

# The SORCERER'S APPRENTICE

August, 1979

Vol. 1, No. 3

## MISCELLANECUS NOTES

Please note the new format (again). This was done to ease the cost of producing this newsletter, not to cause any eye strain. Excuse the small print.

A few of you who responded to my letter in Kilobaud Micro-computing have asked about ON LINE. ON LINE is a "buy and sell forum for the computer hobbyist", published 17.3925 times a year by Dave Beetle. ON LINE is a small magazine chock full of advertisements submitted by computer enthusiasts. It is available free of charge in some microcomputer stores, or you can subscribe directly: \$1.00 for 4 issues, \$3.75 for 18, or \$7.00 for 36. That's a hard price to beat, and so are most of the prices inside ON LINE as well. Worth looking into if you need any equipment.

Going to Australia? Perhaps you should drop the Sorcerer Users Group there a line (P.O. Box 144, Doncaster, Vic., 3108). They sent me a copy of their newsletter, some of which is reproduced in this issue (see the articles on program merging and recovery). I have entered into an agreement with them which allows the unlimited reprinting of material from the other groups newsletter (ala Creative Computing). Note: if for any reason you submit something to me and do not want it to appear in their newsletter, tell me so upon submission. I'll make sure that they do not receive it. By the way, you can subscribe directly to their newsletter by sending \$10.00 to the given address.

Back issues of The Sorcerer's Apprentice are available, but due to the unanticipated demand for them, there will now be a \$.75 fee to cover the costs of copying and mailing. No SASE is needed, but please send a check or other form of paper money (no stamps) and let me know what issues you want.

If you live outside the U.S.A., self-addressed, stamped envelopes are probably a little hard to come by (especially the American stamps). There is a simple solution, however. Instead of sending an SASE near publication time, send \$1.50 (US). This will get the Apprentice to almost anyplace in the world, hot off the press, of course.

Ralph Ruh (623 Medill Ave., Lancaster, OH, 43130) has a log-keeping program for HAM's. He says it runs well in a 16K system. Contact Ralph for more details.

The Sorcerer's Apprentice is published approximately 8 times a year by The Sorcerer User's Group, Dave Bristol, 1530 Wabtenaw, Ann Arbor, MI, 48104. Send a self-addressed, stamped envelope for membership and subscription details.

## QUESTIONS AND ANSWERS

I am happy to personally answer any questions you may have, although you should keep in mind that I am not an expert in any fashion. If you would like a direct answer, please send an extra self-addressed, stamped envelope. Of course, if enough people ask the same question, it will be answered in the Apprentice, and I'll use your SASE for another issue.

## YOU CAN'T ALWAYS GET WHAT YOU WANT

Especially if you don't tell me what you want to see! I am prepared to write about and print anything I can get my hands on. If you don't think you can write well enough to have something printed, send me your basic idea and I will spruce it up. You will, of course, still receive credit as author.

Can't think what to write about? Here are some ideas: book reviews of the Technical Manual, the word processing and development ROM PAC's, the Creative Computing programs, interfacing through either one of the I/O ports or through the expansion connector, review of the S-100 unit, or anything you have hooked onto your Sorcerer. The Sorcerer is still a new computer, and almost everything needs to be said about it.

## PROGRAM EXCHANGE

After the release of the next issue, you should be able to get the programs listed in these pages on a cassette tape. Thanks to all of you who volunteered to serve as exchanger. You will be hearing from me shortly if you have not already.

## SASE INFO

Self-addressed, stamped envelopes have so far proven to be a good way of keeping the Apprentice's costs down. However, a few guidelines about the use of SASE's are in order. Do not use those nifty little return address labels for the mailing address; they seem to have a slight tendency to fall off when placed in the middle of an envelope. Do include some sort of return address on your envelope to assure that your Apprentice eventually gets somewhere. And lastly, to avoid issue mix-ups, mark your envelope with the number of the issue you want.

## PUBLICATION SCHEDULE

These are only approximate dates, but I will try to stick to them as much as possible. I am printing them for you to know when to send an SASE: #4, September 30; #5, November 17; #6, December 30. After December we'll just have to see what happens. The December issue will contain information on the future of The Sorcerer's Apprentice.

