Royce, Christopher L.; Fletcher, John S.; Risser, Paul G.; McFarlane, James C.; Benenati, Frank E. "PHYTOTOX: A Database Dealing with the Effect of Organic Chemicals on Terrestrial Vascular Plants". *J. Chem. Inf. Comput. Sci.* **1984**, *24*, 7-10.
100(9):62823n CASCT 03
Chen, H. R.; Dayhoff, M. O.; Barker, W. C.; Hunt, L. T.; Yeh, L. S.; George, D. G.; Orcutt, B. C. "Nucleic Acid Sequence Database. V: Completely Sequenced Genomes". *DNA* **1983**, *2*(4), 275-280.

REFERENCES AND NOTES

(1) Prior to 1972, section 1 of *Chemical Abstracts* covered history, education, and documentation.
(2) *Current Awareness Profile on Chemical Information*; ISSN 0276-8712; Wiggins, Gary, Ed.; Chemical Information Center: Bloomington, IN, 1981-1985. The publication was issued quarterly in all years but the first and was based on a CAS Individual Search Service SDI profile covering vol. 94-103 (1981-1985) of *CA*.
(3) Appendix I originally included two 1984 articles from *Online Review* and a 1984 *Online* article. However, these were removed when it was discovered that Chemical Abstracts Service added those references to the *CA* database in the summer of 1986.
(4) "CAS 1986 DDS: Directory of Publications; Serials Currently Received at CAS". Chemical Abstracts Service: Columbus, OH, 1986. It should be noted that *Online* was previously included in *CA*'s coverage. Prior to 1982, there are eight citations to the journal in the STN International CA File. Neither *Database* nor *Online Review* had any entries in the CA File until the summer of 1986 when four references to *Database*, two to *Online*, and six to *Online Review* were added.
(5) It cannot be argued that CAS's editorial policy excludes articles of the type listed in Appendix I. One can, for example, find the following reference in *CA* section 20, vol. 100, issue 9, abstract number 67418u: Revesz, Gabrielle S.; Cassidy, Phoebe A. "Modern Literature Searching; the Case of C_4". *CHEMTECH* **1984**, *14*, 18-25. That article discusses online literature searching, including the steps and logic involved, as well as the search costs and the databases used. These are topics typically found in articles published in *Online*, *Online Review*, or *Database*.
(6) Ayers, Jerry B. "Journal and Conference Proceedings of Chemical History, Education, and Documentation". *J. Chem. Inf. Comput. Sci.* **1981**, *21*, 71-72. Ayers, Jerry B. "The Journals of Chemical History, Education, and Documentation". *J. Chem. Doc.* **1971**, *11*, 12-13.
(7) The fact that *Nucleic Acids Research* appears among the top six journals of chemical information science which publish articles in English points out one of the pitfalls in such ranked lists. All of the references are taken from a special issue (vol. 12, issue 1, parts 1-2) that was devoted to the applications of computers to research on nucleic acids.
(8) Curiously, Chemical Abstracts Service neglects to include some relevant articles from *Journal of Chemical Education* in its chemical information science coverage. For example, none of the following articles had been abstracted by CAS as of Sept 20, 1986: Gorin, George "An Approach to Teaching Chemical Information Retrieval". *J. Chem. Educ.* **1982**, *59*, 991-994. Wiggins, Gary "The Indiana University Chemical Information Center Program of Chemical Literature Instruction". *J. Chem. Educ.* **1982**, *59*, 994-997. Hendrickson, W. A. "Library Searching; An Industrial User's Viewpoint". *J. Chem. Educ.* **1982**, *59*, 997-999. Allen, Ferne C. "Instruction in Chemical Literature; Industrial Librarian Viewpoint". *J. Chem. Educ.* **1982**, *59*, 999-1002.
(9) Only this paper of the nine listed in Appendix II has a cross-reference from section 20.

Computer as a Versatile Research Assistant

V. ZITKO

Fisheries and Environmental Sciences, Biological Station, Department of Fisheries and Oceans, St. Andrews, N. B., Canada E0G 2X0

Received May 21, 1986

Simple uses of a computer in routine research tasks are described. They include extraction of information from texts, programming by using a library of routines, combined data and word processing, and reprint indexing. The applications require only moderate experience in programming. Listings of programs and a library of routines, written mostly in a fairly standard BASIC, are included as supplementary material.

