```
        ,0,1,1,0,0,2,0,0,0,2,2
        ,0
XK 1040 DATA 0,1,0,0,1,1,1,0,0
        ,0,0,0,2,2,2,0,1,1,0,0
        ,1,1,1,0,0,2,0,0,0,2,0
        ,0
CE 1050 DATA 1,1,1,0,1,1,1,0,0
        ,1,0,0,2,2,2,0,1,1,0,0
        ,1,1,1,0,0,2,0,0,2,2,2
        ,0
ED 1060 DATA 1,1,0,0,1,1,1,0,0
        ,1,0,0,2,2,2,0,1,1,0,0
        ,0,1,1,0,0,2,0,0,0,2,2
        ,0
BX 1070 DATA 1,1,0,0,1,1,1,0,0
        ,1,0,0,2,2,2,0,1,1,0,0
        ,0,1,1,0,1,2,0,0,0,2,2
        ,0
```

Program 2: BASIC Relocator

```
EG 5 REM COPYRIGHT 1987 COMPUT
     E! PUBLICATIONS, INC.
     {5 SPACES}ALL RIGHTS RESE
     RVED.
JC 10 PRINT"{CLR}{4 SPACES}
      {RVS}BASIC RELOCATOR
      {OFF}{DOWN}"
CK 12 PRINT"{4 SPACES}COPYRIGH
      T 1987":PRINT"COMPUTE! P
      UBLICATIONS, INC."
QP 14 PRINT"{2 SPACES}ALL RIGH
      TS RESERVED."
GG 16 FOR X=1 TO 1000:NEXT
FR 20 SB=PEEK(43)+PEEK(44)*256
      :EB=PEEK(55)+PEEK(56)*25
      6
GR 30 PRINT"{DOWN}CURRENT STAR
      T:"SB"{LEFT},";:N=SB:GOS
      UB 2000:PRINTHX$:SB$=HX$
RP 40 PRINT"CURRENT{3 SPACES}E
      ND:"EB"{LEFT},";:N=EB:GO
      SUB 2000:PRINTHX$:EB$=HX
      $
FD 50 PRINT"{DOWN}NEW START? "
      SB$"{7 LEFT}";:INPUT N$:
      GOSUB 1000:NS=N
PC 55 PRINT"NEW{3 SPACES}END?
      {SPACE}"EB$"{7 LEFT}";:I
      NPUT N$:GOSUB 1000:NE=N
MA 60 IF NE<NS THEN PRINT"
      {DOWN}TRY AGAIN":GOTO 50
JS 70 PRINT"{DOWN}COMPLETED."
DK 80 PRINT"{3 DOWN}POKE43,"NS
      -INT(NS/256)*256"{LEFT}:
      POKE44,"INT(NS/256)"
      {LEFT}:";
HR 81 PRINT"POKE55,"NE-INT(NE/
      256)*256"{LEFT}:POKE56,"
      INT(NE/256)"{LEFT}:POKE"
      NS"{LEFT},0:NEW";
GC 90 POKE 631,145:POKE 632,14
      5:POKE 633,13:POKE 198,3
GG 999 END
PP 1000 N1$=LEFT$(N$,1):N$=MID
        $(N$,2)
HS 1010 IF N1$<>"$" THEN N=VAL
        (N1$+N$):RETURN
PS 1020 N=0:FOR N1=1 TO 4:N2=A
        SC(MID$(N$,N1,1))-55:N
        2=N2-7*(N2<10)
SR 1030 N=N+N2*16↑(4-N1):NEXT:
        RETURN
DB 2000 H$="0123456789ABCDEF":
        HX$="$"
CC 2010 FOR N1=3 TO 0 STEP -1:
        N2=16↑N1:N3=INT(N/N2):
        N=N-N3*N2
DB 2020 HX$=HX$+MID$(H$,N3+1,1
        ):NEXT:RETURN
```


SoftSprite

Richard Schramm

This compact machine language routine for Atari computers allows you to move playfield shapes rapidly anywhere on the screen using simple BASIC commands.

Eight-bit Atari computers—the 400, 800, XL, and XE models—have a variety of powerful graphics features. However, the only way to draw, move, or animate a playfield shape (any shape that's not a player/missile) from BASIC is with a series of PLOT or POKE statements, an extremely slow process. "SoftSprite" permits you to move such shapes quickly and easily, with simple BASIC commands.