## PROGRAM MERGING

contributed by I. McMillan and D. Trussel  
Sorcerer Users Group newsletter, vol. 1, no. 2  
Australia

"The procedure described here has been empirically derived. When it is followed precisely I have always found it to be successful. The steps are time consuming. Unless you wish to add a program of some size (at least 10 lines) it is probably not practical to use.

"The merge operation involves 3 main steps. Firstly, monitor LOAD is used to stack a new program right behind one already existing in memory. Secondly, contents of the memory location holding the program length are increase to allow for the expanded size of the combined programs. Thirdly, a fictitious line is added to force the interpreter to rectify the line header of the appended program lines.

"The example illustrates how two programs are combined. In detail these steps are:

1. If the first program is not already in the computer, CLOAD it from a cassette.
2. Leave BASIC and enter monitor mode.
3. DUMP memory in the vicinity of the last program line. The end of the last program line will be denoted by a set of 3 or 4 00's.
4. LOAD the program to be added using the address of the second 00 (023B in the example) for the starting address. This command will be of the form LO NAME 1 abcd where abcd is the address of the second 00. The tape unit number (1 in the example) must be used (see Guided Tour Manual page 50).
5. Memory is DUMPed in the vicinity of the end of the second program. The format is identical to that described in step 3 above. The location can be roughly determined by adding the BLCK length of the second program to its start address. (Note: addresses and BLCK are hexadecimal numbers.)
6. The address of the fourth 00 is noted (02A5 in the example). This address is ENTERed into LB7 and LB8 in the following manner -
  - (a) enter: EN, space, LB7, return
  - (b) enter: second pair of address digits (A5 in the example), space, the first pair of address digits (02, the back slash, and then return.
7. Return to BASIC.
8. Before you do anything else, enter a fictitious line number (i.e. one not used in either program) and statement. 0 REM has been used in the example. This line may now be removed if you wish.

"The two programs are now combined into a single program. The procedure will work correctly even if there are identical line numbers in the two programs. However, it is desirable to have line numbers of the original program all lower than those of the second program. The renumbering program in next month's issue of the newsletter will facilitate the combining of BASIC programs.

"There is one problem which can occur when the above procedure

is used. Sometimes there are only 3 00's present at the end of the combined program (see step 6). In this case it is necessary to ENTER a 00 in the location following the third 00 (and then proceed to step 7)."

## PROGRAM RECOVERY

contributed by the Sorcerer Users Group  
newsletter, Australia

"Programs can be recovered after a NEW command has been executed. All that the new command does is place 00 in locations LD5 and LD6. These locations normally hold the memory location of the beginning of the second line of the program. Program lines in BASIC end with 00. Therefore, the location after the first 00 after LD8 (LD7 and LD8 hold the line number of the first line) is the number needed for the contents of LD5 and LD6. To look at memory, use the DUMP command. The last two digits are ENTERed in LD5 and the first two digits are ENTERed in LD6. The contents of memory locations LB7 and LB8 must be corrected in a similar fashion. These locations must contain the low and high order bytes respectively of the end of the last line in BASIC. After that is carried out return to BASIC (PP). Enter a new line number (eg 0 REM). The program should now be recovered."

Thanks for the tips, folks. We'll be looking forward to that renumbering program.

## MORE ONE-STROKE STATEMENTS

Thanks to all who sent in this information. As you may or may not know, the Sorcerer's BASIC stores statements such as PRINT and STOP not as words themselves but as a single character. The interpreter translates from character to word for listings. The character is usually one of the graphics, as shown in appendix F in "A Short Tour of BASIC" that is provided with your Sorcerer. There are some further statements, however, that were omitted from this section in the book. These are not just "graphic" statements, but "shift-graphic" statements. In other words, you need to depress the SHIFT and GRAPHIC keys along with the proper key on the keyboard to get the desired statement. The table shows how to get what:

1 - STR\$	
2 - VAL	
3 - ASC	Remember to depress the SHIFT
4 - CHR\$	and GRAPHIC keys at the same time.
5 - LEFT\$	
6 - RIGHT\$	
7 - MID\$	

None of the other keys on the Sorcerer seems to produce anything relevant when used in this manner. They do produce some interesting garbage, though.