INTRODUCTION

Word processing is now an established and valuable tool in research. However, there are many specialized operations the word-processing programs were not designed to perform. Similarly, the commercial statistical and mathematical packages may not meet specific needs, and commercial database software may be too cumbersome for a relatively straightforward task of reprint indexing.

This paper describes the use of a computer in the daily activities of a researcher. It deals with entry and manipulation of text in a computer. The manipulation includes the extraction of specific information such as names and mass spectra of chemicals from a database output and assorted calculations. A library of routines is discussed, and a system for indexing of reprints is described. The applications require some knowledge of programming BASIC and general familiarity with the computer available. An HP 3000 computer (Hewlett-Packard Co., Cupertino, CA) was used, but the approaches and procedures are generally applicable and most of the programs and routines described are implementable after minor modifications on other systems.

The emphasis is on the use of the computer as a flexible tool rather than on sophisticated programming. The objective is to make versatile use of the computer by simple commands and programs and to be able to perform assorted tasks on short notice. It is often better to write a simple program to perform a specific task than to spend a lot of time developing a complex program that can handle all eventualities or working with a poorly documented commercial package.

DATA INPUT

Data discussed here consist of either text or numbers or both. They are stored in the computer in the form of text (ASCII) files (even numbers are stored as "text"). Text files are usually easier to edit and to communicate between computers than files containing numbers in a "numeric" format. Files consist of records.

With interactive applications and screen displays, it is advantageous to use text files with records 72-80 characters long. "Editor" programs, present on most computers, can perform considerable editing of such files. In this way one makes good use of the software existing on the computer.

0095-2338/87/1627-0003$01.50/0

Table I. Simple Uses of Files (See Text for Explanation)

A. Keyboard Entry

```
/A
1       capillary GC                              8248
2       fraction          complex     mixtures    8249
3       PCB               accumul     fish        8250
/KQ
```

B. Examples of Commands to Direct Output to Files

command	comment
:BUILD filename; REC=-80,,F,ASCII	build file first,
>RUN program name, OUT=filename	then run BASIC program
:FILE BASLIST=filename	compiled BASIC
:FILE FTN06=filename	FORTRAN

C. File Received from a Database

```
                    file 1; entry   1; accession no.    1
2 (CAS)  CAS Registry Number:  67-72-1
4 (MAT) common name:  hexachloroethane

                    file 1; entry   2; accession no.    2
2 (CAS)  CAS Registry Number:  126-98-7
4 (MAT) common name:  methacrylonitrile
```

D. Edited Database File

67-72-1	hexachloroethane
126-98-7	methacrylonitrile
90-13-1	1-chloronaphthalene

E. Edited Database File Sorted by Compound

67-64-1	acetone
75-86-5	acetone cyanohydrin
107-13-1	acrylonitrile

F. Use of "Ruler" To Count Columns

```
2 (CAS) CAS Registry Number:  67-72-1
.....................................................................
        1         2         3         4         5
12345678901234567890123456789012345678901234567890123456
  6
89012345678
.....................................................................
```

Data may be entered manually from the keyboard, produced as output (printout) of programs, or received from external sources such as commercial databases over the phone.

The simplest way to enter data from the keyboard is through an editor program such as QEDIT[1] on the HP 3000. The editor handles the file manipulations without much effort on the part of the user. In a specific QEDIT example, a file may be opened by the command "/A" ("/" is the QEDIT prompt). To make a file of reprints, keywords and a reprint number may be typed (Table IA).

In this example the keywords are in fixed positions (columns 10–19, 20–29, 30–39). This simplifies subsequent reading of the entries. A more complicated program could deal with keywords in any position and of variable length. The file may be closed and saved by the command "/KQ filename".

Any text and numbers that appear on the screen can be saved as a file. All one has to do is to build a file and direct the output to it. The commands to build files vary from computer to computer and so do the commands directing the output. A few examples of the latter for the HP 3000 are in Table IB.

Files received over the phone are easiest to record on a magnetic tape or diskette at the terminal. Subsequently, they may be sent to the computer via an editor program. Commercial communications packages usually handle the transfer of text files very well.

