High Octane Transfer For Atari

Steve Kaufman

Data storage and retrieval with disk files can be a slow process. Using Atari's unique string length capability, the author presents programming techniques which will read a file of 10,000 characters in 14 seconds.

The storage and retrieval of data is a relatively simple procedure on most microcomputer systems. Unfortunately, the speed of the process can leave a great deal to be desired.

The programs throughout this article will illustrate a method of achieving high-speed data transfer to disk. Let's start with Program 1, which will create a file of names and occupations.

Program 1: Names And Occupations File

```
10 DIM NAME$(30),OCC$(15)
20 OPEN #1,8,0,"D:NAMEOCCU.DAT"
30 ? "NAME (RETURN ONLY TO END)";::IN
   PUT NAME$
40 IF LEN(NAME$)=0 THEN CLOSE #1:END
50 ? "OCCUPATION";:INPUT OCC$
60 ? #1;NAME$;CHR$(155);OCC$
70 GOTO 30
```

This program is straightforward and simple, but simplicity does not imply efficiency. It does not cause disk access after every entry because DOS will write only to the disk when its sector buffer is full (every 125 bytes on single density drives, 253 on double density), but it does waste a lot of disk time in cases where a long file is being created.

Storing Data In Strings

More important, however, is the status of the data. Once a new name and occupation are entered, all the prior data is eliminated from RAM. It's on the disk, to be sure, but it becomes a rather tedious process to retrieve and modify it. A far better procedure, in terms of both disk access time and data manipulation flexibility, is to store all the data, as it is generated, in a single long string as demonstrated in Program 2.

Program 2: String Storage Of Data

```
10 DIM NAME$(20),OCC$(15),A$(10000)
20 ? "NAME (RETURN ONLY TO END)";::IN
   PUT NAME$
30 IF LEN(NAME$)=0 THEN GOTO 100
40 ? "OCCUPATION";:INPUT OCC$
50 REM now pad the data with spaces
   so that each entry is the same le
   ngth
60 IF LEN(NAME$)<20 THEN NAME$(LEN(N
   AME$)+1)=" ":GOTO 60
70 IF LEN(OCC$)<15 THEN OCC$(LEN(OCC
   $)+1)=" ":GOTO 70
80 A$(LEN(A$)+1)=NAME$:A$(LEN(A$)+1)
   =OCC$
90 GOTO 20
100 FOR X=0 TO LEN(A$)/35-1
110 PRINT A$(X*35+1,X*35+20);"
    {4 SPACES}";A$(X*35+21,X*35+35)
120 NEXT X
```

In this example, though we've wasted what may turn out to be a substantial amount of space by padding the data fields with blanks (lines 60-70) so that they're all the same length, the padding creates a series of easily accessed records that can be edited or (as in lines 100-120) printed out in a formatted fashion without any difficulty. Equally significant is the speed with which the entire file of data can be sent to the disk (or cassette):

```
200 OPEN #1,8,0,"D:NAMEOCCU.DAT"
210 PRINT #1;A$:CLOSE #1
```

In addition, in this instance, even after you have sent the file to the disk, it is still sitting in RAM (in A$) where it can be manipulated further. You could also combine the two methods, writing each field of data to the disk and putting it into the long string A$ at the same time.

Note that Atari BASIC provides you with a substantial speed advantage in disk time here over most other microcomputer BASICs inasmuch as it allows strings to be up to 32K long. In other BASICs (for example, Microsoft BASIC on the Atari), the file would have to be saved in an array of strings (or in this case perhaps in two arrays,

NAME$( ), and OCC$( )), and the save to the disk would have to be done in a significantly slower FOR/NEXT loop:

```
10 FOR X=1 TO ARRAYSIZE
20 PRINT #1,NAME$(X)
30 PRINT #1,OCC$(X)
40 NEXT X
```

Data Retrieval By INPUT

While creating and writing data files is a relatively straightforward procedure, reading the files is by no means as simple, or, in many cases, as rapid. The easiest way in BASIC to read such files is to use the INPUT statement. This would work very well on the data file created in Program 1. Indeed it is a simple matter to read in the data, pad each field with blanks, and put it all into a long string as in Program 3.

Program 3: Reading Data With INPUT

```
10 DIM NAME$(20),OCC$(15),A$(10000)
20 OPEN #1,4,0,"D:NAMEOCCU.DAT"
30 TRAP 200
40 INPUT #1,NAME$:INPUT #1,OCC$
50 IF LEN(NAME$)<20 THEN NAME$(LEN(N
   AME$)+1)=" ":GOTO 50
60 IF LEN(OCC$)<15 THEN OCC$(LEN(OCC
   $)+1)=" ":GOTO 60
70 A$(LEN(A$)+1)=NAME$:A$(LEN(A$)+1)
   =OCC$:GOTO 40
200 CLOSE #1
```

INPUT in Atari BASIC is relatively problem-free. Unlike most BASICs, you can use it easily even if the data contains commas or colons which would signal end of field to the INPUT statement of other BASICs (that is, it really works more like the LINPUT/LINE INPUT available in some dialects).

The problem with the INPUT statement is that it can be used only when there is a carriage return character (ATASCII 155) in the file after every line of text. The file created by Program 1 has such carriage returns embedded in it automatically because every PRINT statement (not followed by a semicolon or comma) automatically attaches a carriage return to the data. This is the reason we had to embed a CHR$(155) in our data in line 60 of Program 1. Alternatively, we could have written ?#1;NAME$:?#1;OCC$.

In Program 2 we do not have carriage returns embedded in the data. The normal way to retrieve such files in BASIC is to use the GET statement, which retrieves one character at a time:

```
10 DIM A$(10000)
20 OPEN #1,4,0,"D:NAMEOCCU.DAT"
30 TRAP 200 :REM to catch the "end o
   f file" error
40 GET #1,X:A$(LEN(A$)+1)=CHR$(X):GO
   TO40
200 CLOSE #1
```

The problem with this procedure is that it is inordinately slow, much slower than using INPUT,

and INPUT itself doesn't even come close to the speed of data access that the disk drive is capable of providing. In this particular case we could revise Program 2 by inserting a CHR$(155) after every field in the storage string A$, then we could use INPUT to retrieve it.

Reading Files Of Unknown Length

There are many kinds of data in which carriage return characters might not only be inappropriate but also might be found within data fields themselves, so that INPUT could not be used. Reading in a segment of machine language or the data for a graphics display would be two common examples. Some computers provide an easy way to directly and quickly store and retrieve chunks of RAM memory, such as the commands BSAVE and BLOAD in Applesoft.

Experienced users realize that the Atari operating system supports such operations, but even with them, the user must declare in advance the exact length of the file to be transferred.

The challenge is to find a simple routine that will read a file from disk into memory at maximum speed, even though we don't know in advance how long the file is. Simply knowing that we have reached end-of-file is not enough. Let us say, for example, that we have provided a 10,000-character block of memory (DIM A$(10000)) in which to store our file. How do we know how far into that memory block the file has reached? No problem. It turns out that the operating system keeps track for us, in RAM locations 40 and 41, of exactly how many bytes are transferred during a Central Input/Output (CIO) operation.

The following routine, then, can be ENTERed into any BASIC program to initialize a string (FAST$) containing machine language which allows us optimum-speed file access:

```
30000 RESTORE 30001:FOR I=1 TO 39:RE
      AD A:FAST$(LEN(FAST$)+1)=CHR$(
      A):NEXT I:GOTO 10
30001 DATA 104,104,104,10,10,10,10,1
      70,169,7,157,66,3,104,157,69,3
      ,104,157,68,3,104,157,73,3
30002 DATA 104,157,72,3,32,86,228,13
      2,212,160,0,132,213,96
```

In order to use this routine, you need only open the file normally in BASIC and do a USR function call:

   Y=USR(ADR(FAST$),CHANNEL,ADR(A$),BSIZE)

where Y is the status returned by the CIO call, CHANNEL is the channel number that you have assigned to this file in your OPEN statement, A$ is the buffer string into which the file will be read, and BSIZE is the maximum length of that buffer (normally the value to which you have DIMensioned A$). Using this procedure on a file of 10,000 characters, it takes only 14 seconds (9 if you have double density) to read the file into RAM. This is

as fast as the drive is capable of reading. Using INPUT, where possible, takes about 42 seconds to transfer the same number of bytes. Using GET to perform this function takes 2½ minutes.

Program 4 is a simple BASIC program using this function that allows you to read any file off disk and print it on the screen. Note that if the USR call returns a status of 136, the end-of-file has been reached. If the status is 1, the file is longer than the buffer you have provided. Any other status indicates an error situation.