## THE COST OF APPRENTICING

With over one hundred and twenty of you now receiving this newsletter, printing costs have risen excruciatingly high. One change I have tried newly with this issue is the reduced format size. This should help somewhat. This newsletter is, however, a self supported project. It was not my intention to go for broke in writing it. I would like to help other Sorcerer owners get the most out of their machines (and likewise help myself). I appreciate very much the donations some of you have sent in, and would like to request of the rest of you reading this issue that you help out too. Any little bit you can give will help assure that there will be an issue following this one.

Along the money lines, a nearby store has informed me that they plan to start their own Sorcerer users group this September. One of the employees there has made mention of the fact that they may want to purchase my mailing list of Sorcerer owners. I know that it is standard practice in the world of magazine publishing to sell mailing lists (how do you think you get all that junk mail, anyway?), but I wanted an OK from you first. On your next SASE simply mark N if you do not want your name to be sold, otherwise I'll assume you have no objections.

One last note. If you do send a contribution, please keep sending SASE's. Once I'm in school again this fall, my time will be more valuable than anything else, and it just won't be possible to address and put stamps on 100 or so envelopes. If you prefer, send a bunch of envelopes at once. I'll make sure that your issue is sent as soon as it is printed.

## PROGRAM LEGALITY

The Sorcerer's Apprentice is in need of someone familiar with copyright laws. Several would-be-program-submitters have asked me about programs. Most claim that they have programs which are not exactly original- they have modified someone else's for their Sorcerer. Some have taken old programs from books, some from other programs on other machines. The main question is this: Can The Sorcerer's Apprentice print programs modified for the Sorcerer that have previously appeared in other copyrighted publications? The Mastermind program in this issue is an example of what I mean. I know the program is original; I helped the author (my sister) write it. I know that there are other Mastermind's running on other systems, although they aren't always called Mastermind. Can anyone out there provide an answer?

## G2 ROULETTE CORRECTIONS

Donald Despres of Columbia, Maryland, sent the following which he received from Joe Hanks at GRT in California.

"The problem is due to the functioning of random numbers. Statement number 2840 is:

```
2840 NN=INT(RND(1)*38):NT=NN+152
```

"The statement RND(1) returns a random number between 0 and 1. The integer value of this number multiplied by 38 is used as a subscript for the array named W\$(1) to W\$(38). Due to the

function of RND, it is possible that the subscript value will be zero. This happens when the number 28 comes up on the "wheel". This is what causes the Illegal Function Call in statement 3010. To correct the problem, change the statement 2840 to be as follows:

```
2840 NN=INT(RND(1)*38+1):NT=NN+152"
```

Thanks to Donald and to Joe for this fix. (PS Anybody else have this program package? How about reviewing it- is it a good buy?)

## DATABASE MANAGEMENT CONTRIBUTED BY YOURS TRULY

This program is a follow-up to the string saving short program that appeared in the last issue. It allows you to create records in a file with variable length fields and later save them on cassette. Many databases can be used with the same program in this case. One such use is an address list with all the names, phone numbers, and addresses of people you communicate with. I am using a modified version of the program to keep all of your names where I can easily find them. The program is fairly straightforward, but I have discovered a few changes that should be made. The main one has to do with the limit on the length of a line. This is set in the monitor's firmware to somewhere around 64 characters. Thus if you try to retrieve a 72 character record from tape, the program will read it as two separate records. The easy way to fix this is to limit the length of the record to 64 characters when INPUTTING it from the keyboard. Do this by changing the 255's to 64's in lines 400 and 410. The other method I know of, which doesn't place a limit on the record lengths, is to add these lines:

```
923 : IF RIGHT$(REC$(I),1)="/" THEN 930  
925 : INPUT R$:REC$(I)=REC$(I)+RIGHT$(R$,LEN(R$-1)):GOTO 920  
928 : R$=""
```

The other problem has to do with CLEARING the proper amount of memory. If there is not enough string space, the program may bomb out after you have entered many records, and all will have gone to waste.