Names of files dealing with related topics should start with the same letters so that such groups of files will be listed together (computers usually list files alphabetically). In other words, one should use suffixes rather than prefixes. Also, certain letters used as suffixes may have fixed meanings such as "A" for text (ASCII), "F" for fast, "S" for source, and "C" for compiled. For example, MASSPA may stand for a text file of a BASIC program MASSP, MASSPF would be a "fast saved" version of MASSP, which can be compiled into MASSPC, etc.

Table II. Extraction of Chemical Abstracts Service (CAS) Registry Numbers and Names of Chemicals from a Database Display (Table IC Yields ID)

```
  5 FILES *
 10 DIM B$[72],B1$[20],B2$[27]
 30 ASSIGN "CESARS",1,F1
 40 IF F1=0 OR F1=1 THEN 100
 50 STOP
100 ON END #1 THEN 500
105 LINPUT #1;B$
110 IF B$[1;1]=" " THEN 105
120 B1$=B$[30;20]
130 LINPUT #1;B$
140 B2$=B$[23;27]
150 PRINT USING "20A,3X,27A";B1$,B2$
160 GOTO 105
500 END
```

DATA EDITING

One may need only certain sections of text from the text files, or one may want to correct erroneous data, add or delete some information, etc. Commercial word-processing programs can perform sophisticated editing of text files. Special editor programs such as QEDIT can perform similar tasks and are usually better suited for adding or deleting columns, duplicating lines, etc. than word processors. Highly specialized editing can be achieved by programming. At present, BASIC is the language of choice. It is easy to learn and interactive, and program checking and corrections are straightforward. Different computers store BASIC programs in different formats, but the programs are transferable in text format. For many applications described here, the slow execution rate of programs in BASIC is not important. In addition, on most systems BASIC programs can be compiled. This speeds up their execution considerably.

Reading Files with BASIC. BASIC can read text files with the LINPUT statement, one record at a time. The statement reads a record as a "string" variable (for example B$). Selected parts of the record may then be extracted by a statement defining a "substring" B1$:

B1$ = B$[position;length]

Thus B1$ = B$[10;10] is a substring of B$ from the 10th to the 19th column (10 columns long). Other versions of BASIC may have slightly different substring statements.

These two statements in conjunction with the ability to open a file allow customized data manipulation. For example, the input received over the phone from the Chemical Evaluation Search and Retrieval System (CESARS) component of Chemical Information Systems (CIS) Inc.[2,3] is in Table IC. One may want to retain only the Chemical Abstracts Service (CAS) Registry Number and name (Table ID).

Editing Files with BASIC. The editing above cannot be done with QEDIT, but just a few statements in BASIC will accomplish it (Table II).

Statements 5–50 handle the opening of a file CESARS, which contains the information received from the database. The rest of the program retains just the CAS Registry Number and the name of each chemical. This yields a considerably smaller file.

The new file may be edited and sorted. The editing includes the separation of numerical prefixes from the names and the conversion of the first characters of the names to lower case so that these will not interfere with the subsequent alphabetical arrangement. On the HP 3000 the editing is done conveniently

Table III. Display of Mass Spectra by the Chemical Information Systems, Inc. (CIS), Database Mass Spectral Search System (MSSS)[2,3]

```
CAS RN     QI   MW   formula, Names
1079-96-5  616  190  C14H22
                     Benzene, 1,3-bis(1-methylpropyl)- (9CI)
                     Benzene, m-di-sec-butyl- (8CI)
  Instrument:  ; Inlet:  ; Source temp.:  0 C;  eV:  0
  Contributor: AMERICAN PETROLEUM INSTITUTE
  There are 92 peaks in this spectrum. List (Y/N)? Y

  User: the min & max m/z are: 50,200

  User: type all m/z's with intensity > 2
```

m/z	INT	m/z	INT	m/z	INT	m/z	INT	m/z	INT
50	2	51	4	53	3	55	7	56	2
57	30	58	2	63	2	65	3	72	2
77	5	78	2	79	3	91	15	103	2
105	13	106	2	115	4	116	2	117	6
119	4	128	2	131	2	132	2	133	3
147	6	161	100	162	11	175	2	190	15

Table IV. Extraction of the Mass Spectrum from an MSSS File

```
110    ON END #F THEN 200
112    LINPUT #F;B$
114    IF B$[4;3]<>"CAS" THEN 112
116    LINPUT #F;C1$
120    LINPUT #F;C2$
121    PRINT C1$
122    PRINT C2$
124    LINPUT #1;B$
126    IF B$[2;3]<>"m/z" THEN 124
127    LINPUT #F;B$
128    IF B$="-------" THEN 200
131    PRINT B$
132    GOTO 127
200    END
```

by QEDIT and the sorting by a special program, SORT, available on the system. A part of the product is in Table IE.

