4
Info File

Once you have this program on tape, you will find it useful in so many ways. There are dozens of uses for it in the home, from keeping a catalogue ofcollections of music and video recordings, or indexing your favourite recipes, to using it as a diary, a telephone book, or a birthday present book. It can be used in the office for holding personnel records or customer records. In the shop, it can be used for holding an up-to-date price list of goods. In the workshop it can catalogue machine parts or component specifications. If your interest lies in research of any kind, use the program for indexing references to the literature of your subject. The Club Secretary could use it for keeping membership details, and a summary of the Agendas of meetings.

How the prograrn works

The information is stored in what is called a file. Here the file is a cassette [ape with data recorded on it. Like the file in the office, the data it holds all relates to the same subject. You can have a file called 'Video', for example, and others called 'Bdays', or 'Address'. Any of these tapes can be taken and used with the INFO FILE program, making it one of the most versatile programs in this book.

Within the file (or tape) the information is separated into records. You may put several hundred records in a file, ifyou like. The records are equivalent to the sheets of paper held in a loose-1eaf binder. Each holds a certain amount of information and each holds information of the same kind, depending on the purpose of the file. For example, in a file called 'Video tapes' there will be a record for each tape in your collection.

The information in each record is divided into a number of fields. This program allows you to have up to four fields in each record. Each

record in a videotape file would hold the same information about each video tape: its title and a summary of the story, or what the tape is about, a cast list or list of performers, a brief note on what vou or your family think of it and perhaps technical details such as its running time, and the number of times it has been run.

The program allows any field to be up to eighty characters long, giving ample room for brief comments such as those in some of the items listed above. However, fields that do not need many characters can be allotted as few as yc;u wish, thus saving memory space and allowing more records to be held in the file.

As another example of fields, consider a file used as a spare parts catalogue. The fields might be:

Part number or code
Specification
Price per unit part
Numbers of parts per box

There is a practical example of the program in use in the next section.

The program provides for the amending of information at any time, so data such as prices or addresses can readily be changed when necessary.

The program includes a ' Find' option which allows you to picli out and view any single record or set of records. You could, fol example, ask it to display the names and addresses of all of your friends and relatives who live in the same town as you do, or all video recordings starring Michael Caine.

Using the program

The first message displayed on running the program is ' DATATAPE READY? (Y/N)' If you have not used the program befort for a given file, key ' N'if; you already have a data tape carrying tht information you want to look at or add to, key ' Y' .

If you key ' N'you are then asked to enter the details of the file tha must first be created . The first item is its name (up to 7 characters), such as ' FRIENDS'' ,RECIPES'' ,SOFTWRE' .Write the name on the label of the cassette, so that you will always remember its exact name. Tht records can have up to four fields, each of which must have a

name too. The fields of a record of a ' FRIENDS' file might be called
' NAME'' ,ADDRESS'' ,TELEPHONE NUMBER' and ' BIRTH-DAY' .
You are also asked to decide how many characters are to be allocated
to each field. Each field can have between 1 and 80 characters. The
more characters you allot to each field, the fewer records the file can
hold. Make sure that you allow enough characters for names and
addresses, for example, as it is not possible to alter the number of
characters later. Remember that the characters include spaces between
words. When you have finished giving your requirements, the
computer calculates how many records the file will be able to hold,
and displays this figure. If this is enough, key ' Y'. If not, key ' N' and
you are asked to repeat the entering of field names and numbers
of characters, perhaps cutting down on the number of characters to
allow an increased number of records.

When all is settled, and you have keyed ' Y' you will see the
message ' RECORD then RETURN' Put an empty C-10 or C-15 tape
into the tape recorder. This must be re-wound to the beginning, but not
to the non-magnetic leader. Press RECORD and PLAY on the tape
recorder, then press the RETURN key of the Electron. After three to
four minutes the new file is created on the tape. When the recorder
stops, the message ' BACK-UP?' appears. It is a very good idea to have
a backup (see general remarks on data recording in the introduction,
Key Points), so key ' Y' and repeat the recording operation with another
new tape. After this, press ' N' which brings the program to an end.
You now have a data tape (or two). RUN the program again and this
time answer ' Y' to the first question.

