Apple /// Text File Case Converter

by Mike Kramer

Houston Area Apple Users' Group

Having been an avid Apple II user for almost three years when I bought an Apple ///, I had come to know and love a number of good Applesoft BASIC programs which I wanted to run under Apple /// Business BASIC. Before I had gathered up the courage to attempt any significant conversions, I was given a copy of Apple-CON (available from the IAC for $10). This program, which has been placed in the public domain by Apple Computer, reads an Applesoft program from a DOS disk, converts it to Business BASIC, and stores it on a SOS disk. Unlike the old Integer BASIC to Applesoft converters on the Apple II, Apple-CON translates Applesoft syntax to Business BASIC syntax. The resulting program does usually require some cleanup to eliminate obscure PEEKs and POKEs or other code that Apple-CON cannot convert. Lines containing such code are flagged with a preceding REM statement containing a row of dashes.

One of many nice features of Apple /// Business BASIC is that keywords entered in lower case are converted to upper case when the program is listed. Hence the whole program can be typed in lower case and it will end up with variables in lower case and the BASIC commands in upper case. This proves particularly valuable in verifying correct syntax, since incorrect commands are not converted to upper case.

Since I had become accustomed to having lower case variable names, I was disappointed, but not surprised, to find that programs converted by Apple-CON retained whatever case was in the original Applesoft program. Being lazy by nature and having better sense than to risk introducing errors by retyping the variable names in lower case, I decided to write a program to convert the resulting Apple-CON text file to lower case. Then when the converted program was EXECd into memory, the variables would be lower case and the keywords in upper case. After finishing the first version of the program, I decided that optional conversion to upper case could be provided with little extra code. The program *Text File Case Converter* is the result.

How to Use it

The output file from Apple-CON or any other SOS text file can be converted either to upper case or to lower case. The disk containing the text file should be placed in one drive and the disk containing Text File Case Converter in the other. If you wish, the source and target files can both be on the same disk. Then Text File Case Converter should be RUN.

The first question asked is whether the file is to be converted to upper case or to lower case. Either a "U" or an "L" is sufficient. Next, you are asked for the pathname of the file to be converted. You may end execution by typing "END" or pressing RETURN, or you may list the directory by typing "CAT". If you list the directory, you are again asked for the pathname of the file. The program will immediately attempt to OPEN the named file. If the file cannot be OPENed for any reason, you are asked to enter another name. Next you are asked for the pathname of the target file. You may optionally END or list the directory at this point. If an improper pathname is entered, you will asked to enter another.

Finally, you are asked if you also want the results listed on a display device. If your response is "YES" or "Y", you will be asked for an output device name. Initially the default is .CONSOLE. If you enter another device name, it will be the default on the next run. If the device specified has not been included in the SOS.DRIVER file on your boot disk, you will be asked to enter another device name. If you have made it through all the dialogue, you will be rewarded with a text file containing the converted file. If you obtained the results on a list device you were able to see (with some sacrifice in execution speed) that the conversion was done as requested. To confirm that the conversion was done, fire up Apple Writer or some other text editor and look at the file.

Listing 1 shows a short Applesoft program that was converted to Apple /// BASIC using Apple-CON. Since there were no particularly unusual commands in the Applesoft program, there is no difference in the code generated for the Apple ///. Listing 2 shows the same program after processing with Text File Case Converter. Note that all BASIC keywords in Listing 1 are in upper case. Note also that you may have to go back and manually change some characters in the resulting program to upper case.

By the way, the program in Listing 1 (or 2) is a good example of the Shell-Metzner sorting technique which you may want to add to your library. It allows entry of up to 20 names (first and last), breaks the name into first and last names (provided there is an intervening space), sorts on

last names, and lists the names as entered and in sorted order.

How the Case Converter Works

Text File Case Converter is actually a simple program, consisting of a loop that reads a line from one TEXT file, steps through the line looking at each character for alpha characters to convert, performs the conversion, and writes the resulting line to another TEXT file. The rest of the program is dialog with the user and error handling. The following comments describe how the program works.