To simplify the counting of columns, one may have a "measuring stick" file that can be inserted by QEDIT in required places of the main file and deleted when no longer needed.

For example, to locate the position of the CAS number in the data received from CIS, the measuring stick may be inserted next to the record containing the CAS number (Table IF). The number starts in column 31, and since nothing else is on this line, B$[31;20] will retrieve it. The length "20" is an insurance against truncation of long numbers.

Specific characteristics of data may be used to find the desired information. For example, the format of mass spectra retrieved from the CIS database Mass Spectral Search System (MSSS)[2,3] is given in Table III.

One usually wants to retain the formula line, one name, and the mass spectrum (m/z and INT). To do this, one would retain the second and third records (lines), then keep reading without retention until finding "m/z" in columns 2–4, and then read and retain all the records. The process will stop when a dotted line is found. The program for this (Table IV) assumes that the appropriate file is opened as #F and must be used in conjunction with such statements as 5–50 in Table II.

Multiline Functions. If the reading of a certain type of file is a more frequent task, it may be advantageous to recast the program in Table IV in a "multiline function" format. Multiline functions are a convenient tool of HP 3000 BASIC[4] and are increasingly being used by other versions of BASIC as well. They are similar to subroutines in FORTRAN. Multiline functions can be incorporated in a main program and can use "local" variable names. This means that the variable names may be the same as the names of other variables in the main program.

When a particular version of BASIC does not have multiline functions, subroutine format may be used. Unfortunately, in this case, a careful selection of the names of the variables must be made. A multiline function program for reading mass spectra is given in Appendix 1 (all appendixes are in the supplementary material).

Table V. Conversion of Mass Spectrum into Numeric Variables

```
700   DEF FNN(F,J,M[*],Y[*])
710     REM Read current spectrum from file #F into numerical
720     REM arrays Mass[*],Intensity[*]
730     REM Input - F file number, the pointer must be current
740     REM Output - J-number of M[ ],Y[ ] pairs
750     REM      M[J],Y[J]
760     INTEGER I
770     DIM B$[60],B1$[3]
780     MAT M=ZER
790     MAT Y=ZER
800     J=0
810     LINPUT #F;B$
820     IF B$[2;3]="  " AND J=0 THEN 810
830     IF B$[2;3]="  " AND J<>0 THEN 930
840     FOR I=2 to 50 STEP 12
850       J=J+1
860       B1$=B$[I;3]
870       IF B1$="  " THEN 930
880       CONVERT B1$ TO M[J]
890       B1$=B$[I+5;3]
900       CONVERT B1$ TO Y[J]
910     NEXT I
920     GOTO 810
930     REDIM M[J],Y[J]
940     J=J-1
945     RETURN J
950   FNEND
```

All the programs so far have dealt with text (strings) rather than numbers. In order to perform calculations one has to convert strings of "numeric data", such as the mass spectral data above, to numbers. This may be needed, for example, for the characterization of the similarity of two mass spectra by the Euclidean distance calculation. The HP 3000 BASIC statement to convert a string variable to a number is "CONVERT B$ to B" (this statement is different in other forms of BASIC). A program converting strings of mass spectral data from the MSSS format to numbers is given in the form of a multiline function FNN in Table V. Note that the MSSS format contains m/z values in fields three columns wide starting in columns 2, 14, 26, 38, and 50, that is, 12 columns apart. The intensities are also in fields three columns wide and are offset five columns (7, 19, ..., 55).

For a quick keyboard entry of text files for subsequent conversion to numbers, it is convenient to separate the entries by one or several spaces and have a multiline function to read such strings and to perform the conversions. In this format, data are entered readily without mistakes and the painstaking counting of columns is avoided (see below and the function FNF in Appendix 2).