This routine can also be used for saving data directly from RAM to a disk file. Simply substitute the value 11 (the binary PUT command value) for the value 7 in the DATA statements or in FAST$ after it has been initialized (FAST$(10,10) = CHR$(11)). Using such a modified FAST$, the following routines will save an entire GRAPHICS 0 screen to disk:

```
10 OPEN #1,8,0,"D:GRØ.DAT"
20 START=PEEK(88)+256*PEEK(89)
30 Y=USR(ADR(FAST$),1,START,40*24)
40 CLOSE #1: IF Y<>1 THEN PRINT "ERR
   OR ";Y
```

Change FAST$(10,10) back to CHR$(7), and you can put the data right back on the screen just as quickly:

```
10 OPEN #1,4,0,"D:GRØ.DAT"
20 START=PEEK(88)+256*PEEK(89)
```

```
30 Y=USR(ADR(FAST$),1,START,40*24)
40 CLOSE #1
```

Program 4: Data Retrieval By A USR Call

```
Ø  BSIZE=10000:DIM FAST$(39),A$(BSIZE
   ),FNAME$(15)
5  A$(1)="{,}":A$(BSIZE)="{,}":A$(2)=
   A$(1):Z=Ø:GOTO 30000
10 ? "NAME OF FILE TO READ";:INPUT F
   NAME$
100 OPEN #1,4,Ø,FNAME$
110 Y=USR(ADR(FAST$),1,ADR(A$),BSIZE
    )
115 Z=PEEK(40)+256*PEEK(41)
116 REM Y is the error status, Z is
    the number of bytes read from th
    e file
119 REM now tell BASIC how long our
    storage string is
120 A$(Z)=A$(Z,Z)
125 IF Y=136 THEN ? A$:? "END OF FIL
    E":END
130 IF Y=1 THEN ? A$:? "RETURNING TO
    GET REMAINDER OF FILE":GOTO 110
139 REM any other value of Y is a di
    sk error
140 ? "ERROR #";Y:END
30000 RESTORE 30001:FOR I=1 TO 39:RE
      AD A:FAST$(LEN(FAST$)+1)=CHR$(
      A):NEXT I:GOTO 10
30001 DATA 104,104,104,10,10,10,10,1
      70,169,7,157,66,3,104,157,69,3
      ,104,157,68,3,104,157,73,3
30002 DATA 104,157,72,3,32,86,228,13
      2,212,160,0,132,213,96              ©
```

www.commodore.ca

U-boat

Mark Vittek

"U-boat," for VIC and 64, starts out quietly enough, but don't be lulled into a false sense of security. Without warning, you'll be called upon to use all your wits to save yourself.

"U-boat" is a nice change of pace from fast action, run-and-shoot games – for a while.

As you calmly patrol the ocean floor, the serenity is soon broken by the sudden explosion of a depth charge. Above you, on the ocean surface, are destroyers on a search-and-destroy mission. And if that isn't bad enough, there is a constant supply of mines to keep you busy.

The object of the game is to sink as many ships as possible before their depth charges reach you. On the surface of the sea, destroyers and aircraft carriers pass to and fro dropping their destructive cargo. Some drop randomly, but others zero in on your U-boat.

You can destroy bothersome mines with your torpedoes, but new ones always replace them after five ships have been sunk. So in order to keep your patrol area tidy, you have to take time out every now and then to remove mines. Also, it's important to remember that when the charges and mines explode, they take the surrounding eight cursor spaces with them. If your U-boat is in this area, it is permanently retired.

VIC Version

For those of you who do not have a joystick, Program 3 shows the necessary replacement lines for Program 2 so that you can use the keyboard. I used the cursor keys to move the sub, but if you find that they are too difficult, you may assign new keys in lines 70-76.

If you do have a joystick, you may want to try the keyboard later on. You'll find it more of a challenge because the game plays faster. It may be a bit difficult at first, but if you master the game using the keyboard, you can become the scourge of the seven seas.

U-boat was designed to provide a large variety of objects on the screen without slowing the game down. As most game programmers know, too many POKEs and PRINTs can spoil the game's fast action quality. For example, the usual two POKEs to location 37154 have been omitted from the joystick reading routine.

Using MID$ is a good technique for achieving fast movement of many objects on screen. In Program 2, MID$ was used to move the ships across the screen because it was faster than POKEing. Also, when I first wrote the game with POKEs, the ship's bow plunged into the sea as the ship wrapped around the screen. This could be remedied with IF statements, but using MID$ turned out to be faster and easier. Since the number of characters displayed by MID$ is set at 22 (VIC's screen width), the ship disappears as if it just sailed out of view.

LEFT$, RIGHT$, and MID$ often provide for faster and more clever effects in game programming.

Type in and SAVE Program 1 first. This program will print instructions and redefine the necessary game characters. Reset the computer, then type in and SAVE Program 2 immediately following Program 1 on the same tape. When you RUN Program 1, it will automatically LOAD and RUN Program 2.

The 64 version plays the same as the VIC version but requires a joystick.

Janice Farv

Program 1: U-boat – VIC Version (Instructions And Loader)

```
BEGINNING PROGRAMMERS
If you're new to computing, please read ''How
To Type COMPUTE!'s Programs'' and ''A
Beginner's Guide To Typing In Programs.''
```

```
5 PRINT "{CLR}{6 DOWN}{BLU}{RVS}
  {8 SPACES}U-BOAT{8 SPACES}"
10 X=28:POKE52,X:POKE56,X:POKE51,PEEK(55)
20 FORI=7168TO7679:POKEI,PEEK(I+25600):NE
   XT
30 READT:IFT=-1THEN50
40 POKE7168+Q,T:Q=Q+1:POKE36879,T:GOTO30
50 POKE36879,28:PRINT"{3 DOWN} INSTRUCTIO
   NS? (Y/N)"
55 FORT=1TO1000:GETA$:IFA$="N"THENT=1000
60 IFA$="Y"THENT=1000:NEXT:GOTO100
65 NEXT:GOTO150
100 PRINT"{CLR}{DOWN} FIRE TORPEDOES FROM
    {3 SPACES}THE SUB'S FRONT{2 SPACES}TO
    {3 SPACES}SINK SHIPS AND "
105 PRINT" DESTROY MINES."
110 PRINT"{DOWN} BEWARE OF THE DEPTH
    {3 SPACES}CHARGES AND MINES{5 SPACES}
    DESTRUCTIVE AREA "
115 PRINT" WHEN THEY EXPLODE."
120 PRINT"{DOWN} DESTROY AS MANY SHIPS AS
     YOU CAN UNTIL YOU{2 SPACES}ARE PERMA
    NENTLY{7 SPACES}RETIRED."
125 FORT=1TO5000:NEXT:PRINT"{3 DOWN}{RVS}
     PRESS ANY KEY "
130 GETA$:IFA$=""THEN130
140 REM THE NEXT LINE{2 SPACES}WILL NEW T
    HE PROGRAM. DO NOT PUT IT IN WHEN TES
    TING{2 SPACES}PROGRAM.
150 PRINT"{CLR}":POKE198,5:FORT=631TO635:
    READX:POKET,X:NEXT
1000 DATA182,193,213,128,201,193,182,255,
     255,159,159,159,243,243,243,255,,,,,
     255,101,63
1002 DATA31,,96,112,252,255,148,255,255,,
     14,30,127,255,145,255,255,,,,,255,16
     6,252,248
1004 DATA,,4,4,15,10,127,63,,192,196,200,
     208,224,255,254
1006 DATA,3,35,19,11,7,255,127,,,32,32,24
     0,80,254,252,254,254,252,252,,133,20
     8,224
1008 DATA255,255,127,63,,80,3,31,255,255,
     254,254,,10,192,248
1010 DATA127,127,63,63,,161,11,7,255,239,
     239,239,239,199,215,255,65,41,68,175
     ,122,189
1012 DATA110,219,17,74,148,125,218,180,22
     1,182,255,255,255,255,255,255,255,25
     5
1014 DATA191,213,187,92,171,237,190,219,2
     47,174,181,125,235,189,173,222
1016 DATA-1,78,69,87,13,131
```

Program 2: U-boat – VIC Version (Main Program)

```
0 GOTO600
50 PA=PEEK(37152)AND128:P=PEEK(37151):IF-
   ((PAND32)=0)ORLTHENGOSUB150
70 IF-(PA=0)THENAY=AY+1:X=12:IFAY>20THENA
   Y=AY-1
72 IF-((PAND16)=0)THENAY=AY-1:X=10:IFAY<0
   THENAY=AY+1
74 IF-((PAND8)=0)THENAX=AX+22:IFAX>198THE
   NAX=AX-22
76 IF-((PAND4)=0)THENAX=AX-22:IFAX<-22THE
   NAX=AX+22
80 A=AX+AY+7900:P=PEEK(A):P0=PEEK(A+1):PO
   KEA1,17:POKEA1+1,17:POKEA,X:POKEA+1,X+
   1:A1=A
82 IFP<2ORP0<2THEN400
84 LM=LM+1:IFLM<RRTHEN92
86 N1=N1+22:CO=PEEK(N1):IFN1=AORN1=A+1THE
   NPOKEN1-22,17:GOTO400
88 IFN1>AORCO<3THEN300
90 POKEN1-22,17:POKEN1,1
92 IFM>VTHEN:PRINT"{HOME}{BLK}{7 DOWN}"MI
   D$(A$(TT),H,22);:H=H+M2:IFH=H1THENM=0:
   H=1
94 M=M+1:IFM<VTHEN50
96 IFM=VTHEN100
98 GOTO50
100 LM=0:N1=7856+AY:T=(RND(1)*6)+1:ONTGOT
    O110,108,106,104
102 TT=(RND(1)*2)+1:H=2:H1=26:M2=1:RR=22-
    (LM+AY):GOTO94
104 TT=(RND(1)*2)+3:H=26:H1=2:M2=-2:RR=-(
    LM-AY)/2:GOTO94
106 TT=(RND(1)*2)+3:H=28:H1=2:M2=-2:T=RND
    (1)*19:N1=7858+T:RR=3-(LM-T)/2:GOTO94
108 TT=4:H=28:H1=2:M2=-1:T=RND(1)*19:N1=7
    858+T:RR=3-(LM-T):GOTO94
110 TT=2:H=2:H1=26:M2=1:T=RND(1)*20:N1=78
    56+T:RR=22-(LM+T):GOTO94
150 L=L+1:IFL<2THENC=A:IFX=12THENC=A+1
152 C=C-22:CO=PEEK(C):IFCO<17THEN160
154 IFC<7856THENPOKEC+22,17:L=0:RETURN
156 POKEC+22,17:POKEC,14:RETURN
160 ONCO+1GOTO162,162:GOTO180
162 POKEZZ-2,(RND(1)*50)+155:FORT=-1TO1:F
    ORI=-1TO1:POKEZZ-1,10-T-I:C1=PEEK(C+I
    +T*22)
164 IFC1=1THENRR=99:LM=0
168 IFC1>9ANDC1<14THENM1=1
169 L=0:POKEC+I+T*22,(RND(1)*2)+18:NEXTI,
    T:POKEZZ-2,0:IFM1=1THEN400
170 FORT=-1TO1:FORI=-1TO1:POKEC+I+T*22,17
    :NEXTI,T:RETURN
180 PRINT"{HOME}{PUR}{7 DOWN}"MID$(A$(0),
    H,22):POKEC+22,17:POKEC-22,32:POKEZZ-
    2,(RND(1)*100)+140
182 FORT=15TO0STEP-.09:POKEZZ-1,T:NEXT:PR
    INT"{HOME}{7 DOWN}{22 SPACES}";
184 M=0:L=0:IFCO<7THENBS=BS+1
186 IFCO>6THENMS=MS+1
188 IFLM<RRTHENRR=99
190 IFMS+BS=5THENPOKEZZ,248:V=20
192 IFMS+BS=15THENPOKEZZ,232:V=10
194 IFMS+BS=30THENPOKEZZ,204
196 IF(MS+BS)/5=INT((MS+BS)/5)THENFORT=1T
    O8:POKE7878+(RND(1)*197),0:NEXT:J=1
198 IFJ=1THENFORT=-1TO2:FORI=-1TO1:POKEA+
    T+I*22,17:NEXTI,T:J=0
200 RETURN
300 J=15:RR=99:ML=0:POKEZZ-2,(RND(1)*30)+
    128:FORT=-1TO1:FORI=-1TO1:C1=PEEK(N1+
    I+T*22)
302 POKEZZ-1,J:J=J-1.6:IFC1=14THENL=0
304 IFC1>9ANDC1<14THENM1=1
306 POKEN1+I+T*22,(RND(1)*2)+18:NEXTI,T:P
    OKEZZ-2,0:IFM1=1THEN400
```

