJUST,IW,IA,ICOL,ICP,ICT,IC       JULK0030
       DIMENSION ICOL(80),ICP(360),ICT(360),IC(8002),IA(80),SWITCH(100)       JULK0040
      J=1                                                                     JULK0050
1     IF (IC(J)-IA(46)) 2,4,2                                                 JULK0060
2     J=J+1                                                                   JULK0070
      IF (J-(N  +1)) 1,3,3                                                    JULK0080
3     RETURN                                                                  JULK0090
4     IF (IC(J+1)-IA(46)) 7,5,7                                               JULK0100
5     J=J+2                                                                   JULK0110
      DO 6 I=J,N                                                              JULK0120
6     IC(I-1)=IC(I)                                                           JULK0130
      GO TO 10                                                                JULK0140
7     IF (IC(J+2)-IA(46)) 11,8,11                                             JULK0150
8     J=J+3                                                                   JULK0160
      DO 9 I=J,N                                                              JULK0170
9     IC(I-1)=IC(I)                                                           JULK0180
```

```
10    N=N-1                                                          JULK0190
      IC(N+1)=IA(47)                                                 JULK0200
      J=J-1                                                          JULK0210
      GO TO 1                                                        JULK0220
11     IF ( N     -7999) 13,13,12                                    JULK0230
12     WRITE (IOTAPE,1001) (IC(K),K=1,119)                           JULK0240
1001  FORMAT(1X,119A1,/94H THE ABOVE LINE BEING CHANGED FROM SHIFT AND LJULK0250
     1OCK HAS EXCEEDED THE 8000 CHARACTERS LINE LIMIT./37H WORK WAS HALTJULK0260
     2ED AT 8000 CHARACTERS.     )                                  JULK0270
      GO TO 3                                                        JULK0280
13     N=N+1                                                         JULK0290
      J=J+3                                                          JULK0300
      K=N                                                            JULK0310
       DO 14 L=J,N                                                   JULK0320
      IC(K)=IC(K-1)                                                  JULK0330
14      K=K-1                                                        JULK0340
      J=J-1                                                          JULK0350
      IC(J)=IA(46)                                                   JULK0360
      GO TO 4                                                        JULK0370
      END                                                            JULK0380
C
       SUBROUTINE LOCK                                               JLOK0010
      COMMON ITAPE,IRTAPE,IOTAPE,IPTAPE                              JLOK0020
      COMMON SWITCH,N,N1,N2,NOPAR,IW2,ITEST,IJUST,IW,IA,ICOL,ICP,ICT,IC JLOK0030
       DIMENSION ICOL(80),ICP(360),ICT(360),IC(8002),IA(80),SWITCH(100) JLOK0040
      J=1                                                            JLOK0050
1     IF (IC(J)-IA(46)) 2,5,2                                        JLOK0060
2     J=J+1                                                          JLOK0070
      IF (J-(N  +1)) 1,1,4                                           JLOK0080
3     WRITE (IOTAPE,1001) (IC(K),K=1,119)                            JLOK0090
1001  FORMAT (1X,119A1,/97H THE ABOVE PARAGRAPH BEING CHANGED TO SHIFT AJLOK0100
     1ND LOCK HAS EXCEEDED THE 8000 CHARACTERS LINE LIMIT. /37H WORK WASJLOK0110
     2 HALTED AT 8000 CHARACTERS.     )                             JLOK0120
4     RETURN                                                         JLOK0130
5     J=J+2                                                          JLOK0140
6     IF (IC(J)-IA(46)) 10,7,10                                      JLOK0150
7     IF (IC(J+1)-IA(47)) 75,70,75                                   JLOK0160
70    J=J+1                                                          JLOK0170
      GO TO 1                                                        JLOK0180
75    N=N-1                                                          JLOK0190
      DO 8 K=J,N                                                     JLOK0200
8     IC(K)=IC(K+1)                                                  JLOK0210
      IC(N+1)=IA(47)                                                 JLOK0220
      J=J+1                                                          JLOK0230
      IF (J-N  )6,6,9                                                JLOK0240
9     N=N+1                                                          JLOK0250
      IC(N  )=IA(46)                                                 JLOK0260
      GO TO 4                                                        JLOK0270
10    IF (N  -7999) 11,9,3                                           JLOK0280
11    N=N+1                                                          JLOK0290
      J=J+1                                                          JLOK0300
      K=N                                                            JLOK0310
      DO 12 L=J,N                                                    JLOK0320
      IC(K)=IC(K-1)                                                  JLOK0330
12    K=K-1                                                          JLOK0340
```

```
      IC(J-1)=IA(46)                                                    JL0K0350
      GO TO 1                                                           JL0K0360
      END                                                               JL0K0370
C
C
      SUBROUTINE PRINTP                                                 JPRI0010