LIBRARY OF ROUTINES

One can build a "personal" library of routines. The multiline function format is particularly convenient because it permits naming of variables independently of the main program. This allows versatile composition of programs for specific tasks. An example of such a library, containing frequently needed routines (Table VI), is in Appendix 2.

To prepare a program, one selects routines from the library, places them into a text file, and, if necessary, edits the statement numbers and function names to eliminate duplication and to maintain the numbering sequence. In the HP 3000 BASIC the multiline function names are limited to one character, and a duplication of names in the library is likely to occur. The editing of statement numbers must also be done carefully in the body of the functions. To minimize problems one should write the functions with the DO constructs

Table VI. Contents of the Library of Routines (Appendix 2)[a]

```
file operations:
   open an existing file
   build and open a text file from a BASIC program
principal components analysis:
   eigenvalues and vectors of a symmetric matrix
   abstract row matrix
   predictions
geometry:
   vector product
   distance of point from line
   Euclidean distance
   projection of vector on plane
linear regression:
   standard algorithm
   algorithm with matrix operations
   multiple regression
   geometric regression
   quadratic regression
miscellaneous:
   MODULO function
   mean and standard deviation
   squared deviation of arrays
   sort
   one-dimensional optimization
   mass spectrum from file
```

[a] Additional routines are in Appendix 3 (see Table VIII).

```
IF conditions THEN DO
.....
DOEND
```

avoiding the reference to statement numbers. All the editing is done by the editor program (QEDIT). The file is then terminated by including the lines

```
SAVE filename!
EXIT
```

and saved under a "Name". The BASIC program is then prepared by the command

```
:BASIC Name
```

This command calls the BASIC subsystem and uses the file "Name" for input. This has the same effect as typing the file "Name" at a terminal and results in a BASIC program "Filename". The described procedure is specific for the HP 3000, but most systems have similar "input" files.

Following the entry of the routines and functions one writes the main program. This is usually a fairly simple task consisting of dimensioning and initializing variables, calling the routines and functions, and formating the output.

It is also useful to have a program that manipulates tables of numbers (matrices). Frequently needed operations include exchanges of rows and columns (transpose), deletions of rows or columns, normalizations, subtractions of means, etc. A program to perform tasks given in the upper part of Table VIII is in Appendix 3.

Example. For example, a program for obtaining the first-order kinetics parameters given by

$$y = a + a^* \exp(k^* t)$$

where y is concentration, t is time, and a and k are fitted parameters can be quickly put together by using the library routine to open a file, functions FND and FNF to read the file, and the function FNY (Appendix 2) to perform the linear regression. The main program uses the method of Christensen[5] and is in the statements 20–290 (Appendix 4).

INTEGRATED DATA AND WORD PROCESSING

By a combination of techniques described above one can use tables of data for a publication as well as for data evaluation by calculations. Any format of tables may be used, but the programming is usually simpler when the alphabetic characters are on the right-hand side and at the bottom of the table. Table VII contains data on aromatic hydrocarbons (file PAHEVA) in this format.

Let us assume that one wants to do principal components analysis (PCA) of these data. A PCA program available to us requires the input in fields 15 columns wide (standard format for BASIC numeric output), with samples in rows and properties (hydrocarbons) in columns (transpose of Table VII). The program in Appendix 3 will do the transposition and will output data in the 15-columns format (Table VIII). One can record the output in another text file and use it as input for the PCA program after the deletion of the menu and selection portion (from MATICE to AUTOCOR 1). The result of the principal components analysis (Table IX) again can be retained as a text file and inserted in a report.

Table VII. Example of File for Combined Calculations and Report Preparation

```
1404                                  ← rows and columns of numbers
0.2    5.6   1.8   0.7   naphthalene
3.3    3.0   1.3   1.2   !2-methylnaphthalene
2.1    3.4   1.1   0.6   !1-methylnaphthalene
0.4    1.7   0.2   0.5   biphenyl
0.02   1.4   1.3   0.8   !2-ethylnaphthalene
0.2    1.3   7.8   2.9   fluorene
0.2    3.0   3.3   1.5   dibenzothiophene
3.7   12.4  10.6  15.8   phenanthrene
33.6  20.4  21.5  33.6   fluoranthene
19.8  16.5  19.2  19.3   pyrene
 4.9   4.8  22.9   8.7   benzo[a]anthracene
 7.7  11.2   1.6   3.5   chrysene
13.3   5.8   3.2   3.8   benzo[e]pyrene
10.9   8.9   5.6   3.0   benzo[a]pyrene

resid
    commerc
          industr
                highway
```

Hoffmann, E. V., G. L. Mills, J. S. Latimer, and J. G. Quinn, 1984. "Urban runoff as a source of polycyclic aromatic hydrocarbons to coastal waters." *Environ. Sci. Technol.* 18:580–587.