SoftSprite can move, animate, or duplicate a playfield shape on any portion of the GRAPHICS 8 screen. The shapes may be any size, as long as the data for each shape does not exceed 250 bytes. SoftSprite also offers collision detection, and the only limit to the number of shapes is the amount of memory available. And since the SoftSprite machine language code is only 281 bytes long, you can use it in any BASIC program without sacrificing a lot of memory.

Type in the demonstration program and run it to see SoftSprite in action. The program contains extensive REMark statements to explain in detail how SoftSprite works; we'll refer to the code throughout this article.

Making A Shape Table

Before you call SoftSprite from BASIC, you must store two things in memory: the SoftSprite machine code itself and the shape data for the image you want to display. The simplest way to store the data is with loops that READ numbers from DATA statements and that POKE them into safe memory areas.

The shape data can be any size, depending on the image, up to a maximum of 256 bytes (see below). This data is organized into what we'll call a shape table, which is simply a list of numbers that defines the shape.

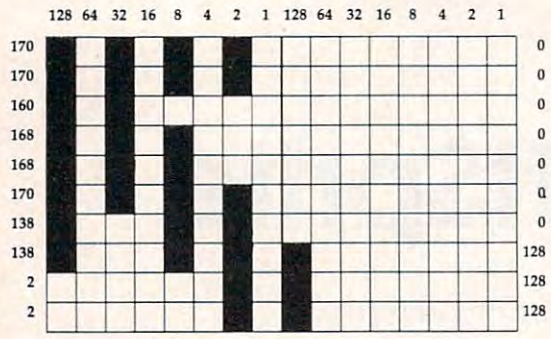
The figure illustrates how to construct a shape table. The shape we want to construct is an arrow like those shown in the demonstration program. By plotting the shape on graph paper, you easily can determine the numbers needed for its shape table.

This particular shape is ten dots high, so we draw a grid ten rows high. Each row of squares in the grid represents a row of dots in the arrow shape. The shape is nine dots wide. However, because it's easier for the program to deal with byte-length quantities, the shape table is 16 (2 * 8) columns wide. Thus, the grid for this shape is 10 dots high and 16 dots wide.

Within the grid, each group of eight horizontal squares represents a byte, or eight bits of memory. Each bit represents a power of 2, in ascending order from right to left. Thus, the rightmost bit has a value of 1, the next bit to the left has a value of 2, and so forth, up to the leftmost bit, which has a value of 128. The numbers at the top of the figure indicate the values assigned to each bit.

The next step is to fill in the squares where a dot will appear. Once this is done, you add up the bit values for each byte in the grid. The upper left byte, for instance, has a value of 170 (128 + 32 + 8 + 2 = 170). The upper right byte, however, has no darkened squares, so its value is zero. The lower right byte

Shape Table Grid



Shape Table For Arrow Image

0	reserved
0	reserved
0	reserved
0	reserved
10	height
2	width
170	row 1, column 1
0	row 1, column 2
170	row 2, column 1
0	row 2, column 2
160	row 3, column 1
0	row 3, column 2
168	row 4, column 1
0	row 4, column 2
168	row 5, column 1
0	row 5, column 2
170	row 6, column 1
0	row 6, column 2
138	row 7, column 1
0	row 7, column 2
138	row 8, column 1
128	row 8, column 2
2	row 9, column 1
128	row 9, column 2
2	row 10, column 1
128	row 10, column 2

has a value of 128, since only its leftmost bit is turned on, and so on.

Once you have written down the values for each byte in the grid, you are ready to construct the shape table. The first six bytes of the shape table are always reserved for SoftSprite's internal use. The first four of these bytes must be zeros, while the fifth and sixth bytes of the shape data represent the shape's height and width, respectively, in terms of bytes. In this case, the shape is 10 bytes high and 2 bytes wide, so the fifth and sixth bytes of its shape table are 10 and 2. The table illustrates the complete shape table for the arrow image.

The shape table can be a maximum of 256 bytes in length, including the six reserved bytes which begin the table. Thus, a shape can be a maximum of 250 bytes in size. You can calculate the size of the shape table by multiplying its height by its width and adding 6; the arrow shape table requires 26 (10 * 2 + 6) bytes.