      COMMON ITAPE,IRTAPE,IOTAPE, IPTAPE                                JPRI0020
      COMMON SWITCH,N,N1,N2,NOPAR,IW2,ITEST,IJUST,IW,IA,ICOL,ICP,ICT,IC JPRI0030
       DIMENSION ICOL(80),ICP(360),ICT(360),IC(8002),IA(80),SWITCH(100) JPRI0040
      IF (SWITCH(1)) 1,1,3                                              JPRI0050
1     CALL PRINT2                                                       JPRI0060
2     RETURN                                                            JPRI0070
3     CALL PRINT1                                                       JPRI0080
       GO TO 2                                                          JPRI0090
       END                                                              JPRI0100
C
       SUBROUTINE PRINT1                                                JPR10010
      COMMON ITAPE,IRTAPE,IOTAPE,IPTAPE                                 JPR10020
      COMMON SWITCH,N,N1,N2,NOPAR,IW2,ITEST,IJUST,IW,IA,ICOL,ICP,ICT,IC JPR10030
       DIMENSION ICOL(80),ICP(360),ICT(360),IC(8002),IA(80),SWITCH(100) JPR10040
1     FORMAT (1X,119A1)                                                 JPR10050
2     FORMAT (120A1)                                                    JPR10060
      K=1                                                               JPR10070
      K1=118                                                            JPR10080
      IF (IW2-K1-1) 10,10,30                                            JPR10090
10    WRITE (IOTAPE,1) (ICP(J),J=K,IW2)                                 JPR10100
      IF (ITEST) 80,20,80                                               JPR10110
20    RETURN                                                            JPR10120
30    J=K1                                                              JPR10130
      DO 40 I=K,K1                                                      JPR10140
      IF (ICP(J) - IA(47)) 40,50,40                                     JPR10150
40    J=J-1                                                             JPR10160
      GO TO 60                                                          JPR10170
50    K1=J                                                              JPR10180
60    WRITE (IOTAPE,1) (ICP(J),J=K,K1)                                  JPR10190
70    K=K1+1                                                            JPR10200
      K1=K1+118                                                         JPR10210
      IF (IW2-K1-1) 10,10,30                                            JPR10220
80    K=1                                                               JPR10230
      K1=69                                                             JPR10240
      IF (IW2-K1-1) 90,90,100                                           JPR10250
90    WRITE (IPTAPE,2) (ICP(J),J=K,IW2),IA(39)                          JPR10260
      GO TO 20                                                          JPR10270
100   J=K1                                                              JPR10280
      IF (ICP(K1) - IA(47)) 105,150,105                                 JPR10290
105   DO 110 I=K,K1                                                     JPR10300
      IF (ICP(J) - IA(47)) 110,120,110                                  JPR10310
110   J=J-1                                                             JPR10320
      GO TO 130                                                         JPR10330
120   K1=J                                                              JPR10340
130   WRITE (IPTAPE,2) (ICP(J),J=K,K1),IA(45),IA(39)                    JPR10350
140   K=K1+1                                                            JPR10360
      K1=K1+69                                                          JPR10370
      IF (IW2-K1-1) 90,90,100                                           JPR10380
150   IF (ICP(K1+1) - IA(47)) 130,151,130                               JPR10390
```

```
151   DO 160 I=K,K1                                                    JPR10400
      IF (ICP(J) - IA(47)) 170,160,170                                 JPR10410
160    J=J-1                                                           JPR10420
      GO TO 130                                                        JPR10430
170   K1=J                                                             JPR10440
      GO TO 105                                                        JPR10450
       END                                                             JPR10460
C
       SUBROUTINE PRINT2                                               JPR20010
      COMMON ITAPE,IRTAPE,IOTAPE,IPTAPE                                JPR20020
      COMMON SWITCH,N,N1,N2,NOPAR,IW2,ITEST,IJUST,IW,IA,ICOL,ICP,ICT,IC JPR20030
       DIMENSION ICOL(80),ICP(360),ICT(360),IC(8002),IA(80),SWITCH(100) JPR20040
1     FORMAT (1X,119A1)                                                JPR20050
2     FORMAT (120A1)                                                   JPR20060
      K=1                                                              JPR20070
      K1=117                                                           JPR20080
      IF (IW2-K1-1) 10,10,30                                           JPR20090
10    IF (K-1) 20,14,16                                                JPR20100
14    WRITE (IOTAPE,1) IA(48),(ICP(J),J=K,IW2)                         JPR20110
      GO TO 18                                                         JPR20120
16    WRITE (IOTAPE,1) (ICP(J),J=K,IW2)                                JPR20130
18    IF (ITEST) 80,20,80                                              JPR20140
20    RETURN                                                           JPR20150
30    J=K1                                                             JPR20160
      DO 40 I=K,K1                                                     JPR20170
      IF (ICP(J) - IA(47)) 40,50,40                                    JPR20180