Cwww.commodore.ca

```
308 FORT=-1TO1:FORI=-1TO1:POKEN1+I+T*22,1
    7:NEXTI,T:GOTO92
309 POKEN1+I+T*22,(RND(1)*2)+18:NEXTI,T:P
    OKEZZ-2,0:IFM1=1THEN400
400 POKEZZ-1,15:POKEA,15:POKEA+1,16
402 FORT=1TO16STEP.15:POKEZZ-2,255-T*5:PO
    KEZZ,T*11:NEXT:POKEZZ-5,0
404 FORT=15TO0STEP-.2:POKEA,(RND(1)*2)+18
    :POKEA+1,(RND(1)*2)+18
406 POKEZZ-1,T:NEXT:POKEZZ-2,0:POKEZZ-10,
    240:POKEZZ,221:PRINT"{CLR}{BLK}{DOWN}
     YOU SANK"BS"CARRIERS"
408 PRINT"{DOWN}AND"MS"DESTROYERS":FORT=1
    TO2000:NEXT
410 PRINT"{3 DOWN} PLAY AGAIN ?":PRINT"
    {RVS}{DOWN} UP=YES DOWN=NO"
412 IF-((PEEK(37151)AND4)=0)THENRUN
414 IF-((PEEK(37151)AND8)=0)THENPOKE37154
    ,255:PRINT"{CLR} SO LONG.":END
416 GOTO412
600 ZZ=36879:POKEZZ,24:POKEZZ-10,255:POKE
    ZZ-1,15:POKE37154,127:X=10:V=30:AX=19
    8:AY=10
602 RR=30:H=2:PRINT"{CLR}{BLK}{RVS}
    {5 SPACES}U-BOAT{2 SPACES}26
    {7 SPACES}{OFF}"CHR$(8);
604 A$(1)="{21 SPACES}BDE{3 SPACES}":A$(2
    )="{21 SPACES}FG{3 SPACES}"
606 A$(3)="{25 SPACES}BCDE ":A$(4)="
    {25 SPACES}HI "
608 A$(0)="{22 SPACES}OPOP "
610 FORT=1TO4:PRINT"{YEL}QQQQQQQQQQQQQQQQQ
    QQQQQQ";:NEXT:PRINT"{3 DOWN}";:FORT=1
    TO12
612 PRINT"{BLU}QQQQQQQQQQQQQQQQQQQQQQQQQ";:N
    EXT
614 FORT=8120TO8186:POKET,17:POKE30720+T,
    0:NEXT:FORT=1TO15:POKE7878+(RND(1)*19
    7),0
618 NEXT:POKE8108,17:POKE8109,17:GOTO50
```
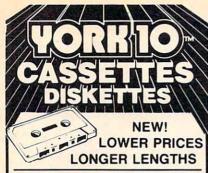
Program 3: U-boat – VIC Version
(Replacement Lines For Keyboard Control)

```
50 GETA$:IFA$=" "ORLTHENGOSUB150
70 IFA$="{RIGHT}"THENAY=AY+1:X=12:IFAY>20
   THENAY=AY-1
72 IFA$="{LEFT}"THENAY=AY-1:X=10:IFAY<0TH
   ENAY=AY+1
74 IFA$="{DOWN}"THENAX=AX+22:IFAX>198THEN
   AX=AX-22
76 IFA$="{UP}"THENAX=AX-22:IFAX<-22THENAX
   =AX+22
```

Program 4: U-boat – 64 Version

```
100 POKE55,0:POKE56,48:CLR:PRINT"{CLR}"CH
    R$(8):GOTO880
110 :
120 FORI=0TO27:POKEI+54272,0:NEXT
130 POKE54296,15:POKE54272,100:POKE54273,
    5:POKE54277,0:POKE54278,249
140 POKE54279,100:POKE54280,2:POKE54284,0
    :POKE54285,249
150 POKE54291,0:POKE54292,249:M=V:GOTO340
160 :
170 MV=1:P=PEEK(56320):IFP=127THENMV=0
180 IF(PAND16)=0ORLTHENGOSUB420
190 IF(PAND8)=0THENAY=AY+1:X=12:IFAY>38TH
    ENAY=AY-1
200 IF(PAND4)=0THENAY=AY-1:X=10:IFAY<0THE
```

```
    NAY=AY+1
210 IF(PAND2)=0THENAX=AX+40:IFAX>360THENA
    X=AX-40
220 IF(PAND1)=0THENAX=AX-40:IFAX<-80THENA
    X=AX+40
230 A=AX+AY+1424:P=PEEK(A):P0=PEEK(A+1):I
    FMVTHENPOKEA1,17:POKEA1+1,17
240 POKEA,X:POKEA+1,X+1:A1=A
250 IFP<2ORP0<2THEN760
260 LM=LM+1:IFLM<RRTHEN300
270 N1=N1+40:CO=PEEK(N1):IFN1=AORN1=A+1TH
    ENPOKEN1-40,17:GOTO760
280 IFN1>AORCO<3THEN680
290 POKEN1-40,17:POKEN1,1
300 IFM>VTHENPRINT"{HOME}{BLK}{7 DOWN}"MI
    D$(A$(TT),H,40);:H=H+M2:IFH=H1THENM=0
    :H=1
310 M=M+1:IFM=VTHEN340
320 GOTO170
330 :
340 LM=0:N1=1344+AY:T=RND(1)*6+1:ONTGOTO3
    90,380,370,360
350 TT=RND(1)*2+1:H=2:H1=44:M2=1:RR=40-(L
    M+AY):GOTO310
360 TT=RND(1)*2+3:H=44:H1=2:M2=-2:RR=-(LM
    -AY)/2:GOTO310
370 TT=RND(1)*2+3:H=46:H1=2:M2=-2:T=RND(1
    )*37:N1=1344+T:RR=3-(LM-T)/2:GOTO310
380 TT=4:H=46:H1=2:M2=-1:T=RND(1)*37:N1=1
    344+T:RR=3-(LM-T):GOTO310
390 TT=2:H=2:H1=44:M2=1:T=RND(1)*38:N1=13
    44+T:RR=40-(LM+T):GOTO310
400 :
410 REM TORP MOVE
420 L=L+1:IFL<2THENC=A:IFX=12THENC=A+1
430 C=C-40:CO=PEEK(C):IFCO<17THEN460
440 POKEC+40,17:IFC<1344THENL=0:RETURN
450 POKEC,14:RETURN
460 IFCO<>0ANDCO<>1THEN540
470 POKE54276,129:POKE54276,128
480 FORT=-1TO1:FORI=-1TO1:C1=PEEK(C+I+T*4
    0)
490 IFC1=1THENRR=99:LM=0
500 IFC1>9ANDC1<14THENM1=1
510 L=0:POKEC+I+T*40,(RND(1)*2)+18:NEXT:N
    EXT:IFM1=1THEN760
520 FORT=-1TO1:FORI=-1TO1:POKEC+I+T*40,17
    :NEXT:NEXT:RETURN
530 :
540 PRINT"{HOME}{PUR}{7 DOWN}"MID$(A$(0),
    H,40):POKEC+40,17:POKEC-40,32
550 POKE54276,129:POKE54276,128
560 FORT=1TO100:NEXT
570 PRINT"{WHT}{HOME}{7 DOWN}{40 SPACES}"
    ;
580 M=0:L=0:IFCO<7THENBS=BS+1
590 IFCO>6THENMS=MS+1
600 IFLM<RRTHENRR=99
610 IFMS+BS=5THENPOKE53281,7:V=20
620 IFMS+BS=15THENPOKE53281,14:V=10
630 IFMS+BS=30THENPOKE53281,10
640 IF(MS+BS)/5=INT((MS+BS)/5)THENFORT=1T
    O8:POKE1384+(RND(1)*359),0:NEXT:J=1
650 IFJ=1THENFORT=-1TO2:FORI=-1TO1:POKEA+
    T+I*40,17:NEXT:NEXT:J=0
660 RETURN
670 :
680 RR=99:ML=0:POKE54283,129:POKE54283,12
    8
690 FORT=-1TO1:FORI=-1TO1:C1=PEEK(N1+I+T*
    40)
```