## MASTERMIND CONTRIBUTED BY EMILY BRISTOR TROY, MI

"This game is one of many possible variations of the game of Mastermind. I chose to use numbers instead of colors because they are easier to work with and easier to remember when playing the game. I also made it possible for the player to choose between two levels of difficulty for beginners at the game and for the more experienced player. The program will not allow the computer to choose the same number twice in any combination it "thinks" up, therefore the player is not allowed to do so either and is reminded if he does so. The graphics I used did not come out as such in the listing. In lines 410, 460, and 500 you should use graphic-x for the exclamation points. In 410 and 460 use graphic-c for the dashes, and in line 500 use graphic-v for the dashes."

Part 3 of a Series of 5

If the value of the expression is negative, you will get an FC error message. If the value is not a whole number, Sorcerer will ignore the fractional part.

You can use CLEAR <expression> as a program statement to reserve string space during the execution of a program.

Example:

```
10 CLEAR 999
20 INPUT "Tell me your life story"; X$
```

However, be careful doing this--CLEAR <expression> sets all your numerical variables to zero.

```
10 REM MASTERMIND BY EMILY BRISTOR, JULY-AUGUST, 1979
20 PRINTCHR$(12)
30 PRINT TAB(16);"WELCOME TO MASTER-MIND!":PRINT:PRINT
40 INPUT "DO YOU WANT INSTRUCTIONS ";K$
50 IF ASC(K$) <> 78 AND ASC(K$) <> 110 THEN GOSUB 1020
60 DIM M(4,9)
70 DIM P(1,9)
80 DIM Q(1,9)
90 PRINT:INPUT "CHOOSE BEGINNER (USES 1-6) OR ADVANCED (USES 1-9)
) GAME ";L$
100 IF ASC(L$)=66 OR ASC(L$)=98 THEN R=6
110 IF ASC(L$)=65 OR ASC(L$)=97 THEN R=9
120 REM
130 PRINT:PRINT:PRINT
140 XX=RND(-RND(1))
150 W=INT(RND(1)*R+1)
160 X=INT(RND(1)*R+1)
170 IF X=W GOTO 160
180 Y=INT(RND(1)*R+1)
190 IF Y=X OR Y=W GOTO 180
200 Z=INT(RND(1)*R+1)
210 IF Z=W OR Z=X OR Z=Y GOTO 200
220 PRINTCHR$(12)
230 FOR I=1 TO 9
240 PRINT:PRINT "GUESS ";I;
250 IF I=9 THEN PRINT "(THIS IS YOUR LAST SHOT)";
260 M(1,I)=0:M(2,I)=0:M(3,I)=0:M(4,I)=0
270 INPUT M(1,I),M(2,I),M(3,I),M(4,I)
280 F=0
290 FOR V=1 TO 3
300 FOR J=(V+1) TO 4
310 IF M(V,I)=M(J,I) THEN GOSUB 920
320 IF F=1 THEN J=4 AND V=3:GOTO 380
330 NEXT J
340 NEXT V
350 IF F=1 GOTO 520
360 P(1,I)=0:Q(1,I)=0
370 GOSUB 530
380 REM
390 N$="I":IF F=1 GOTO 410
400 PRINTCHR$(12):PRINT
410 PRINT"!---!---!---!---!---!---!---!---!"
420 PRINTN$;" ";N$:N$;" A ";N$;" B ";N$;" C ";N$;" D ";
430 PRINTN$;N$;" N ";N$;" P ";N$
440 FOR E=1 TO I
450 IF F=1 THEN P(1,I)=0:Q(1,I)=0
460 PRINT"!---!---!---!---!---!---!---!---!"
470 PRINTN$;E;N$;N$:M(1,E);N$;M(2,E);N$;M(3,E);N$;M(4,E);
480 PRINTN$;N$;P(1,E);N$;Q(1,E);N$
490 NEXT E