Lines 260-300: Perform some initialization and display the title block, setting a window to leave it on the screen.

Line 310: Determine if the file is to be converted to all upper case or all lower case.

Listing 1
After APPLECON

```
100   REM         SORT DEMO
110   REM  WRITTEN BY MIKE KRAMER
120   REM          1/1/83
130   DIM R(20),G$(20,2)
140   TEXT:HOME
150   NR=1
160   PRINT:INPUT"NAME:";NA$:IF NA$="" OR NR=20 THEN 250
170   NR=NR+1:R(NR-1)=NR-1
180   FLAG=0
190   FOR J=1 TO LEN(NA$)
200     IF MID$(NA$,J,1)=CHR$(32) THEN G$(R(NR-1),1)=LEFT$(NA$,
      J-1):G$(R(NR-1),2)=RIGHT$(NA$,LEN(NA$)-J):FLAG=1
210     NEXT J
220   IF FLAG=0 THEN PRINT:PRINT"ENTER FIRST & LAST NAME WITH S
      PACE"
230   GOTO 160
240   REM   SORT FILE
250   M=NR-1
260   M=INT(M/2):IF M=0 THEN 330
270   L=1
280   I=L
290   IF G$(R(I),2)<=G$(R(I+M),2) THEN 310
300   X=R(I):R(I)=R(I+M):R(I+M)=X:I=I-M:IF I>=1 THEN 290
310   L=L+1:IF L>(NR-M-1) THEN 260
320   GOTO 280
330   FOR J=1 TO NR-1
340     PRINT G$(J,1);" ";G$(J,2),G$(R(J),1);" ";G$(R(J),2)
350     NEXT
```

Listing 2
After TEXT FILE CASE CONVERTER

```
100   REM         sort demo
110   REM  written by mike kramer
120   REM          1/1/83
130   DIM r(20),g$(20,2)
140   TEXT:HOME
150   nr=1
160   PRINT:INPUT"name:";na$:IF na$="" OR nr=20 THEN 250
170   nr=nr+1:r(nr-1)=nr-1
180   flag=0
190   FOR j=1 TO LEN(na$)
200     IF MID$(na$,j,1)=CHR$(32) THEN g$(r(nr-1),1)=LEFT$(na$,
      j-1):g$(r(nr-1),2)=RIGHT$(na$,LEN(na$)-j):flag=1
210     NEXT j
220   IF flag=0 THEN PRINT:PRINT"enter first & last name with s
      pace"
230   GOTO 160
240   REM   sort file
250   m=nr-1
260   m=INT(m/2):IF m=0 THEN 330
270   l=1
280   i=l
290   IF g$(r(i),2)<=g$(r(i+m),2) THEN 310
300   x=r(i):r(i)=r(i+m):r(i+m)=x:i=i-m:IF i>=1 THEN 290
310   l=l+1:IF l>(nr-m-1) THEN 260
320   GOTO 280
330   FOR j=1 TO nr-1
340     PRINT g$(j,1);" ";g$(j,2),g$(r(j),1);" ";g$(r(j),2)
350     NEXT
```

Lines 390-410: Get pathname of the file to be converted, list directory, or END.

Lines 420-440: Set variable "which$" to indicate to error routine where error occurred, set up for branch to error routine, and OPEN file to be converted.

Lines 520-560: Get pathname of destination file as done above for source file and OPEN file.

Lines 640-700: See if converted text is also to be listed on display device. Default is initially .CONSOLE, although the next run will use the last device name entered. When RETURN is pressed, the cursor is positioned in the proper line and the device name is output. Line 690 assures correct positioning of device name if the cursor is on the bottom line of the screen when RETURN is pressed. Line 700 OPENs the device for output.

Lines 780-790: Prepare to continue execution at Line 1040 when the end of the file is reached. INPUT line of text from the file to be converted.

Lines 870-880: Initialize temporary string variables to null. Determine whether to convert to upper case or lower case and set conversion variables accordingly.