Cwww.commodore.ca

```
700 IFC1=14THENL=0
710 IFC1>9ANDC1<14THENM1=1
720 POKEN1+I+T*40,(RND(1)*2)+18:NEXT:NEXT
    :IFM1=1THEN760
730 FORT=-1TO1:FORI=-1TO1:POKEN1+I+T*40,1
    7:NEXT:NEXT:GOTO300
740 :
750 REM SUB DEATH
760 POKEA,15:POKEA+1,16:POKE54286,0:POKE5
    4290,129
770 FORT=230TO255STEP.15:POKE54287,255-T:
    POKE53281,15ANDT:NEXT:POKE54290,128
780 FORT=1TO50:POKEA,(RND(1)*2)+18:POKEA+
    1,(RND(1)*2)+18
790 NEXT:POKE53272,21:POKE53280,0:POKE532
    81,5
800 PRINT"{CLR}{BLK}{3 DOWN}"SPC(9)"YOU S
    ANK"BS"CARRIER";CHR$(-83*(BS<>1))
810 PRINT SPC(10)"{2 DOWN}AND"MS"DESTROYE
    R";CHR$(-83*(MS<>1))
820 PRINTSPC(6)"{5 DOWN}PRESS TRIGGER TO
    {SPACE}PLAY AGAIN":I=0
830 IF(PEEK(56320)AND16)=0THENRUN890
840 I=I+1:IFI>1000THENPRINT"{CLR}";:END
850 GOTO830
860 :
870 REM INIT
880 GOSUB1060
890 PRINT"{CLR}":POKE53280,0:POKE53281,1
900 POKE53272,28:POKE54296,15:ZZ=80000
910 X=10:V=30:AX=360:AY=20:RR=30:H=2
920 PRINTSPC(13)"{BLK}{RVS}{2 SPACES}U-BO
    AT{2 SPACES}26{2 SPACES}"
930 A$(1)="{39 SPACES}BDE{3 SPACES}"
940 A$(2)="{39 SPACES}FG{3 SPACES}"
950 A$(3)="{43 SPACES}BCDE "
960 A$(4)="{43 SPACES}HI "
970 A$(0)="{40 SPACES}OPOP "
980 FORT=1TO4:PRINT"{YEL}QQQQQQQQQQQQQQQQ
    QQQQQQQQQQQQQQQQQQQQQQQQ";:NEXT
990 PRINT"{DOWN}"
1000 FORT=1TO12:PRINT"{BLU}QQQQQQQQQQQQQQ
     QQQQQQQQQQQQQQQQQQQQQQ";:NEXT
1010 FORT=1824TO2023:POKET,17:POKE54272+T
     ,0:NEXT
1020 FORT=1TO15:POKE1384+RND(1)*359,0:NEXT
1030 POKE1794,17:POKE1795,17:GOTO120
1040 :
1050 REM CHARSET
1060 POKE53280,0:POKE53281,4:PRINT"{CLR}
     {6 DOWN}{BLU}"SPC(13)"{RVS}
     {2 SPACES}U-BOAT 26{2 SPACES}"
1070 PRINTSPC(10)"{3 DOWN}ONE MOMENT PL
     EASE"
1080 FORI=0TO58:READA:POKE49152+I,A:NEXT:
     SYS49152
1090 READT:IFT=-1THE?N1130
1100 POKE12288+Q,T:Q=Q+1:GOTO1090
1110 :
1120 REM INSTRUCTIONS
1130 PRINT"{CLR}{3 DOWN} FIRE TORPEDOES F
     ROM THE SUB'S FRONT"
1140 PRINT"{3 SPACES}TO SINK SHIPS AND DE
     STROY MINES."
1150 PRINT"{3 DOWN} BEWARE OF THE DESTRUC
     TIVE AREA WHEN"
1160 PRINT"{4 SPACES}MINES AND DEPTH CHAR
     GES EXPLODE."
1170 PRINT"{3 DOWN}{3 SPACES}DESTROY AS M
     ANY SHIPS AS YOU CAN"
1180 PRINT" BEFORE YOU ARE PERMANENTLY RE
     TIRED."
1190 PRINTSPC(12)"{4 DOWN}{RVS} PRESS
     {2 SPACES}TRIGGER ":POKE198,0
1200 IFPEEK(56320)AND16THEN1200
1210 RETURN
1220 :
1230 DATA 173,14,220,41,254,141,14,220,16
     5,1,41,251,133,1,169
1240 DATA 48,133,252,169,208,133,254,169,
     0,133,251,133,253
1250 DATA 162,8,177,253,145,251,200,208,2
     49,230,254,230,252
1260 DATA 202,208,242,165,1,9,4,133,1,173
     ,14,220,9,1,141,14,220,96
1270 :
1280 DATA 182,193,213,128,201,193,182,255
1290 DATA 255,159,159,159,243,243,243,255
1300 DATA 0,0,0,0,255,101,63,31
1310 DATA 0,96,112,252,255,148,255,255
1320 DATA 0,14,30,127,255,145,255,255
1330 DATA 0,0,0,0,255,166,252,248
1340 DATA 0,0,4,4,15,10,127,63
1350 DATA 0,192,196,200,208,224,255,254
1360 DATA 0,3,35,19,11,7,255,127
1370 DATA 0,0,32,32,240,80,254,252
1380 DATA 254,254,252,252,0,133,208,224
1390 DATA 255,255,127,63,0,80,3,31
1400 DATA 255,255,254,254,0,10,192,248
1410 DATA 127,127,63,63,0,161,11,7
1420 DATA 255,239,239,239,239,199,215,255
1430 DATA 65,41,68,175,122,189,110,219
1440 DATA 17,74,148,125,218,180,221,182
1450 DATA 255,255,255,255,255,255,255,255
1460 DATA 191,213,187,92,171,237,190,219
1470 DATA 247,174,181,125,235,189,173,222
1480 DATA-1
```

©

www.commodore.ca

NEWS&PRODUCTS

Singing Speech Synthesizer

The Alien Group has made its Voicebox Speech Synthesizer available for the VIC and Commodore 64.

The Voicebox plugs directly into the user port, includes its own speaker, volume and pitch controls, and has an unlimited vocabulary potential.

Programs supplied with the Voicebox include: a machine language text-to-speech program that can be merged with BASIC programs; a text-to-speech program that incorporates the "Alien" face, whose mouth moves in sync with what is being said; a program demonstrating the Voicebox's ability to sing; and a spelling quiz program using the talking face as the quizmaster.

The Voicebox is available for the VIC and 64 for $95. For an

The Voicebox Speech Synthesizer plugs directly into the VIC-20 or Commodore 64 user port.

additional $25, Commodore 64 users can add a comprehensive music and singing system. The capabilities of the 64's SID chip are incorporated into this program, which also allows the user to enter melodies through the keyboard.

*The Alien Group
27 West 23rd St.
New York, NY 10010*

Four Programs For Atari
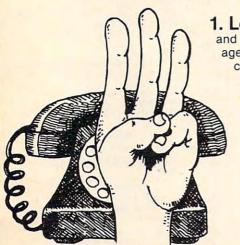
Advanced Computing Enterprises has added four programs to its Atari product list. They include an article indexer, a portfolio management system, a spelling game, and a math game.

MAX is the magazine article index program. It allows the user to code articles with keywords, and it performs sorts and searches in seconds. *MAX* is available on disk for $29.95.

Ace Money Machine makes use of a dollar cost averaging formula to help you manage your investment portfolio. The program maintains a complete history of the progress of your investments, and can provide a detailed graph upon request. The program sells for $19.95.

Word Wizard is a spelling tutorial game that allows the

www.commodore.ca

entry of word lists of up to 50 words. The program adjusts skill levels in relation to each student's progress. *Word Wizard* sells for $29.95.

Math Master is similar to *Word Wizard* in that it allows a parent or teacher to determine the problems to be solved and adjusts speed according to each student's success. The program is available for $29.95.

Advanced Computing Enterprises
5516 Rosehill Road
Shawnee Mission, KS 66216

TI-99/4A Peripherals And Software

Texas Instruments has changed its marketing strategy for TI-99/4A Home Computer peripherals. Prices for peripheral equipment and some software have been cut, and TI has begun selling its disk-storage related peripherals as a package.

The Disk Drive Memory System, which now has a suggested retail price of $550, consists of the peripheral expansion box, the disk controller card, one disk drive, and a 32K memory expansion card. Under TI's old pricing system, the suggested retail price for the above combination of equipment was $1,200.

Other reductions include: the P-Code Card, cut from $250 to $100; the RS-232 card, reduced from $175 to $100; the telephone coupler, from $200 to $100; the Impact Printer, from $750 to $500; and TI Logo II, from $129.95 to $99.95.

In other news from TI, the company has signed manufacturing and marketing agreements for software from Brøderbund Software, Spinnaker Software, and Sega Enterprises. Under the agreements, the software makers will supply programs to TI, and TI will translate them to ROM cartridge and market them.