500 PRINT"!---!---!---!---!---!---!---!---!"
510 IF I=9 GOTO 630
520 NEXT I
530 FOR B=1 TO 4
540 IF M(B,I)=WORM(B,I)=XORM(B,I)=YORM(B,I)=ZTHENP(1,I)=P(1,I)+1
550 NEXT B
560 IF M(1,I)=W THEN Q(1,I)=Q(1,I)+1
570 IF M(2,I)=X THEN Q(1,I)=Q(1,I)+1
580 IF M(3,I)=Y THEN Q(1,I)=Q(1,I)+1
590 IF M(4,I)=Z THEN Q(1,I)=Q(1,I)+1
```

```

620 IF P(1,1)=4 AND Q(1,1)=4 THEN GOSUB 630
610 IF P(1,1)=4 AND Q(1,1)=4 GOTO 650
620 RETURN
630 IF I=9 THEN PRINT:PRINT "YOU BLEW IT, IDIOT."
640 IF I=9 THEN PRINT:PRINT "THE RIGHT COMBINATION WAS";M;X;Y;Z
650 PRINT:PRINT:INPUT "DO YOU WANT TO PLAY AGAIN ";NS
660 IF ASC(NS)=72ORASC(NS)=110 THEN PRINT "X THANKS FOR PLAYING WITH ME.":STOP
670 GOTO 140
680 REM
690 IF I=1 THEN PRINT:PRINT
700 IF I=1 THEN PRINT "NOW, THAT WAS TOO EASY."
710 IF I=1 THEN PRINT:PRINT "OR ELSE YOU CHEATED."
720 IF I=1 THEN PRINT:PRINT "TRY AGAIN."
730 IF I=1 GOTO 120
740 IF I=2 THEN PRINT:PRINT
750 IF I=2 THEN PRINT "YEAH, YOU GOT IT. BUT DON'T GET"
760 IF I=2 THEN PRINT:PRINT "SO EXCITED. PURE LUCK. TRY AGAIN."
":GOTO 120
770 FOR S=3 TO 7
780 IF I=S THEN PRINT CHR$(12)
790 IF I=S THEN PRINT " HEY, NOT BAD, SPORT!!"
800 IF I=S THEN PRINT:PRINT:PRINT:PRINT:PRINT
810 IF I=S THEN PRINT " YOU GOT IT IN ONLY";I;" TRIES!!!"
920 IF I=S THEN PRINT:PRINT:PRINT:PRINT:PRINT
830 IF I=S THEN PRINT "FAAAAARRRRR FREEEAAAKKKKINNN' OUUUUUUUUUU
!!!"
840 IF I=S THEN PRINT:PRINT:PRINT "I'M IMPRESSED!"
850 IF I=S THEN RETURN
860 NEXT S
870 FOR S=8 TO 9
880 IF I=S THEN PRINT:PRINT:PRINT "EHI YOU DID OKAY.";I;" TRIES."
890 IF I=S THEN PRINT:PRINT:PRINT "BUT YOU CAN DO BETTER THAN THAT, RIGHT ?"
900 IF I=S THEN RETURN
910 NEXT S
920 PRINT CHR$(12)
930 PRINT:PRINT "CHEATER! CHEATER!! CHEATER!!!"
940 PRINT "YOU GOTTA USE FOUR DIFFERENT NUMBERS, CHEATER."
950 IF I=9 THEN PRINT "AND THAT USED YOUR LAST GUESS, TOO."
960 IF I=9 THEN PRINT "YOU DON'T DESERVE ANOTHER CHANCE.":STOP
970 PRINT "AND THAT USED UP ONE OF YOUR GUESSES, TOO."
980 PRINT "TRY AGAIN, CHEATER."
990 PRINT
1000 F=1
1010 RETURN
1020 PRINT CHR$(12)
1030 PRINT TAB(20);"I AM THE COMPUTER GOD."
1040 PRINT "I WILL CHOOSE A FOUR-NUMBER COMBINATION."
1050 PRINT "YOUR JOB IS TO FIGURE OUT WHICH NUMBERS"
1060 PRINT "I HAVE CHOSEN, AND TO PUT THEM IN THE CORRECT"
1070 PRINT "POSITIONS, FROM THE CLUES I GIVE YOU.":PRINT:PRINT
1080 PRINT "HOW TO PLAY:"
1090 PRINT:PRINT "1) CHOOSE EITHER THE BEGINNER'S OR"
1100 PRINT " ADVANCED PLAYER'S GAME."
1110 PRINT " FOR BEGINNERS, I WILL CHOOSE"
1120 PRINT " NUMBERS BETWEEN ONE AND SIX."
1130 PRINT " FOR ADVANCED PLAYERS, I CHOOSE"
1140 PRINT " NUMBERS BETWEEN ONE AND NINE."
1150 PRINT:PRINT "2) ENTER YOUR GUESS (EXAMPLE: 1,2,3,4)."
1160 PRINT:PRINT "3) FOR EACH OF YOUR GUESSES, I WILL"
1170 PRINT " PRINT OUT A CHART LISTING ALL OF YOUR