If you already had a data tape, or have just started one, you will
have keyed ' Y' to the question, ' DATA TAPE READY?' You are then
asked to type in the file name; copy this exactly from the name written
on the cassette. Make sure the tape is re-wound, and then press PLAY.
When the recorder stops, the menu is displayed on the screen.

The Menu offers three options:

(1) Enter record: this is for adding information to your file. When you
choose this option the screen displays the name of the first field and,
in brackets after the name, the maximum number of characters
allowed. Type in whatever information you want to go into this field.
Typing in is slightly different from normal typing on the computer. For
one thing, it uses both lower-case and upper-case letters, as does a
typewriter. You press the SHIFT key to get upper-case (capital) letters,
just as on a typewriter. There is no cursor but, as you type, your entry

appears on the next line of the screen. When you have typed to within five characters of the allowed number, a gentle ' beep'from the loudspeaker reminds you that you are running out of space in that field. When you have typed as many characters ts are allowed, the computer refuses to accept any more. You will iotice that the routine refuses to accept double quotes (""). Fhis is because the string-handling routines of the computer would take the double quotes to be the end of the text in the field.

When you have finished entering the information in that field, press RETURN. The computer will now display the name of the second field. Carry on in this way until all fields have been filled in. The computer then returns you to the menu.

If you want to leave a field empty for the moment, just press RETURN without typing anything. If you make a mistake when typing information, press the DELETE key. This does not delete characters as it normally does. Instead it prints a vertical line at the end of your incorrect entry, to show that it has been discarded. You can now retype the correct version, which appears on the line below. You can use the delete function as often as you want until the entry is correct.

(2) Find record: this is the option by which you can examine (and alter, if you wish to) any record. When you select this option you are first asked which fields are to be searched to find the record you want. If you are not sure, or want all fields to be searched, key ' Y'in response to all queries. Otherwise key ' N'to those fields which are not relevant, for it takes longer to search all fields rather thanjust one or two. ihis option is a very poWerful one, for it can find a piece of information is such a variety of ways. When you have decided which fields are to be included in the search, you are asked to type in ' WHAT IS TO BE FOUND' .The computer accepts up to ten characters, and the typing in is just the same as making an entry in Option 1 .

What you type depends on how you want the search to be made. Here are some examples to give you an idea of what to type:

(a) If you want to look through every record on the file, type nothing - just press RETURN.
(b) If you know exactly which record you want to see, type in the details fairly fully. For example, if you have selected the ' NAME'field for the search and want to see the record ofJoseph Bloggs, type ' JosephBloggs' and (unless you have more than one person of that

name of your file) the computer will find and display Joseph Bloggs'
record and no other.

(c) If you are not sure how Joe spells his name (is it Bloggs or Blogs
or Blogg? Is it Joseph or Joe?) try typing ' Blog'The computer will
find all records for which one ofthe names begins with ' Blog'so you
are bound to find the record you want.

(d) If you want to find all names beginning with a given letter, type ' B'
for example, and you will then be shown all records with names
beginning with that letter, from Babb to Bywaters.

(e) The program can just as easily find groups of characters which
occur in the middle of words or phrases. For example, if you search an
' ADDRESS'field for ' W1X4BD' it will pick out all records which
have that post code in the address. Similarly, if an employee' scode
includes the letter ' M'for all male employees, searching the
corresponding field for ' M'results in all records of male employees
being displayed.

If there is no record which has the features you are searching for,
the message ' NOTFOUND' appears and, after a few seconds, you are
returned to the menu.

As each record is displayed, you are given three options:

C = CONTINUE
A = AMEND
SPACE BAR = MENU

If you key ' C'the search continues to display the next record which
conforms to your request. When no more such records are to be found
in the file, the message ' SEARCHFINISH ED' is displayed and a few
seconds later you are returned to the menu.

Keying ' A'gives you the chance to alter or delete the whole or part
of a record. You are shown each field of the record in turn. You can
then type one of the following:

(a) RETURN: this deletes the field completely, leaving it empty.

(b) ' /' : the field is retained, unaltered. Then press RETURN.