Once the shape table is complete, you must find a safe memory area to store it. One common technique is to store the table in a string variable. The demonstration program shows how to do this. It first reserves the needed amount of memory with DIM, setting aside 278 bytes for the string SOFTSPRITE$ and 26 bytes for the string SHAPE$. The ADDR function lets us determine the addresses where the strings SOFTSPRITE$ and SHAPE$ begin. These values are assigned to the numeric variables SOFTSPRITE and SHAPE. (Since string variables and numeric variables are of different types, BASIC has no trouble distinguishing SOFTSPRITE from SOFTSPRITE$ and SHAPE from SHAPE$.)

After finding the addresses where the strings begin, we can use simple FOR-NEXT loops to READ the required values from DATA statements elsewhere in the program and to POKE them into the string space.

Calling SoftSprite From BASIC

The SoftSprite routine works in GRAPHICS 8, so the next step is to select that display mode. At this point, you are ready to display the shape with a USR statement. Here is the general syntax for calling SoftSprite with USR:

COL = USR(*ML address, x, y, shape address*)

The USR statement always takes four values inside parentheses. The first value is the address where the SoftSprite machine language routine begins. In the demonstration program, the variable SOFTSPRITE equals this address. The second and third parameters are the horizontal *(x)* and vertical *(y)* screen coordinates where you want the shape to appear. The fourth parameter is the memory address of the shape table (SHAPE in the demonstration program).

A simpler USR statement can be used to turn the shape off, once it is on the screen. In this case you can omit the coordinates, supplying only the address of the ML routine and the address of the shape data:

COL = USR(*ML address, shape address*)

Collision Detection

SoftSprite tests for collisions while it draws each byte of data from the shape table. The result of this test is stored in the variable which precedes the USR function (COL in the preceding statements). This variable is set to a nonzero value when a collision occurs and to zero when no collision takes place.

Experienced Atari programmers may have noticed that both the demonstration program and the arrow shape itself take advantage of an effect known as artifacting, which results when you display colored shapes (often a single pixel) in a higher resolution than the computer's display hardware can handle. Artifacting explains why only alternate columns of the arrow shape contain data. As a result, the arrow is either green or blue depending on whether it begins on an odd-numbered screen column or an even-numbered column. You can find more information about artifacting in *COMPUTE!'s Second Book of Atari Graphics*, available from COMPUTE! Books.

Artifacting also affects collision detection in SoftSprite. Simply put, SoftSprite cannot detect a collision between two artifacted shapes if they are different colors. For instance, if one shape is solid blue and the other is solid green, they can pass through one another without triggering a collision in SoftSprite. In that case, the area where the shapes overlap turns white. This effect can be observed in the demonstration program; when the moving arrow overlaps the blue arrow, no collision is detected, but the program does detect a collision when the moving arrow overlaps the green arrow. One way to circumvent the problem is to design one of the shapes so that it's completely enclosed by a solid border.

Animation

Lines 150 and 170 show how SoftSprite can copy a shape from the shape table to any portion of the display screen. The first USR call in line 150 puts the shape on the screen as usual. Next, the program

POKEs a zero into the first byte of the shape table. The effect of this POKE is to turn off the shape without removing its image from the screen. Once this is done, however, the data for that shape's location is lost, and the only way to erase it is to redraw it at exactly the same screen position.

You may copy a shape to the screen as many times as you like. The first and second bytes of the shape table contain the last screen address where the shape appeared, held in low byte/high byte format.

Animation is simply a matter of turning a series of shapes on and off at the same screen position. Unlike other bitmapped graphics routines, SoftSprite does not require that each frame of the animation be the same size.

The shapes drawn by SoftSprite will always flicker when you move them. Because the program shifts each byte of the shape as it is being drawn, there is not enough time during the computer's vertical blank interrupt to perform the shift. Because the drawing occurs outside the vertical blank period, some flickering is inevitable.

Softsprite does not check for boundary errors and faithfully draws the designated shape at whatever screen coordinates you assign. If you attempt to draw outside the screen boundaries, you might corrupt important information, such as the data stored above the computer's RAMTOP pointer. The only error checking performed by the program is to determine whether you have supplied the right number of parameters in the USR call.

Machine language programmers should note that SoftSprite uses nearly all of the computer's floating point registers; any attempt to call the SoftSprite ML code from within a vertical blank interrupt routine will probably cause a system crash.