As a result of the agreements, TI's software library soon will include Brøderbund's *Choplifter*, a daredevil rescue game, and *David's Midnight Magic*, a high-resolution pinball game simulation.

Spinnaker's first offerings to TI are *Facemaker*, an educational game for 4- to 12-year-olds, and *Story Machine*, a program that teaches children to write sentences, paragraphs, and simple stories, and then brings the stories to life through color graphics and sound.

Sega, a recognized leader in coin-operated amusement games, will provide TI with *Congo Bongo*, a cartoon-like adventure that takes players over and through jungle obstacles; *Star Trek*, in which the player controls the starship Enterprise against the Klingons; and *Buck Rogers: Planet of Zoom*, a space adventure filled with challenging tunnels and channels to negotiate and bizarre aliens to outsmart.

Most of the above software is expected to sell for $39.95 in cartridge format.

Texas Instruments, Inc.
P.O. Box 53
Lubbock, TX 79408
(800)858-4565

Baseball For Atari, Commodore 64

Star League Baseball, an action and strategy game from Gamestar, is available for Atari and Commodore 64.

The game features realistic animation including windup, delivery, and the arc and shadow of a fly ball. Sounds generated by the program include the crack of the bat, the cheers of the crowd, and the music of the seventh inning stretch.

Players choose their starting team and pitcher, can bring in relievers, and can take batting practice against the legendary "Heat" Muldoon.

Star League Baseball, which can be played against the computer or a human opponent, sells for $31.95.

Gamestar, Inc.
1302 State St.
Santa Barbara, CA 93101
(805) 963-3487

Investment Manager, Disk Manager, And Games

Bytes and Bits has released a handful of programs for the Commodore 64 and VIC-20 computers.

Investment Portfolio Manager is a program to track volatile assets such as stocks and stock options.

The program, for disk-based Commodore 64 systems, can handle entries of up to $99,999 and can report on nine investment categories. It is available for $14.95.

Disk Directory Manager, which can sort more than 1500 filenames into an organized list, is available for $19.95 for the VIC or 64. The program is written in machine language.

In *Dungeons* for the VIC-20 with 16K expansion, the player creates characters who explore a 12-level, 1200-room dungeon. The game is available on tape or disk for $19.95.

Pak Alien for the unexpanded VIC is a machine language game that includes 100 difficulty levels. Guide your alien through a maze of interplanetary space particles while dodging seven aliens. *Pak Alien*, which can be played with joystick or keyboard, sells for $14.95 on tape or disk.

Bytes and Bits
524 East Canterbury Lane
Phoenix, AZ 85022
(602) 942-1475
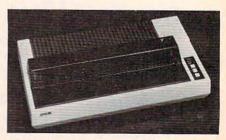
High Speed Printer

Epson America has introduced its new FX-100 dot matrix printer. The printer, which carries a suggested price of $895, provides a printing speed of 160 characters per second.

The printer offers a wide range of features including elite or pica spacing and a one-to-one graphics ratio so accurate graphics – including circles – can be drawn. Users also can create their own character formats on screen and download the font into the printer's memory.

The printer includes a rubber platen that can handle single sheet or roller-fed paper plus a removable tractor to handle pin-fed paper and forms.

Epson America, Inc.
3415 Kashiwa St.
Torrance, CA 90505
(213) 378-2220

The Epson FX-100 printer has a printing speed of 160 characters per second.

Cwww.commodore.ca

Arcade-Quality Apple

Third Millennium Engineering has introduced the Arcade Board, which generates arcade-quality color graphics and sound effects on Apple II, II+, and IIe computers.

The board uses many of the same video and sound chips used in coin-operated video-games.

The Arcade Board comes with more than 20 program demos, games, and utilities, and it adds more than 40 commands to Applesoft BASIC. These commands make it possible to create action with the convenience of BASIC, but with the speed of machine language. The Arcade Board sells for $225.

Third Millennium Engineering Corp.
1015 Gayley Ave., Suite 394
Los Angeles, CA 90024
(213) 473-2102

Speech In Many Voices

Don't Ask Software has produced a version of its *Software Automatic Mouth* (S.A.M.) for the Commodore 64 computer. In addition, the Apple and Atari versions of the program have been enhanced to include KNOBS, a feature that provides a range of different voices.

S.A.M., which occupies 2K of user memory, will convert phonetic spellings to speech. A text-to-speech Reciter program, available on the *S.A.M.* disk, will convert English spelling to speech; it takes 8K of user RAM. KNOBS gives the program access to a variety of voices, so characters can be made to converse.

The Commodore 64 version of the program and the updated Atari version sell for $59.95. The Apple II version, which contains a digital-to-analog converter and audio amplifier, sells for $124.95. Present owners of the program can receive the updated versions for a small fee, provided they have sent in their owner registration cards.

Other new products from Don't Ask include *Chatterbee*, a spelling game that includes a version of the *S.A.M.* voice synthesizer. *Chatterbee* is available for Atari and Commodore 64 for $49.95.

Wordrace for Commodore 64 is a vocabulary game with 2600 words. It is available in a game-pack that includes Claim to Fame, a history game, and Sports Derby, a sports fact game. The *Wordrace* gamepack is available on disk or cassette for $34.95.

Add $2 for shipping and handling for Don't Ask products.

Don't Ask Computer Software
2265 Westwood Blvd., Suite B-150
Los Angeles, CA 99064
(213) 477-4514

www.commodore.ca

Cwww.commodore.ca

Holiday Adventures

Chartscan Data is producing *bitCards*, text and graphic adventures designed as holiday gifts. *bitCards* are customized with personal references to the recipient, and the player's reward for completing the adventure is a personal message chosen by the sender.

The first *bitCard*, *A Christmas Adventure*, is available on disk for 48K computers in the Apple II series, and on tape for Atari 400/800, Radio Shack Color Computer, Commodore 64, unexpanded VIC, and VIC with 8K expansion. All versions sell for $16.95.

bitCards
120 South University Drive, Suite F
Plantation, FL 33317
(305) 473-4741

Games Converted For Atari

Activision has converted two of its best-selling home videogames for Atari home computers.

The games, *River Raid* and *Kaboom!*, have been enhanced to make use of the increased power and memory availability of the Atari computers.

In *River Raid* the player pilots a B1 Strato-wing Assault Jet over a constantly changing river course. New twists added to the game include hovering hot-air reconnaissance balloons, helicopters and tanks that shoot back, vivid graphics and battle sounds, an expanded control panel, a pause feature, and the ability for more advanced players to select more difficult river sections.

Kaboom! features the Mad Bomber, a convict who roams the rooftops dropping bombs. The player maneuvers water buckets to catch the falling bombs. New additions to the game include a variation that allows one player to be the Mad Bomber and the other to catch the falling bombs, a high-score table, and a musical score which features the *1812 Overture*.

Both games retail for $34.95.

Activision, Inc.
2350 Bayshore Frontage Road
Mountain View, CA 94043
(415) 960-0410

Software Assortment For Timex/Sinclair

Dynacomp has introduced an assortment of software products for the 16K Timex/Sinclair computer.

The new programs are: *Blackjack*, $12.95; *Checkbook*, $14.95; *Data Filer*, $19.95; *Graphics Drawer*, $14.95; *Phone Book*, $12.95; *Pixel Drawer*, $14.95; *Tic-Tac-Toe*, $9.95; and *Word Scramble*, $9.95.

Each of the programs is described in the free product catalog available from Dynacomp.

Dynacomp, Inc.
1427 Monroe Ave.
Rochester, NY 14618

Game Cartridges For The TI

Funware, which recently was acquired by Creative Software, has added to its list of available cartridge software for the TI-99/4A.

Among the new games are: *St. Nick*, in which the player helps Santa fly through a maze and pick up toys while avoiding flying witches.

Schnoz-ola, a game based on a tale about a Mayan prince with tremendous jumping powers and a tremendous nose. The prince must climb a four-tiered pyramid to collect sacred flowers, while dodging fireballs tossed at him by angry gods.

C www.commodore.ca

Ambulance, in which the player serves as an ambulance driver helping sick people get treatment. The player must decide who needs help first and whether the patient should be taken to the hospital or an emergency center.

Driving Demon, a game in which the player tries to drive as far and as fast as possible within a time limit. Problems include oil slicks, other cars, and engine troubles; navigational aids include a tachometer, speedometer, and a long-range scanner.

Funware's cartridge games sell for $44.95, and are guaranteed to work in all present and future versions of the TI-99/4A home computer.

Funware, Inc.
230 East Caribbean Drive
Sunnyvale, CA 94089
(408) 745-1655

Pinball Simulation For The 64

Bróderbund Software has converted *David's Midnight Magic*, a pinball simulation game, into Commodore 64 format.

The game, described as the next best thing to a real pinball machine, simulates dual flipper controls, bumper action, rollovers, sounds, and lights. The player can even put English on the ball, but too much will result in a tilt.

David's Midnight Magic, originally sold in Apple and Atari versions, is available for the Commodore 64 on disk for $34.95.

Bróderbund Software
500 Fifth Ave.
New York, NY 10110

Apple Graphics Program

T & G is an Apple graphics system from C & C Software. The program does not rely on shape tables, so execution is faster than for many other graphics programs.

T & G includes three character sets, the largest of which is 1½ times the size of standard Apple characters.

The characters can be placed on the screen using one of two commands. One is similar to BASIC's PRINT command; the other permits greater control over text placement. Lowercase is available in the two larger character sets, and the ability to print superscripts, subscripts, and text overlays is provided.