(c) Type in the new information which is to replace the old. Note
that the new information *replaces* the old information - it is not added
to it. Then press RETURN.

If you delete all the fields in a record, the empty record is removed
from the file, making room for a new record. All records following the
deleted one are renumbered.

After you have amended a record, the search continues to find any

further records conforming to your original description. This allows you to call up a set of records or even all of the records and amend each one in turn.

(3) Finish: It is essential not to finish the program by switching off the computer, or by pressing the ESCAPE or BREAK keys. Instead, choose Option 3. Rewind the data tape when the message ' RECORD then RETURN' appears. Make sure it is positioned with the magnetic part against the tape head. Press RECORD and PLAY on the recorder, then press the RETURN key on the Electron. When the data has been saved, you will then be asked if you want a backup. If so, repeat the operation, using your backup data tape. The program then finishes.

INFO in action

Here is how INFO can be used to hold the details of a personal library of books. The SHIFT and CAPS LK keys were pressed before the program was run, so that we could use capitals and lower-case letters. But you can use all capitals if you prefer.

Figure 4. 1 shows a printout of the first four records. As you can see the fields are:

TITLE of book
Name of AUTHOR
Name of PUBLISHER
DATE of publication

```
RECORD NO. 1
TITLE
Origin of species
AUTHOR
Charles Darwin
PUBLISHER
John Murray
YEAR
1859
RECORD NO. 2
TITLE
Life on Earth
AUTHOR
David Attenborough
PUBLISHER
```

```
           Collins
           YEAR
           1979
           RECORD NO. 3
           TITLE
           Animals with backbones
           AUTHOR
           Ralph Buchsbaum
           PUBLISHER
           Univ. Chicago P
           YEAR
           1940
           RECORD NO. 4
           TITLE
           West African trees
           AUTHOR
           Gledhill
           PUBLISHER
           Longman
           YEAR
           1972
```

Fig. 4.1.

The records were set up with 30 characters for TITLE, 20 for AUTHOR and also for PUBLISHER, and 4 for DATE (only the year is normally required). This allowed more than 100 books to be listed.

Figure 4.2 shows the printout obtained when we asked INFO FILE to search for all titles relating to Africa. We keyed (2) to get to

```
           RECORD NO.4
           TITLE
           West African trees
           AUTHOR
           Gledhill
           PUBLISHER
           Longman
           YEAR
           1972
           RECORD NO. 7
           TITLE
           West African snakes
           AUTHOR
           Gledhill
```

```
          PUBLISHER
          Longman
          YEAR
          1972
          RECORD NO. 10
          TITLE
          Butterflies of Africa
          AUTHOR
          John Williams
          PUBLISHER
          Collins
          YEAR
          1969
          RECORD NO. 12
          TITLE
          Birds of West Africa
          AUTHOR
          Serle, Morel, Hartig
          PUBLISHER
          Collins
          YEAR
          1977
```
Fig. 4.2.

' FINDRECORD' .Then we keyed ' Y'in response to ' FIELD1
(TITLE)' and ' N'in response to the other fields. Finally we keyed
*Africa' when asked what was to be found. The books shown in Fig.
4.2 were listed on the screen, one at a time. Note that some titles have
' Africaln them while others have ' African'The search routine finds
both sets of titles. If we had been interested only in West Africa, we
could have keyed ' WestAfric' (only ten characters allowed!). This
would have given us records no 4, 7 and !2, but not record 10.

```
          RECORD NO. 12
          TITLE
          Birds of West Africa
          AUTHOR
          Serle, Morel, Hartig
          PUBLISHER
          Collins
          YEAR
          1977
          RECORD NO. 16
          TITLE
```

```
Birds of New Guinea
AUTHOR
John Gould
PUBLISHER
Methuen
YEAR
1970
```
               Fig. 4.3.

```
RECORD NO. 6
TITLE
Invest. in woodland ecology
AUTHOR
Prime
PUBLISHER
Heinemann
YEAR
1970
RECORD NO. 13
TITLE
The young botanist
AUTHOR
Prime
PUBLISHER
Nelson
YEAR
1963
```
   Fig. 4.4. Result of a search for books by a selected author

   Figure 4.3 shows the results of a search for ' Birds'. This gives us the West African book we obtained in Fig. 4.2, together with a book on the birds of New Guinea. Note that if some of the titles had included the word ' birds' (no capital B), or ' bird' (singular) they would not have been picked out in the search. Provided you do not have titles such as ' The third dimension' in your list, it would be better to key ' ird' when searching for bird books.