```

```

1180 PRINT " GUESSES. NEXT TO THESE ARE COLUMNS
1190 PRINT " LABELED 'N' AND 'P'. 'N' IS THE NUMBER
1200 PRINT " OF NUMBERS YOU HAVE GUESSED CORRECTLY.
1210 PRINT " 'P' INDICATES HOW MANY OF 'N' ARE IN THE
1220 PRINT " CORRECT POSITIONS.
1230 PRINT:INPUT "PRESS RETURN TO CONTINUE";XXS
1240 PRINT CHR$(12)
1250 PRINT:PRINT "4) YOU HAVE NINE CHANCES TO GUESS THE
1260 PRINT " COMBINATION I HAVE CHOSEN.
1270 PRINT:PRINT "5) I WILL NEVER PICK A COMBINATION IN WHICH
1280 PRINT " A NUMBER IS REPEATED. THEREFORE, ANY
1290 PRINT " GUESS WHICH USES A NUMBER MORE THAN ONCE IS ILLE
GAL.
1300 PRINT:PRINT:PRINT:PRINT:PRINT
1310 RETURN
READY

```

```

10 REM DATABASE MANAGEMENT BY DAVE BRISTOR, AUG. 79
20 CLEAR 5000
30 DIM RECS(200)
40 PRINT CHR$(12):PRINT:PRINT TAB(15);"DATA STORAGE AND RETRIEVAL
SYSTEM
50 REM SET UP MACHINE SUBROUTINES IN LOW MEMORY
60 DATA 17,19,224,253,115,63,253,114,64,231
70 DATA 17,27,224,253,115,63,253,114,64,201
80 DATA 17,15,224,253,115,65,253,114,66,201
90 DATA 17,24,224,253,115,65,253,114,66,221
100 FOR I=10 TO 49
110 : READ A
120 : POKE I,A
130 NEXT I
140 REM COMMAND LOOP
150 PRINT:PRINT "SELECT AN OPTION:
160 PRINT
170 PRINT TAB(10);"1 - CONSTRUCT A NEW DATABASE
180 PRINT TAB(10);"2 - SAVE A FILE
190 PRINT TAB(10);"3 - LOAD A FILE
200 PRINT TAB(10);"4 - LIST FILE
210 PRINT TAB(10);"5 - ADD TO END OF FILE
220 PRINT TAB(10);"6 - SEARCH FOR ENTRY
230 PRINT TAB(10);"7 - END PROGRAM
240 PRINT:INPUT "NUMBER ";OP$:IF ASC(OP$) > 57 OR ASC(OP$) < 49
THEN 240
250 OP=VAL(OP$)
260 ON OP GOTO 300,640,820,980,1040,1150,270
270 REM END PROGRAM
280 CLEAR
290 END
300 REM CONSTRUCT NEW DATABASE
310 PRINT:INPUT "HOW MANY RECORDS WILL YOU ENTER NOW ";N
320 FOR I=1 TO N
330 : GOSUB 370
340 NEXT I
350 PRINT CHR$(12):GOTO 150
360 REM CONSTRUCT RECORD
370 : PRINT:PRINT "ENTER RECORD ";I:PRINT "TYPE EACH PART AND
RETURN: ";
380 : PRINT "WHEN FINISHED TYPE 0 AND RETURN.
390 : INPUT "FIRST PART ";P1