Lines 890-920: Convert to selected. Step through each character of the line read from the file. If the conversion is to lower case and the character's ASCII code is in the range 65 through 90 (A-Z) subtract 32 to convert to lower case. If the conversion is to upper case and the character's ASCII code is in the range 97 through 122 (A-Z) add 32 to convert to upper case. Concatenate characters into new string "temp$".

Lines 1000-1030: Output converted line to destination file and to the optional display device. Return to Line 790 to read another line.

Line 1160: CLOSE the files and the optional display device.

Lines 1120-1130 See if another conversion is wanted. If not, reset the window, clear screen, and END.

Lines 1210-1350: Check for error conditions and set up proper error message. If none of conditions in Lines 1210 - 1330 exists then display error code and line number and END. If one of these conditions exists, display message and return to appropriate line according to current value of "which$".

Listing 3
TEXT FILE CASE CONVERTER

```
100  REM    ***********************************************
110  REM    **                                           **
120  REM    **          Text File Case Converter         **
130  REM    **                                           **
140  REM    **           Written by Mike Kramer          **
150  REM    **       Houston Area Apple Users Group      **
160  REM    **                12/27/82                   **
170  REM    **                                           **
180  REM    ***********************************************
190  REM
200  REM    ***********************************************
210  REM    **                                           **
220  REM    **             Print Title Block             **
230  REM    **                                           **
240  REM    ***********************************************
250  REM
260  bell$=CHR$(7):blank$="
                                                    ":output.devi
     ce$=".CONSOLE"
270  TEXT:HOME:INVERSE:VPOS=1:FOR i=1 TO 5:PRINT blank$:NEXT i
280  VPOS=2:HPOS=16:PRINT"***          Text File Case Converter
          ***":HPOS=16:PRINT"***          Written by Mike K
     ramer          ***":HPOS=16:PRINT"***      Houston Area Ap
     ple Users Group     ***":NORMAL
290  WINDOW 0,6 TO 80,24
300  HOME:VPOS=2
310  INPUT"Convert file to upper case or lower case? ";case$:c
     ase$=LEFT$(case$,1):IF case$<>"U" AND case$<>"u" AND case
     $<>"L" AND case$<>"l" THEN 310
320  REM
330  REM    ***********************************************
340  REM    **                                           **
350  REM    **      Get Name of File to Be Converted     **
360  REM    **                                           **
370  REM    ***********************************************
380  REM
390  PRINT:INPUT"Pathname of text file to be converted, CAT, E
     ND: ";source.pathname$
400  IF source.pathname$="CAT" OR source.pathname$="cat" OR so
     urce.pathname$="Cat" THEN CATALOG:GOTO 390
410  IF source.pathname$="" OR source.pathname$="END" OR sourc
     e.pathname$="End" OR source.pathname$="end" THEN 1130
420  which$="source"
430  ON ERR GOSUB 1210
440  OPEN#1 AS INPUT,source.pathname$
450  REM
460  REM    ***********************************************
470  REM    **                                           **
480  REM    **           Get Name of Output File         **
490  REM    **                                           **
500  REM    ***********************************************
510  REM
520  PRINT:INPUT"Pathname of target text file, CAT, END: ";tar
     get.pathname$
530  IF target.pathname$="CAT" OR target.pathname$="cat" OR ta
     rget.pathname$="Cat" THEN CATALOG:GOTO 520
540  IF target.pathname$="" OR target.pathname$="END" OR targe
     t.pathname$="End" OR target.pathname$="end" THEN 1130
550  which$="target"
560  OPEN#2 AS OUTPUT,target.pathname$
570  REM
580  REM    ***********************************************
590  REM    **                                           **
600  REM    **         Get Name of Hardcopy Device       **
610  REM    **                                           **
620  REM    ***********************************************
630  REM
640  PRINT:PRINT"List program lines on output device? (Y/N) ";
     :GET yes.or.no$:PRINT yes.or.no$:IF yes.or.no$<>"Y" AND y
     es.or.no$<>"y" THEN 780
650  PRINT:vtab= VPOS:PRINT"Pathname for output: ";output.devi
     ce$;:HPOS=22:INPUT"";response$
660  IF response$="" THEN 690
670  output.device$=response$
680  which$="display"
690  VPOS=vtab-1*(vtab=19):HPOS=22:PRINT output.device$
700  OPEN#3 AS OUTPUT,output.device$
```