   INFO FILE can also search the records looking for books from a given publisher. In Fig. 4.5 we see all the books published by Collins.

```
RECORD NO. 2
TITLE
Life on Earth
AUTHOR
David Attenborough
```

```
PUBLISHER
Collins
YEAR
1979
RECORD NO. 10
TITLE
Butterfiles of Africa
AUTHOR
John Williams
PUBLISHER
Collins
YEAR
1969
RECORD NO. 12
TITLE
Birds of West Africa
AUTHOR
Serle, Morel, Hartig
PUBLISHER
Collins
YEAR
1977
RECORD NO. 15
TITLE
Guide to the sea shore
AUTHOR
Barrett and Yonge
PUBLISHER
Collins
YEAR
1958
```

Fig. 4.5.

Finally, in Fig. 4.6, we see the result of a search for all the books published in the nineteenth century. In this example, we keyed ' Y' in response to ' FIELD 4 (YEAR)', and then typed in ' 18' as the search field. The search would have picked out books published in 1918, too, but there were no books of that vintage in the list.

```
RECORD NO. 1
TITLE
Origin of Species
AUTHOR
Charles Darwin
```

```
PUBLISHER
John Murray
YEAR
1859
RECORD NO. 9
TITLE
Natural History of Selbourne
AUTHOR
Gilbert White
PUBLISHER
Routledge
YEAR
1853
RECORD NO. 14
TITLE
The vegetable kingdom
AUTHOR
William Rhind
PUBLISHER
Blackie
YEAR
1868
```

Fig. 4.6.

Keying in

There are no special problems in keying in this program. Take care with the ' apostrophes'or ' ticks'in lines 460—500, as well as elsewhere. If you type the wrong number of these, the display will be spoilt.

Program design

20 Setting Mode.
30 Dimensioning arrays.
40 Filling E$ and S$.
50-330   Routine for setting up a new data tape (also used from line 230 for saving data).
340-450 Reading data from tape.
460-540 The menu.
550-640 Entering records.
650-920 Searching records.
930-1090 Amending records.

100-110   Finishing (completed in lines 190-280).
1120-1140   PROCyn to get key-press.
1150-1260   PROCinput used when typing into fields.

Points of interest

This program has a special input procedure (PROCinput). As you type in an entry, this procedure builds up the entry, character by character, into a string variable, called E$. E$ can be up to 80 characters long. The string storing routines of the Electron puts a string into a new part of memory whenever a string is increased in length. If a string is increased in length several times, a lot of memory can be required, most of which is being wasted since it holds only the earlier, shorter versions of the string.

This action would be very serious in this program, for E$ could be increased in length from] byte in steps of I byte until it was 80 bytes long (the maximum size for a field entry). This could take up to $1 + 2 + 3 + 4 + . . . + 77 + 78 + 79 + 80$ bytes - a total of 3240 bytes! Of these, only 80 bytes would hold the latest E$ and the other 3160, almost 3 kilobytes of memory, would be wasted. In a program which also has to store as much data as possible, we cannot afford to waste so much memory.

The way to avoid using up too much memory is to give E$ its maximum length at the beginning of the program. This is what is done in line 40. E$ is made a string ofeighty ' E' The Electron stores this full-1ength E$ away, and it occupies 80 bytes. Whenever E$ is made shorter, as at line 1180, this has no effect on the amount of memory that has already been allocated for E$. As PROCinput accepts more input, E$ grows and grows in length, but there is always sufficient room in those 80 bytes already set aside. No additional memory is taken for E$.