```

```

400 : IF LEN(REC$(I))+LEN(R$) < 255 THEN 430
410 : PRINT "TOO LONG OR RECCRD FULL - MAXIMUM LENGTH 255 CHARA
CTERS.
420 : GOTO 370
430 : REC$(I)=REC$(I)+R$
440 : INPUT "NEXT PART ";R$
450 : IF R$="0" THEN REC$(I)=REC$(I)+CHR$(173):GOTO 430
460 : R$=CHR$(142)+R$
470 : GOTO 400
480 : PRINT:PRINT "RECORD ";I;"LOOKS LIKE THIS:
490 : GOSUB 540
500 : PRINT:INPUT "IS THIS RECORD OK AS IT STANDS (Y/N) ";O$
510 : IF O$<>"N" AND O$<>"N" AND O$<>"Y" AND O$<>"Y" THEN 5
00
520 : IF O$="N" OR O$="N" THEN REC$(I)="" :GOTO 370
530 RETURN
540 REM PRINTING ROUTINE
550 ST=1
560 FOR J=1 TO LEN(REC$(I))
570 : IF MID$(REC$(I),J,1)<>CHR$(142) THEN 510
580 : IF MID$(REC$(I),J,1)=CHR$(173) THEN 520
590 : PRINT MID$(REC$(I),ST,J-ST);" ";
600 : ST=J+1
610 NEXT J
620 PRINT MID$(REC$(I),ST,J-ST-1)
630 RETURN
640 REM SAVE FILE
650 PRINT:PRINT "PUT YOUR DATA CASSETTE IN THE RECORDER, REWIND
IT,
660 PRINT "PRESS PLAY AND RECORD AND WAIT FOR THE LEADER TO PASS
.
670 INPUT "RETURN WHEN READY":X$
680 REM CHANGE OUTPUT TO TAPE
690 POKE 260,10:POKE 261,0:XX=USR(0)
700 FOR J=1 TO 100:NEXT J
710 PRINT ""
720 REC$(0)=STR$(N)
730 FOR I=0 TO N
740 : PRINT REC$(I)
750 : FOR J=1 TO 100:NEXT J
760 : IF I=0 THEN FOR J=1 TO 1000:NEXT J
770 NEXT I
780 REM CHANGE OUTPUT BACK TO VIDEO
790 POKE 260,20:POKE 261,0:XX=USR(0)
800 PRINT:PRINT "FILE SAVED - REWIND DATA TAPE FOR FUTURE USE.
810 GOTO 150
820 REM LOAD FILE
830 PRINT:PRINT "PUT YOUR DATA CASSETTE IN THE RECORDER,
840 PRINT "REWIND IT, AND PRESS PLAY.
850 INPUT "RETURN WHEN READY":X$
860 REM CHANGE INPUT TO TAPE
870 POKE 260,30:POKE 261,0:XX=USR(0)
880 INPUT X$
890 INPUT N$:V$=RIGHT$(V$,LEN(N$)-1):N=VAL(N$)
900 FOR I=1 TO N
910 : INPUT REC$(I)
920 : REC$(I)=RIGHT$(REC$(I),LEN(REC$(I))-1)
930 NEXT I
940 REM CHANGE INPUT BACK TO KEYBOARD
950 POKE 260,40:POKE 261,0:XX=USR(0)
960 PRINT:PRINT "FILE LOADED - REWIND DATA TAPE FOR FUTURE USE.
970 GOTO 150

```

```

980 REM LIST FILE
990 FOR I=1 TO N
1000 : PRINT:PRINT "RECORD ";I
1010 : GOSUB 540
1020 NEXT I
1030 GOTO 150
1040 REM ADD TO END OF FILE
1050 PRINT:INPUT "HOW MANY RECORDS DO YOU WANT TO ADD ";N1
1060 PRINT "THERE ARE";N;"RECORDS IN THE FILE
1070 PRINT "THE FIRST NEW RECORD ADDED IS ";N+1
1080 R=2:REM SET RETURN LOCATION
1090 FOR I=N+1 TO N1+N
1100 : GOSUB 370
1110 NEXT I
1120 N=N+N1
1130 PRINT CHR$(12)
1140 GOTO 150
1150 REM SEARCH FOR ENTRY
1160 PRINT:INPUT "ENTER ITEM TO BE SEARCHED FOR ";SR$
1170 LN=LEN(SR$)
1180 FOR I=1 TO N
1190 : FOR J=1 TO LEN(REC$(I))
1200 : IF MID$(REC$(I),J,LN) = SR$ THEN PRINT:GOSUB 540:F=1
1210 : NEXT J
1220 NEXT I
1230 IF F=1 THEN 1250
1240 PRINT:PRINT "ITEM ";SR$;" NOT IN ANY RECORD IN FILE.
1250 GOTO 150
READY

```