The program also is a powerful utility for producing static or animated graphics. The editor program supports full color and includes a command for clearing a precisely defined area of the screen.

T & G is available for $65.

C & C Software
5713 Kentford Circle
Wichita, KS 67220
(316) 683-6056

Hardware/Software Combo For Atari 800

MEM/EX is a memory expansion device for the Atari 800 computer. It replaces the operating system board and includes 4K of low-power CMOS static RAM mapped to the unused $C000 address space.

Included with the board is a powerful command-extender program called *COM/EX*. This adds ten edit-mode commands to the standard BASIC or Assembler cartridges, including the most used disk commands.

MEM/EX is compatible with all standard Atari 800 configurations; COM/EX requires at least one disk drive and DOS 2.0. The package is available for $129.95.

Prairie Physics
P.O. Box 2369
Wichita, KS 67201

C▪www.commodore.ca

Games For The VIC And 64

Creative Software has produced *In the Chips*, an educational game for the VIC-20.

In the game, one or two players are given $100,000 with which to organize a software company, research and develop a product line, and plan marketing strategies. The winner will be the one who assesses the market correctly and turns the largest profit.

The game, which is available in a cartridge, sells for $29.95.

Another new offering from Creative is *Crisis Mountain*, previously available for Apple and Atari. The game, licensed to Creative from Synergistic Software, is an adventure in which the player must guide mine explorer Kip Armstrong to treasure while avoiding traps, bombs, and bats.

Crisis Mountain, sold in cartridge format, is available for $34.95 for the 64, and $29.95 for the VIC.

Creative Software
230 East Caribbean Drive
Sunnyvale, CA 94089
(408) 745-1655

Touch-Sensitive Input Device

The Chalk Board PowerPad is an easy-to-use device that replaces the keyboard as a means of computer input.

The PowerPad has a 12-inch by 12-inch active surface that can handle input from multiple points.

The device, which can be used with the Commodore 64, VIC-20, Atari, Apple, and IBM computers, sells for $99.95.

Selections from "Leonardo's Library" are designed to be used with the PowerPad. Programs presently available include *Leonardo's Logo*, *Leo's 'Lectric Paintbrush*, and *Micro Maestro*. Programs from the library, which is expected to grow to as many as 30 selections, range in price from $24.95 to $49.95.

Miller Communications
607 Boylston St.
Copley Square
Boston, MA 02116
(617) 536-0470

Cwww.commodore.ca

CAPUTE!

Modifications Or Corrections To Previous Articles

Atari Caves Of Ice

Due to a problem with variable initialization, the Atari version of this game from the September issue (p. 60) provides more exits the longer the game is played. To insure there is always only one exit from the Caves, reader Jim Vail suggests that the A$(750) = "O" in line 100 be changed to:

```
A$="O":A$(750)="O":A$(2)=A$:
```

Also, Ohio reader Neil Morris notes that the program can be modified to run in less than 16K of memory by changing the GRAPHICS 8 in line 305 to GRAPHICS 6, then dividing all the numbers after the PLOT and DRAWTO commands in lines 32-85 by two (dropping any fractions).

VIC Diamond Drop

Some Commodore 64 color controls slipped into the VIC version of this game from the September issue (p. 84). For proper operation, change the {6} in line 50 to {GRN} (CTRL-6 for green) and the {7} in lines 100 and 102 to {BLU} (CTRL-7 for blue).

Mystery Spell For TI And VIC

The TI version of this educational game from September (p. 117) stores word lists properly to disk. However, tape users need to make the following changes:

```
1810 DISPLAY AT(23,1):"PLACE TAPE O
     R DISK IN DEVICE" :: ACCEPT AT
     (11,3):F$
1815 IF SEG$(F$,1,1)="C" THEN 1842
     ELSE OPEN #1:F$,INTERNAL,UPDAT
     E,FIXED 50
1842 IF J=0 THEN 1846
1844 OPEN #1:F$,OUTPUT,INTERNAL,FIX
     ED 50 :: FOR I=1 TO 20 :: PRIN
     T #1:B$(I):: NEXT I :: CLOSE #
     1 :: GOTO 230
1846 OPEN #1:F$,INPUT ,INTERNAL,FIX
     ED 50 :: FOR I=1 TO 20 :: INPU
     T #1:B$(I):: NEXT I :: CLOSE #
     1 :: CALL CLEAR :: GOTO 1760
```

Also, there is a minor typo in the VIC version (p. 127). In line 2170, LEFRT$ should be LEFT$.

VIC PILOT

There is a minor flaw in the turtle graphics feature of the PILOT interpreter in BASIC from the September issue (p. 166). If the turtle goes off the graphics screen during a loop, subsequent graphics commands are not interpreted properly. If you are using Program 1, make this change:

```
128 IFOS%=1THENPRINT"*PLOT WENT OFF SCREE
    N":D=0
```

If you are using Program 2, the Super Expander version, make this change:

```
143 GOSUB127:IFOS%=1THEND=0:GOTO11
```

Cracking The Kernal

The instructions for using Kernal ROM routines to LOAD data from tape or disk into the 64 (September 1983, p. 270) are incorrect. The proper series of machine language instructions to OPEN logical file 2 from device 8 (disk) with secondary address 0 and LOAD a file into memory starting at address 8192 ($2000) is:

```
        LDA     #$02
        LDX     #$08
        LDY     #$00
        JSR     SETLFS
        LDA     #$04
        LDX     #L,NAME
        LDY     #H,NAME
        JSR     SETNAM
        LDA     #$00
        LDX     #$00
        LDY     #$20
        JSR     LOAD
        RTS
NAME .BY     'FILE'
```

Note that the accumulator must be set to 0 for a LOAD and 1 for a VERIFY, not vice versa as stated in the article. Our thanks to Lewis Kleinsmith for pointing this out.

Atari Dragon

Program 2 (August 1983, p. 88) fails to leave the current score on the screen at the end of the game. To eliminate this minor flaw, Stuart Goldenberg suggests the following correction and addition:

```
410 COLOR 32:PLOT 0,0:DRAWTO 19,0:PL
    OT 0,1:DRAWTO 10,1:POSITION 0,0:
    ? #6;"GAME OVER score ";W
435 POSITION 0,0:? #6;"{20 SPACES}"
```

TI Towers

The program for this article was renumbered before it was listed. As a result, the line numbers of the program do not agree with those mentioned in the article. However, the program will work as listed.

64 Ultrasort

When using Program 4 to test the 64 version of this machine language sorting routine (September 1983, p. 202), you should replace line 300 with:

```
300 SYS 49152,N,AA$(1)
```

The REM statement in line 291 is incorrect.

VIC Dots

In line 560 of this program from the September issue (p. 132), the spurious question mark should be removed. The first part of the line should read:

```
560 IFFNBX(ML+22) = ...
```

COMPUTE! Back Issues

Here are some of the applications, tutorials, and games from available back issues of **COMPUTE!**. Each issue contains much, much more than there's space here to list, but here are some highlights:

May 1981: Named GOSUB/GOTO in Applesoft, Generating Lower Case Text on Apple II, Copy Atari Screens to the Printer, Disk Directory Printer for Atari, Realtime Clock on Atari, PET BASIC Delete Utility, PET Calculated Bar Graphs, Running 40 Column Programs on a CBM 8032.

June 1981: Computer Using Educators (CUE) on Software Pricing, Apple II Hires Character Generator, Ever- expanding Apple Power, Color Burst for Atari, Mixing Atari Graphics Modes 0 and 8, Relocating PET BASIC Programs, An Assembler In BASIC for PET, QuadraPET: Multitasking?

July 1981: Home Heating and Cooling, Animating Integer BASIC Lores Graphics, The Apple Hires Shape Writer, Adding a Voice Track to Atari Programs, Machine Language Atari Joystick Driver, Four Screen Utilities for the PET, Saving Machine Language Programs on PET Tape Headers, Commodore ROM Systems, The Voracious Butterfly on OSI.

August 1981: Minimize Code and Maximize Speed, Apple Disk Motor Control, A Cassette Tape Monitor for the Apple, Easy Reading of the Atari Joystick, Blockade Game for the Atari, Atari Sound Utility, The CBM "Fat 40," Keyword for PET, CBM/PET Loading, Chaining, and Overlaying.

October 1981: Automatic DATA Statements for CBM and Atari, VIC News, Undeletable Lines on Apple, PET, VIC, Budgeting on the Apple, Switching Cleanly from Text to Graphics on Apple, Atari Cassette Boot-tapes, Atari Variable Name Utility, Atari Program Library, Train your PET to Run VIC Programs, Interface a BSR Remote Control System to PET, A General Purpose BCD to Binary Routine, Converting to Fat-40 PET.

December 1981: Saving Fuel $$ (multiple computers: versions for Apple, PET, and Atari), Unscramble Game (multiple computers), Maze Generator (multiple computers), Animating Applesoft Graphics, A Simple Printer Interface for the Apple II, A Simple Atari Wordprocessor, Adding High Speed Vertical Positioning to Atari P/M Graphics, OSI Supercursor, A Look At SuperPET, Supermon for PET/CBM, PET Mine Maze Game.

January 1982: Invest (multiple computers), Developing a Business Algorithm (multiple computers), Apple Addresses, Lowercase with Unmodified Apple, Cryptogram Game for Atari, Superfont: Design Special Character Sets on Atari, PET Repairs for the Amateur, Micromon for PET, Self-modifying Programs in PET BASIC, Tinymon: a VIC Monitor, Vic Color Tips, VIC Memory Map, ZAP: A VIC Game.