Table VIII. Data from Table VII Transposed by a Program in Appendix 3 for Principal Components Analysis (PCA)

```
MATICE
FILENAME PAHEVA
operations on the current matrix:
    0. end
    1. transpose
    2. normalize rows to 100
    3. normalize columns to 100
    4. subtract mean row
    5. subtract mean column
    6. delete row
    7. delete column
    8. print input (will cancel changes!)
    9. autocorrelation (must be a row matrix)
0E,1T,2NR,3NC,4SR,5SC,6DR,7DC,8PINPUT,9AUTOCOR 1
D0414     ← matrix dimensions          \both required
B         ← BASIC (15 column) format\by the PCA program
    .2          3.3          2.1          .4          .02
    .2          .2           3.7          33.6        19.8
    4.9         7.7          13.3         10.9

    5.6         3            3.4          1.7         1.4
    1.3         3            12.4         20.4        16.5
    4.8         11.2         5.8          8.9

    1.8         1.3          1.1          .2          1.3
    7.8         3.3          10.6         21.5        19.2
    22.9        1.6          3.2          5.6

    .7          1.2          .6           .5          .8
    2.9         1.5          15.8         33.6        19.3
    8.7         3.5          3.8          3

0E,1T,2NR,3NC,4SR,5SC,6DR,7DC,8PINPUT,9AUTOCOR 2
```

Table IX. Result of the Principal Components Analysis (PCA) of Data in Table VIII, Ready To Be Inserted in a Report

```
filename PAHEVA1F
          eigenvalues          % variance
 1          3.886E+02            56
 2          1.804E+02            82

abstract row matrix
 1         -1.222E+01           -3.821E+00
 2         -2.878E+00            1.116E+01
 3          1.520E+01           -1.001E+00
 4         -1.053E-01           -6.340E+00

abstract column matrix
 1          2.245E-02            3.077E-01
 2         -7.544E-02            6.635E-02
 3         -4.834E-02            1.387E-01
 4         -1.748E-02            7.803E-02
 5          3.964E-02            5.087E-02
 6          2.884E-01           -6.901E-02
 7          1.002E-01            1.104E-01
 8          2.022E-01            7.474E-02
 9         -3.755E-01           -7.496E-01
10          1.108E-03           -1.833E-01
11          7.038E-01           -2.396E-01
12         -2.634E-01            3.980E-01
13         -3.369E-01           -7.412E-02
14         -1.904E-01            1.833E-01

mean
2.075     2.2     1.8      .7      .88
3.05      2       10.625   27.275  18.7
10.325    6       6.525    7.1

        reconstructed data matrix Y(1), N(2) 2
        generate AUTOPLOT file? Y(1), N(2) 1
        AUTOPLOT filename PAHEVAP
        x-axis label FACTOR 1
        y-axis label FACTOR 2
        plot label PAHEVA
        perform target analysis? Y(1), N(2) 2
        outline positive subspace? Y(1), N(2) 2
        filename/
```

REPRINT INDEXING

Two files may be used to advantage for indexing reprints. One is a "category" file and contains 49 subjects. The other is a "keyword in context" file. The former can always be kept up to date, the latter is updated yearly. The category file is available only in printed form; the keyword file is printed yearly and stored on a magnetic tape. An "annual increment" file is kept on the system.

The entry program (REPRINT, Appendix 5) is interactive (Table XA) and produces BASIC-formated files. These are then converted (CONVERT, Appendix 5, Table XB) to text files. Finally, the keyword file is produced by a FORTRAN program (NRDR, Appendix 5, Table XC). A similar program in BASIC, easier to implement on other computers, is provided as well. Existing system software (MERGE) is used to merge the keyword files.

The main advantage of the described system is its speed. It enables the user to locate the desired reprint quickly and avoids the need for entering full bibliographic information.