For the same reason, S$ is filled with ' E' (line 40) and the main data array is created (lines 210-220) with all its fields in every record filled with ' D' If an entry is deleted, the space at the ' bottom of the list is refilled with ' D' (line 1060). This does not prevent the program from using extra memory if several long fields are amended and made longer. In working out how many records may be allowed, we have left some memory to spare for amending fields. The more memory set aside for this, the fewer records the program can be allowed to accept. If you are intending to lengthen very many of your records, it is wise

to amend only a few at a time, saving to tape after amending each small batch. Each time the program is rerun to amend the next batch of records, memory is re-allocated without waste.

The program

```
   10 REM ** INFO FILE **
   20 MODE 4
   30 DIM N$(4),NC(4),FS(4),F$(4):LE=0:N
C(4)=10
   40 E$=STRING$(80,"E"):S$=E$
   50 PRINT''"DATA TAPE READY? (Y/N)"
   60 PROCyn
   70 IF INSTR("Yy",KEY$) THEN 340
   80 NN=0:TC=0:CLS:INPUT''"FILE NAME "N
$
   90 N$=LEFT$(N$,7)
  100 INPUT TAB(0,4)"NO. OF FIELDS (1-4)
 "NA$
  110 N=VAL(NA$):IF N<1 OR N>4 THEN 80
  120 FOR J=0 TO N-1
  130 PRINT TAB(0,6+4*J)"NAME OF FIELD "
;J+1;:INPUT" "N$(J)
  140 INPUT TAB(0,8+4*J)"NO. OF CHARACTE
RS (MAX 80) "NC$
  150 NC(J)=VAL(NC$):IF NC(J)<1 OR NC(J)
>80 THEN 140
  160 TC=TC+NC(J):NEXT
  170 NR=INT(7000/TC)
  180 PRINT''"YOU CAN HAVE UP TO ";NR;" R
ECORDS.":PRINT''"OK? (Y/N)"
  190 PROCyn
  200 IF KEY$="N" THEN 80
  210 DIM D$(NR,N-1)
  220 FOR J=1 TO NR:FOR K=0 TO N-1:D$(J,
K)=STRING$(NC(K),"D"):NEXT:NEXT
  230 X=OPENOUT N$
  240 PRINT#X,N,NR,NN
  250 FOR J=0 TO N-1
  260 PRINT#X,N$(J),NC(J)
  270 NEXT
  280 FOR J=1 TO NR:FOR K=0 TO N-1:PRINT
#X,D$(J,K):NEXT:NEXT
```

```
290 CLOSE#X
300 PRINT'"BACK-UP? (Y/N) "
310 PROCyn
320 IF KEY$="Y" OR KEY$="y" THEN 230
330 CLS:PRINT'''"INFOFILE FINISHED":END
340 INPUT''"FILE NAME "N$
350 X=OPENIN N$
360 INPUT#X,N,NR,NN
370 FOR J=0 TO N-1
380 INPUT#X,N$(J),NC(J)
390 NEXT
400 DIM D$(NR,N-1)
410 FOR J=1 TO NR
420 FOR K=0 TO N-1
430 INPUT#X,D$(J,K)
440 NEXT:NEXT
450 CLOSE#X
460 CLS:PRINT'''"FILE:   "N$
470 PRINT''TAB(5)"1)  ENTER RECORD"
480 PRINT''TAB(5)"2)  FIND RECORD"
490 PRINT''TAB(5)"3)  FINISH"
500 PRINT'''"<  SELECT BY PRESSING NUM
BERED KEY  >"
510 KEY$=GET$
520 A=VAL(KEY$):IF A<1 OR A>3 THEN 510
530 IF A=2 AND NN=0 THEN 510
540 ON A GOTO 550,650,1100
550 NN=NN+1:CLS:PRINT'"FILE:   "N$,TAB
(25)"RECORD NO. ";NN
560 FOR J=0 TO N-1
570 PRINT'"FIELD ";J+1;": "N$(J)" (";N
C(J)")"
580 PROCinput
590 D$(NN,J)=E$
600 PRINT"":NEXT
610 IF NN=NR THEN PRINT''"**FILE FULL"