40    J=J-1                                                            JPR20190
      GO TO 60                                                         JPR20200
50    K1=J                                                             JPR20210
60    IF (K-1) 20,64,66                                                JPR20220
64    WRITE (IOTAPE,1) IA(48),(ICP(J),J=K,K1)                          JPR20230
      GO TO 70                                                         JPR20240
66    WRITE (IOTAPE,1) (ICP(J),J=K,K1)                                 JPR20250
70    K=K1+1                                                           JPR20260
      K1=K1+117                                                        JPR20270
      IF (IW2-K1-1) 16,16,30                                           JPR20280
80    K=1                                                              JPR20290
      K1=68                                                            JPR20300
      IF (IW2-K1-1) 90,90,100                                          JPR20310
90    IF (K-1) 20,92,94                                                JPR20320
92    WRITE (IPTAPE,2) IA(48),(ICP(J),J=K,IW2),IA(39)                  JPR20330
      GO TO 20                                                         JPR20340
94    WRITE (IPTAPE,2) (ICP(J),J=K,IW2),IA(39)                         JPR20350
      GO TO 20                                                         JPR20360
100   J=K1                                                             JPR20370
      IF (ICP(K1) - IA(47)) 105,150,105                                JPR20380
105   DO 110 I=K,K1                                                    JPR20390
      IF (ICP(J) - IA(47)) 110,120,110                                 JPR20400
110   J=J-1                                                            JPR20410
      GO TO 130                                                        JPR20420
120   K1=J                                                             JPR20430
130   IF (K-1) 20,132,134                                              JPR20440
132   WRITE (IPTAPE,2) IA(48),(ICP(J),J=K,K1),IA(45),IA(39)            JPR20450
      GO TO 140                                                        JPR20460
134   WRITE (IPTAPE,2) (ICP(J),J=K,K1),IA(45),IA(39)                   JPR20470
```

```
140   K=K1+1                                                                JPR20480
      K1=K1+68                                                              JPR20490
      IF (IW2-K1-1) 94,94,100                                               JPR20500
150   IF (ICP(K1+1) - IA(47)) 130,151,130                                   JPR20510
151   DO 160 I=K,K1                                                         JPR20520
      IF (ICP(J) - IA(47)) 170,160,170                                      JPR20530
160    J=J-1                                                                JPR20540
      GO TO 130                                                             JPR20550
170   K1=J                                                                  JPR20560
      GO TO 105                                                             JPR20570
       END                                                                  JPR20580
C
C
      SUBROUTINE CHECK3(K)                                                  JCHK0010
      COMMON ITAPE,IRTAPE,IOTAPE,IPTAPE                                     JCHK0020
      COMMON SWITCH,N,N1,N2,NOPAR,IW2,ITEST,IJUST,IW,IA,ICOL,ICP,ICT,IC     JCHK0030
       DIMENSION ICOL(80),ICP(360),ICT(360),IC(8002),IA(80),SWITCH(100)     JCHK0040
      DO 10 I=1,26                                                          JCHK0050
      IF (IA(I) - ICOL(I)) 40,10,40                                         JCHK0060
10    CONTINUE                                                              JCHK0070
      K=1                                                                   JCHK0080
      WRITE (IOTAPE,19) (IA(I),I=1,80)                                      JCHK0090
      IF (ITEST) 20,50,20                                                   JCHK0100
20    WRITE (IPTAPE,29) (IA(I),I=1,80)                                      JCHK0110
      IF (IPTAPE - 6) 50,50,30                                              JCHK0120
30    END FILE IPTAPE                                                       JCHK0130
      GO TO 50                                                              JCHK0140
40    K=0                                                                   JCHK0150
50    RETURN                                                                JCHK0160
19    FORMAT (1X,80A1)                                                      JCHK0170
29    FORMAT (80A1)                                                         JCHK0180
      END                                                                   JCHK0190
C
C
```


☆U.S. GOVERNMENT PRINTING OFFICE : 1969 O - 330-47

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, $6.00; foreign, $7.25*.

• 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, $2.25; foreign, $2.75*.

• 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, $2.75; foreign, $3.50*.

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*.

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

NONPERIODICALS

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

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

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

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

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

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

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

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

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

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

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