I strongly recommend that each professional maintain personal reprint files. The process of entering the reprints helps to digest what is available and how it is likely to be indexed. One has to be reasonably consistent with the keywords. A keyword file, containing keywords, abbreviations, and their definitions, helps to keep the system consistent (Dr. K. Haya, personal communication).

It is not necessary to follow closely the title of the article. Use anything that may help you to retrieve it, including the author's name or a comment ("FANTASTIC"), but stay away from nonspecific terms such as "determination", "identification", and "effect". Use abbreviations such as PCB (polychlorinated biphenyls), AEC (adenylate energy charge), etc. More information can be stored in this way.

Table X. Reprint File Input and Conversion to Text Format

A. Reprint Entry

```
REPRINT
ADD (1), DISPLAY (2), SORT (3), PRINT SORTED (4) 1
FILE NAME STREDA
STARTING RECORD NUMBER 529
READY TO ACCEPT REPRINTS
TO TERMINATE TYPE E
Do not use commas between keywords!!!!
Adding number 529 ⇐
KEYWORDS, TOTAL MAX. 60 CHARACTERS NAPHTHAL
  ALKYL TOXIC COD
HIT ] IF KEYWORDS OK, ANY KEY IF NOT ]
DO YOU WANT TO SEE LIST OF CATEGORIES -YES (/),NO
  (])/
1-Air 2-Inorg 3-Org 4-Antimony 5-Antioxidants ...
26-Oil 27-OP's 28-OS's 29-PCB 30-Phthalates ...
41-Toxicity 42-Toxicology 43-Treatment 44-Water Polln 45-Bibl...
YOU CAN INPUT UP TO 3 CATEGORIES; TERMINATE BY 0
CATEGORY 26
CATEGORY 41
CATEGORY 0
Adding number 530 ⇐
```

B. Conversion of File to Text (ASCII) Format

```
CONVERT
REPRINT FILE CONVERSION PROGRAM
Explanation? Yes(1), No(2) 2
CONVERSION basic→ASCII {1}, ASCII→basic {2} 1
INPUT FILE STREDA
264100529  NAPHTHAL ALKYL TOXIC COD
243841530  SURFACT TOXIC ZEBRAFSH
294300531  PCB EXPOSURE INCINER OCEAN TERREST
```

C. "Keyword in Context" File Prepared by Program NRDR

ALKYL	NAPHTHAL ALKYL TOXIC COD	529
COD	NAPHTHAL ALKYL TOXIC COD	529
EXPOSURE	PCB EXPOSURE INCINER OCEAN TERREST	531
INCINER	PCB EXPOSURE INCINER OCEAN TERREST	531
NAPHTHAL	NAPHTHAL ALKYL TOXIC COD	529
OCEAN	PCB EXPOSURE INCINER OCEAN TERREST	531
PCB	PCB EXPOSURE INCINER OCEAN TERREST	531
SURFACT	SURFACT TOXIC ZEBRAFSH	530
TERREST	PCB EXPOSURE INCINER OCEAN TERREST	531
TOXIC	NAPHTHAL ALKYL TOXIC COD	529
TOXIC	SURFACT TOXIC ZEBRAFSH	530
ZEBRAFSH	SURFACT TOXIC ZEBRAFSH	530

ACKNOWLEDGMENT

Steve Bellis coauthored the program NRDRSRC. I thank Drs. K. Haya and K. Waiwood for comments and Madelyn M. Irwin for proofreading the manuscript.

Supplementary Material Available: Appendixes 1–5, listings of various programs (40 pages). Ordering information is given on any current masthead.

REFERENCES AND NOTES

(1) QEDIT Text Editor for the HP 3000. Robelle Consulting Ltd., Langley, B.C., Canada V3A 4P9.
(2) Chemical Information Systems, Inc., 7215 York Road, Baltimore, MD 21212.
(3) Heller, S. R. *J. Chem. Inf. Comput. Sci.* **1985**, *25*, 224.
(4) HP 3000 computer system, *BASIC Interpreter Reference Manual*; Hewlett-Packard: Santa Clara, CA, 1978; pp 6.4–6.12.
(5) Christensen, M. K. *Anal. Chem.* **1983**, *55*, 2324.