620 PRINT'''"<ANY KEY FOR MORE : SPACE
BAR FOR MENU>"
630 KEY$=GET$:IF KEY$<>" " THEN 550
640 GOTO 460
650 CLS:PRINT'"FILE: "N$
660 FOR J=0 TO 3:FS(J)=0:NEXT
670 FOR J=0 TO N-1
680 PRINT'''"SEARCH FIELD ";J+1;" ("N$(
```

```
J)") Y/N";
  690 KEY$=GET$:PRINT" ";KEY$
  700 IF INSTR("Yy",KEY$) THEN FS(J)=1
  710 NEXT
  720 PRINT''"TYPE WHAT IS TO BE FOUND (
MAX 10 CHARS)"
  730 FF=0:J=4:PROCinput:S$=E$
  740 L=1
  750 J=0
  760 IF FS(J)=0 THEN 790
  770 X=INSTR(D$(L,J),S$)
  780 IF X>0 THEN 840
  790 J=J+1:IF J<N THEN 760
  800 L=L+1:IF L<NN+1 THEN 750
  810 IF FF=0 THEN PRINT''"NOT FOUND":GO
TO 830
  820 PRINT''"SEARCH COMPLETED"
  830 TIME=0:REPEAT UNTIL TIME>500:GOTO
460
  840 FF=1:CLS:PRINT''"RECORD NO. ";L:FO
R M=0 TO N-1
  850 PRINT';N$(M)
  860 PRINT'D$(L,M)
  870 NEXT
  880 PRINT''"C=CONTINUE : A=AMEND : SPA
CE BAR=MENU"
  890 KEY$=GET$
  900 IF KEY$=" " THEN 460
  910 IF KEY$="C" OR KEY$="c" THEN 800
  920 IF KEY$<>"A" AND KEY$<>"a" THEN 89
0
  930 FOR J=0 TO N-1
  940 CLS:PRINT''"AMEND RECORD ";L
  950 PRINT''"FIELD ";J+1;" IS:":PRINT D
$(L,J)
  960 PRINT'"KEY / OR THE NEW VERSION: "
;
  970 PROCinput:IF E$<>"/" THEN D$(L,J)=
E$
  980 NEXT
  990 FD=0:FOR P=0 TO N-1
 1000 IF D$(L,P)<>"" THEN FD=1
 1010 NEXT:PRINT''"RECORD AMENDED"
 1020 IF FD=1 THEN 800
```

```
 1030 IF L=NN THEN 1060
 1040 FOR P=L TO NN-1
 1050 FOR Q=0 TO N-1:D$(P,Q)=D$(P+1,Q):N
EXT:NEXT
 1060 FOR Q=0 TO N-1:D$(NN,Q)=STRING$(NC
(Q),"D")
 1070 NN=NN-1
 1080 IF L>NN THEN 820
 1090 GOTO 750
 1100 CLS:PRINT''"FINISHING"
 1110 GOTO 230
 1120 DEF PROCyn
 1130 REPEAT:KEY$=GET$:UNTIL INSTR("YyNn
",KEY$)
 1140 ENDPROC
 1150 DEF PROCinput
 1160 E$=""
 1170 A$=GET$:IF A$="" THEN 1170
 1180 IF A$=CHR$(34) THEN 1170
 1190 IF A$=CHR$(13) THEN ENDPROC
 1200 IF A$=CHR$(127) THEN E$="":PRINT"|
":GOTO 1160
 1210 PRINT A$;
 1220 E$=E$+A$
 1230 IF LEN(E$)=NC(J) THEN ENDPROC
 1240 IF LEN(E$)>NC(J)-5 THEN PRINT CHR$
(7);
 1250 GOTO 1170
 1260 ENDPROC
```

Variations

If you have a printer attached to your Electron, you may like to be able
to print out lists of the information in INFO FILE, or the results of
searches. Look in the handbook of your printer, or printer interface to
find out what command is required to enable output to the printer.
Insert this in the INFO FILE program at line 840. after the ' CLS' .
Make sure that this command has a colon (:) before and after it. To
disable the printer later in the program, type an extra line, 875, holding
the disabling command (see your handbook). The program should then
produce printouts similar to those shown in Figs. 4.1 to 4.6.

   If you are using tape and would like to use 10-character file names,
alter the ' 7' in line 90 to ' 10' .