Listing 3 continued

```
710   REM
720   REM   ***********************************************
730   REM   **                                           **
740   REM   **     Read Lines of Text From Source File   **
750   REM   **                                           **
760   REM   ***********************************************
770   REM
780   ON EOF#1 GOTO 1040
790   INPUT#1;line$
800   REM
810   REM   ***********************************************
820   REM   **                                           **
830   REM   **       If Alpha Convert to Proper Case     **
840   REM   **                                           **
850   REM   ***********************************************
860   REM
870   temp$="":c$=""
880   IF case$="L" OR case$="l" THEN lower.code=65:upper.code=9
      0:delta=32:ELSE lower.code=97:upper.code=122:delta=-32
890   FOR i=1 TO LEN(line$)
900     IF ASC(MID$(line$,i,1))>=lower.code AND ASC(MID$(line$,
        i,1))<=upper.code THEN c$=CHR$(ASC(MID$(line$,i,1))+del
        ta):ELSE c$=MID$(line$,i,1)
910     temp$=temp$+c$
920     NEXT i
930   REM
940   REM   ***********************************************
950   REM   **                                           **
960   REM   **     Output to Specified Destination(s)    **
970   REM   **                                           **
980   REM   ***********************************************
990   REM
1000    IF yes.or.no$="n" OR yes.or.no$="N" OR yes.or.no$="" THE
        N 1020
1010    PRINT#3;temp$
1020    PRINT#2;temp$
1030    GOTO 790
1040    CLOSE
1050    REM
1060    REM   ***********************************************
1070    REM   **                                           **
1080    REM   **      Check if Another Conversion Wanted   **
1090    REM   **                                           **
1100    REM   ***********************************************
1110    REM
1120    PRINT:PRINT"Another conversion? (Y/N) ";:GET yes.or.no$:
        IF yes.or.no$="Y" OR yes.or.no$="y" THEN 300
1130    TEXT:HOME:END
1140    REM
1150    REM   ***********************************************
1160    REM   **                                           **
1170    REM   **                Error Handling             **
1180    REM   **                                           **
1190    REM   ***********************************************
1200    REM
1210    IF ERR=23 THEN message$="Files Busy":GOTO 1350
1220    IF ERR=25 THEN message$="I/O Error":GOTO 1350
1230    IF ERR=26 THEN message$="File Too Large":GOTO 1350
1240    IF ERR=27 THEN message$="Disk is Write Protected":GOTO 1
        350
1250    IF ERR=29 THEN message$="Bad Path":GOTO 1350
1260    IF ERR=30 THEN message$="File Not Found":GOTO 1350
1270    IF ERR=31 THEN message$="Path Not Found":GOTO 1350
1280    IF ERR=32 THEN message$="Volume Not Found":GOTO 1350
1290    IF ERR=33 THEN message$="Duplicate File":GOTO 1350
1300    IF ERR=34 THEN message$="Disk Full":GOTO 1350
1310    IF ERR=35 THEN message$="File Locked":GOTO 1350
1320    IF ERR=36 THEN message$="File Not Open":GOTO 1350
1330    IF ERR=37 THEN message$="Device Not Found":GOTO 1350
1340    HOME:VPOS=12:HPOS=30:PRINT"Error "; ERR;" in Line "; ERR
        LIN:TEXT:END
1350    PRINT:PRINT bell$;:PRINT"** ";message$;" **":POP:IF whic
        h$="source" THEN 390:ELSE IF which$="target" THEN 520:EL
        SE 650
```