May 1982: VIC Meteor Maze Game, Atari Disk Drive Speed Check, Modifying Apple's Floating Point BASIC, Fast Sort For PET/CBM, Extra Atari Colors Through Artifacting, Life Insurance Estimator (multiple computers), PET Screen Input, Getting The Most Out Of VIC's 5000 Bytes.

August 1982: The New Wave Of Personal Computers, Household Budget Manager (multiple computers), Word Games (multiple computers), Color Computer Home Energy Monitor, Intelligent Apple Filing Cabinet, Guess That Animal (multiple computers), PET/CBM Inner BASIC, VIC Communications, Keyprint Compendium, Animation With Atari, VIC Curiosities, Atari Substring Search, PET and VIC Electric Eraser.

September 1982: Apple and Atari and the Sounds of *TRON*, Commodore Automatic Disk Boot, VIC Joysticks, Three Atari GTIA Articles, Color Computer Graphics, The Apple Pilot Language, Sprites and Sound on the Commodore 64, Peripheral Vision Exerciser (multiple computers), Banish INPUT Statements (multiple computers), Charades (multiple computers), PET Pointer Sort, VIC Pause, Mapping Machine Language, Editing Atari BASIC With the Assembler Cartridge, Process Any Apple Disk File.

January 1983: Sound Synthesis And The Personal Computer, Juggler And Thunderbird Games (multiple computers), Music And Sound Programs (multiple computers), Writing Transportable BASIC, Home Energy Calculator (multiple computers), All About Commodore WAIT, Supermon64, Perfect Commodore INPUTs, Atari Autonumber, Copy VIC Disk Files, Commodore 64 Architecture.

March 1983: An Introduction To Data Storage (multiple computers), Mass Memory Now And In The Future, Games: Closeout, Boggler, Fighter Aces, Letter And Number Play (all for multiple computers), VIC Music, Direct Atari Disk Access, TRS-80 Color Computer Data Base, Apple Subroutine

Capture, PET Quickplot, TI Graphics Made Easy, VIC and Atari Memory Management.

April 1983: Selecting The Right Word Processor, Air Defense (multiple computers), Scriptor: An Atari Word Processor, Retirement Planner (multiple computers), TI-99 Match-Em, Dr. Video For Commodore, Atari Filefixer, Video 80: 80 Columns For The Atari, VICword, Magic Commodore BASIC.

May 1983: The New Low Cost Printer/Plotters, Jumping Jack (multiple computers), Deflector (multiple computers), VIC Kaleidoscope, Graphics on the Sinclair/Timex, Bootmaker For VIC, PET and 64, VICSTATION: A "Paperless Office," The Atari Musician, Apple Fast Sort, TI BASIC One-liners.

June 1983: How To Buy The Right Printer, The New, Low-cost Printers, Astrostorm (multiple computers), The Hawkmen Of Dindrin (multiple computers), MusicMaster For The Commodore 64, Commodore Data Searcher, Atari Player/Missile Graphics Simplified, TI Structured BASIC, UnNEW For The VIC and 64, Atari Fast Shuffle, VIC Contractor, Sinclair/Timex Screen Splitter.

Home and Educational COMPUTING!
(Fall 1981 and Summer 1981 – count as one back issue): Exploring The Rainbow Machine, VIC As Super Calculator, Custom Characters, Alternate Screens, Automatic Line Numbers, Using The Joystick (Spacewar Game), Fast Tape Locater, Window, VIC Memory Map.

Back issues are $3 each or six for $15. Price includes freight in the US. Outside the US add $1 per magazine ordered for surface postage. $4 per magazine for air mail postage. All back issues subject to availability.

In the Continental US call
TOLL FREE 800-334-0868
(In NC Call 919-275-9809)

Or write to **COMPUTE!** Back Issues, P.O. Box 5406, Greensboro, NC 27403 USA. Prepayment required in US funds. MasterCard, Visa and American Express accepted. North Carolina Residents add 4% sales tax.

A Beginner's Guide To Typing In Programs

What Is A Program?

A computer cannot perform any task by itself. Like a car without gas, a computer has *potential*, but without a program, it isn't going anywhere. Most of the programs published in **COMPUTE!** are written in a computer language called BASIC. BASIC is easy to learn and is built into most computers (on some computers, you have to purchase an optional BASIC cartridge).

BASIC Programs

Each month, **COMPUTE!** publishes programs for many machines. To start out, type in only programs written for your machine, e.g., "TI Version" if you have a TI-99/4. Later, when you gain experience with your computer's BASIC, you can try typing in and converting certain programs from one computer to yours.

Computers can be picky. Unlike the English language, which is full of ambiguities, BASIC usually has only one "right way" of stating something. Every letter, character, or number is significant. A common mistake is substituting a letter such as "O" for the numeral "0", a lowercase "l" for the numeral "1", or an uppercase "B" for the numeral "8". Also, you must enter all punctuation such as colons and commas just as they appear in the magazine. Spacing can be important. To be safe, type in the listings *exactly* as they appear.

Brackets And Special Characters

The exception to this typing rule is when you see the curved bracket, such as "{DOWN}". Anything within a set of brackets is a special character or characters that cannot easily be listed on a printer. When you come across such a special statement, refer to the appropriate key for your computer. For example, if you have an Atari, refer to the "Atari" section in "How to Type COMPUTE!'s Programs."

About DATA Statements

Some programs contain a section or sections of DATA statements. These lines provide information needed by the program. Some DATA statements contain actual programs (called machine language); others contain graphics codes. These lines are especially sensitive to errors.

If a single number in any one DATA statement is mistyped, your machine could "lock up," or "crash." The keyboard, break key, and RESET (or STOP) keys may all seem "dead," and the screen may go blank. Don't panic – no damage is done. To regain control, you have to turn off your computer, then turn it back on. This will erase whatever program was in memory, so always SAVE a copy of your program before you RUN it. If your computer crashes, you can LOAD the program and look for your mistake.

Sometimes a mistyped DATA statement will cause an error message when the program is RUN. The error message may refer to the program line that READs the data. *The error is still in the DATA statements, though.*

Get To Know Your Machine

You should familiarize yourself with your computer before attempting to type in a program. Learn the statements you use to store and retrieve programs from tape or disk. You'll want to save a copy of your program, so that you won't have to type it in every time you want to use it. Learn to use your machine's editing functions. How do you change a line if you made a mistake? You can always retype the line, but you at least need to know how to backspace. Do you know how to enter inverse video, lowercase, and control characters? It's all explained in your computer's manuals.

A Quick Review

1) Type in the program a line at a time, in order. Press RETURN or ENTER at the end of each line. Use backspace or the back arrow to correct mistakes.

2) Check the line you've typed against the line in the magazine. You can check the entire program again if you get an error when you RUN the program.

3) Make sure you've entered statements in brackets as the appropriate control key (see "How To Type COMPUTE!'s Programs" elsewhere in the magazine.)

www.commodore.ca

How To Type COMPUTE!'s Programs

Many of the programs which are listed in **COMPUTE!** contain special control characters (cursor control, color keys, inverse video, etc.). To make it easy to tell exactly what to type when entering one of these programs into your computer, we have established the following listing conventions. There is a separate key for each computer. Refer to the appropriate tables when you come across an unusual symbol in a program listing. If you are unsure how to actually enter a control character, consult your computer's manuals.

Atari 400/800

Characters in inverse video will appear like: ▨INVERSE VIDEO▨
Enter these characters with the Atari logo key, {⅄}

When you see	Type	See	
{CLEAR}	ESC SHIFT <	▨	Clear Screen
{UP}	ESC CTRL –	↑	Cursor Up
{DOWN}	ESC CTRL =	↓	Cursor Down
{LEFT}	ESC CTRL +	←	Cursor Left
{RIGHT}	ESC CTRL *	→	Cursor Right
{BACK S}	ESC DELETE	◄	Backspace
{DELETE}	ESC CTRL DELETE	▨	Delete character
{INSERT}	ESC CTRL INSERT	▨	Insert character
{DEL LINE}	ESC SHIFT DELETE	▨	Delete line
{INS LINE}	ESC SHIFT INSERT	▨	Insert line
{TAB}	ESC TAB	►	TAB key
{CLR TAB}	ESC CTRL TAB	▨	Clear tab
{SET TAB}	ESC SHIFT TAB	▨	Set tab stop
{BELL}	ESC CTRL 2	▨	Ring buzzer
{ESC}	ESC ESC	▨	ESCape key

Graphics characters, such as CTRL-T, the ball character ● will appear as the "normal" letter enclosed in braces, e.g. {T}.

A series of identical control characters, such as 10 spaces, three cursor-lefts, or 20 CTRL-R's, will appear as {10 SPACES}, {3 LEFT}, {20 R}, etc. If the character in braces is in inverse video, that character or characters should be entered with the Atari logo key. For example, {◼} means to enter a reverse-field heart with CTRL-comma, {5◼} means to enter five inverse-video CTRL-U's.

Commodore PET/CBM/VIC/64

Generally, any PET/CBM/VIC/64 program listings will contain words within braces which spell out any special characters: {DOWN} would mean to press the cursor down key. {5 SPACES} would mean to press the space bar five times.

To indicate that a key should be *shifted* (hold down the SHIFT key while pressing the other key), the key would be underlined in our listings. For example, S would mean to type the S key while holding the shift key. If you find an underlined key enclosed in braces (e.g., {10 N}), you should type the key as many times as indicated (in our example, you would enter ten shifted N's). Some graphics characters are inaccessible from the keyboard on CBM Business models (32N, 8032).

For the VIC and 64, if a key is enclosed in special brackets, {< >}, you should hold down the *Commodore key* while pressing the key inside the special brackets. (The Commodore key is the key in the lower left corner of the keyboard.) Again, if the key is preceded by a number, you should press the key as many times as indicated.

Rarely, you'll see in a Commodore 64 program a solitary letter of the alphabet enclosed in braces. These characters can be entered by holding down the CTRL key while typing the letter in the braces. For example, {A} would indicate that you should press CTRL-A.

About the *quote mode*: you know that you can move the cursor around the screen with the CRSR keys. Sometimes a programmer will want to move the cursor under program control. That's why you see all the {LEFT}'s, {HOME}'s, and {BLU}'s in our programs. The only way the computer can tell the difference between direct and programmed cursor control is the quote mode.

Once you press the quote (the double quote, SHIFT-2), you are in the quote mode. If you type something and then try to change it by moving the cursor left, you'll only get a bunch of reverse-video lines. These are the symbols for cursor left. The only editing key that isn't programmable is the DEL key; you can still use DEL to back up and edit the line. Once you type another quote, you are out of quote mode.

You also go into quote mode when you INSerT spaces into a line. In any case, the easiest way to get out of quote mode is to just press RETURN. You'll then be out of quote mode and you can cursor up to the mistyped line and fix it.

Use the following tables when entering special characters:

When You Read:	Press:		See:	When You Read:	Press:		See:
{CLEAR}	SHIFT	CLR/HOME	▨	{GRN}	CTRL	6	▨
{HOME}		CLR/HOME	▨	{BLU}	CTRL	7	▨
{UP}	SHIFT	↕CRSR↕	▨	{YEL}	CTRL	8	▨
{DOWN}		↕CRSR↕	▨	{F1}	f1		▨
{LEFT}	SHIFT	←CRSR→	▨	{F2}	f2		▨
{RIGHT}		←CRSR→	▨	{F3}	f3		▨
{RVS}	CTRL	9	▨	{F4}	f4		▨
{OFF}	CTRL	0	▨	{F5}	f5		▨
{BLK}	CTRL	1	▨	{F6}	f6		▨
{WHT}	CTRL	2	▨	{F7}	f7		▨
{RED}	CTRL	3	▨	{F8}	f8		▨
{CYN}	CTRL	4	▨		←		◄
{PUR}	CTRL	5	▨		SHIFT		▨

All Commodore Machines

Clear Screen	{CLR}	Cursor Left	{LEFT}
Home Cursor	{HOME}	Insert Character	{INST}
Cursor Up	{UP}	Delete Character	{DEL}
Cursor Down	{DOWN}	Reverse Field On	{RVS}
Cursor Right	{RIGHT}	Reverse Field Off	{OFF}

Apple II / Apple II Plus

All programs are in Applesoft BASIC, unless otherwise stated. Control characters are printed as the "normal" character enclosed in brackets, such as {D} for CTRL-D. Hold down CTRL while pressing the control key. You will not see the special character on the screen.

Texas Instruments 99/4

The only special characters used are in PRINT statements to indicate where two or more spaces should be left between words. For example, ENERGY {10 SPACES} MANAGEMENT means that ten spaces should be left between the words ENERGY and MANAGEMENT. Do not type in the braces or the words 10 SPACES. Enter all programs with the ALPHA LOCK on (in the down position). Release the ALPHA LOCK to enter lowercase text.

www.commodore.ca

www.commodore.ca

Advertisers Index

www.commodore.ca

COMPUTE!

My Computer Is:

☐ PET ☐ Apple ☐ Atari ☐ OSI ☐ VIC-20 ☐ TI 99/4A ☐ Sinclair ZX-81
☐ Radio Shack Color Computer ☐ Other _____ ☐ Don't yet have one...

☐ $20.00 One Year US Subscription
☐ $36.00 Two Year US Subscription
☐ $54.00 Three Year US Subscription

(Readers outside of the US, please see our foreign readers subscription card or inquire for rates).

Name _____

Address _____

City _____ State _____ Zip _____

☐ Payment Enclosed ☐ VISA ☐ Bill me
☐ MasterCard ☐ American Express
Account No. _____ Expires _____ /

Your subscription will begin with the first available issue.
Please allow 4-6 weeks for delivery of first issue.

33B101

COMPUTE! Books

Quan.	Title	Price	S/H	Total	Quan.	Title	Price	S/H	Total
___	Machine Language for Beginners	$14.95	+ $2.00*	___	___	COMPUTE!'s First Book of Atari Graphics	$12.95	+ $2.00*	___
___	Home Energy Applications	14.95	+ 2.00*	___	___	Mapping The Atari	14.95	+ 2.00*	___
					___	Inside Atari DOS	19.95	+ 2.00*	___
___	COMPUTE!'s First Book of VIC	12.95	+ 2.00*	___	___	Programmers Reference Guide for TI-99/4A	14.95	+ 2.00*	___
___	COMPUTE!'s First Book of VIC Games	12.95	+ 2.00*	___	___	COMPUTE!'s First Book of TI Games	12.95	+ 2.00*	___
___	COMPUTE!'s First Book of 64	12.95	+ 2.00*	___	___	Every Kid's First Book of Robots and Computers	4.95	+ 1.00*	___
___	COMPUTE!'s First Book of Atari	12.95	+ 2.00*	___	___	The Beginner's Guide to Buying A Personal Computer	3.95	+ 1.00*	___
___	COMPUTE!'s Second Book of Atari	12.95	+ 2.00*	___					

For Fastest Service Call Our **TOLL FREE** US Order Line **800-334-0868.** In NC call 919 275-9809.
All orders must be prepaid (money order, check, or charge). All payments must be in US funds. NC residents add 4% sales tax.

☐ Payment enclosed Please charge my: ☐ VISA ☐ MasterCard ☐ American Express
Acc't No. _____ Expires _____

Name _____

Address _____

City _____ State _____ Zip _____

Country _____

Allow 4-5 weeks for delivery. For air mail outside US. *$5.00

11 12

COMPUTE!

Subscription rates outside the US:

☐ $25.00 Canada
☐ $38.00 Europe, Australia/Air Delivery
☐ $48.00 Middle East/Air Delivery
☐ $68.00 Elsewhere/Air Delivery
☐ $25.00 International Surface Mail (lengthy, unreliable delivery)

Name _____

Address _____

City _____ Postal Code _____

Country _____

Payment must accompany this card.
Payment in US Funds drawn on a US Bank; International Money Order; or charge card: ☐ VISA ☐ MasterCard ☐ American Express
Account No. _____ Expires _____ /
Your subscription will begin with the first available issue.
Please allow 6-12 weeks for delivery of first issue.

33B101

The Editor's Feedback:

Computer: ☐ Pet ☐ Apple ☐ Atari ☐ OSI ☐ VIC-20 ☐ TI 99/4A ☐ Sinclair ZX-81
☐ Radio Shack Color Computer ☐ Other _____ ☐ Don't yet have one...

Are you a COMPUTE! Subscriber? ☐ Yes ☐ No I would like to see:

More	Just Right	Fewer		More	Just Right	Fewer	
☐	☐	☐	Specific applications programs.	☐	☐	☐	Games.
				☐	☐	☐	Reviews of game software.
☐	☐	☐	BASIC programs.				
☐	☐	☐	Machine language programs.	☐	☐	☐	Reviews of business software.
☐	☐	☐	Tutorials.	☐	☐	☐	Reviews of educational software.
☐	☐	☐	Educational articles.				
☐	☐	☐	Detailed explanations of programs.	☐	☐	☐	Reviews of hardware.

What do you like best about COMPUTE! ?

What do you like least?

COMPUTE! Magazine
P.O. Box 914
Farmingdale, NY 11737

BUSINESS REPLY MAIL
FIRST CLASS PERMIT NO. 2312   GREENSBORO, NC

POSTAGE WILL BE PAID BY ADDRESSEE

COMPUTE! Magazine
P.O. Box 914
Farmingdale, NY 11737

COMPUTE! Magazine
Post Office Box 5406
Greensboro, NC 27403

BUSINESS REPLY MAIL
FIRST CLASS PERMIT NO. 2312   GREENSBORO, NC

POSTAGE WILL BE PAID BY ADDRESSEE

COMPUTE! Books
Post Office Box 5406
Greensboro, NC 27403

www.commodore.ca

TRUNKS FOR THE MEMORIES.

Introducing the most logical place to store Elephant Memory Systems® (or lesser brands of disks): The Trunk.

With its alphabetized library index, you can file or retrieve up to 60 disks, instantly.

The Trunk is made of durable molded plastic with a hinged, one-piece lid, to keep disks safe from dust, dirt, and other detriments which disks despise.

And, it's portable. Because the lid doubles as a carrying handle so your Elephant Memory Systems® disks can go anywhere you do.

There's a model for 5¼" and 8" floppies, as well as a cassette-and-game file and a special Atari® version.

So if you're looking for the best disk storage system on the market . . .

The Trunk is an open-and-shut case.

THE TRUNK. ENDORSED BY ELEPHANTS.

Elephant Memory Systems® Disks

A full line of top-quality floppies, in virtually every 5¼" and 8" model, for compatibility with virtually every computer on the market. Guaranteed to meet or exceed every industry standard, certified 100% error-free and problem-free, and to maintain its quality for at least 12 million passes (or over a life-time of heavy-duty use).

Marketed exclusively by Leading Edge Products, Inc., Information Systems and Supplies Division, 55 Providence Highway, Norwood, Massachusetts 02062. Call: toll-free 1-800-343-8413; or in Massachusetts call collect (617) 769-8150. Telex 951-624.

www.commodore.ca