SoftSprite Demonstration

For instructions on entering these programs, please refer to "COMPUTE!'s Guide to Typing In Programs" elsewhere in this issue.

```
KL 5 REM COPYRIGHT 1987
BJ 6 REM COMPUTE! PUBLICATIO
     NS, INC.
FD 7 REM ALL RIGHTS RESERVED
     .
PA 10 ? CHR$(125);"LOADING D
      ATA FILES.."
OD 20 REM reserve memory for
       SOFTSPRITE{9 SPACES}p
      rogram and shape table
       using{9 SPACES}string
      s:
NK 30 DIM SOFTSPRITE$(281),S
      HAPE$(26)
GL 40 REM initialize both st
      rings:
KK 50 SOFTSPRITE$(1)="A":SOF
      TSPRITE$(281)="A":SOFT
      SPRITE$(2)=SOFTSPRITE$
      :SHAPE$=SOFTSPRITE$
HL 60 REM find starting addr
      esses of{12 SPACES}SOFT
      SPRITE program and sha
      pe{10 SPACES}table:
LL 70 SOFTSPRITE=ADR(SOFTSPR
      ITE$):SHAPE=ADR(SHAPE$
      )
AH 80 REM load shape table i
      nto the{13 SPACES}memor
      y reserved for the str
      ing{8 SPACES}"SHAPE":
FA 90 FOR OFFSET=0 TO 25:REA
      D DATA:POKE SHAPE+OFFS
      ET,DATA:NEXT OFFSET
ID 100 REM load SOFTSPRITE p
       rogram into
       {10 SPACES}memory rese
       rved for the string
       {8 SPACES}"SOFTSPRITE
       ":
EO 110 FOR OFFSET=0 TO 280:R
       EAD DATA:POKE SOFTSPR
       ITE+OFFSET,DATA:NEXT
       OFFSET
BB 120 REM open screen to gr
       aphics 8 full
       {8 SPACES}screen mode
        with a black
       {14 SPACES}background:
AH 130 GRAPHICS 8+16:POKE 71
       0,0
CL 140 REM put shape on an e
       ven X{16 SPACES}coordi
       nate causing it to
       {14 SPACES}appear gree
       n:
DJ 150 COL=USR(SOFTSPRITE,10
       0,95,SHAPE):POKE SHAP
       E,0
MP 160 REM put shape on a od
       d X{18 SPACES}coordina
       te causing it to
       {14 SPACES}appear blue
       :
EE 170 COL=USR(SOFTSPRITE,18
       1,95,SHAPE):POKE SHAP
       E,0
AB 180 X=140:Y=95
EJ 190 REM put shape on even
        coordinate,
       {9 SPACES}but do not
       rubber-stamp it,
       {11 SPACES}will be use
       d as a sprite:
JP 200 COL=USR(SOFTSPRITE,X,
       Y,SHAPE):SOUND 0,COL,
       10,8
GE 210 REM scan keyboard for
        key press:
BP 220 KEY=PEEK(764):IF KEY=
       255 THEN 220
DB 230 IF KEY=14 THEN Y=Y-2:
       GOTO 200
DB 240 IF KEY=15 THEN Y=Y+2:
       GOTO 200
AC 250 IF KEY=6 THEN X=X-2:G
       OTO 200
AC 260 IF KEY=7 THEN X=X+2:G
       OTO 200
AA 270 IF KEY=33 THEN COL=US
       R(SOFTSPRITE,SHAPE):G
       OTO 220
GH 280 GOTO 220
ML 290 REM shape table of ar
       row, 1st four
       {8 SPACES}bytes MUST
       ALWAYS be zero;
       {12 SPACES}followed by
        height & width:
MM 300 DATA 0,0,0,0,10,2
KP 310 REM rest of shape tab
       le, actual{11 SPACES}s
       hape data of arrow:
EN 320 DATA 170,0,170,0,160,
       0,168,0,168,0,170,0,1
       38,0,138,128, 2,128,2
       ,128
HL 340 REM
OC 350 REM the folowing data
        file the{12 SPACES}SO
       FTSPRITE program :
HN 360 REM
KI 10000 DATA 216,104,133,22
         2,201,1,240,29,201,
         3,240,11,10,168,104
         ,136,208,252,169,32
         ,133,195,96,169,0,1
         33
IK 10010 DATA 222,104,133,22
         1,104,133,220,104,1
         04,133,219,104,133,
         225,104,133,224,160
         ,6,132,231,136,177,
         224,133
DI 10020 DATA 228,136,177,22
         4,133,229,133,230,1
         36,177,224,133,233,
         136,177,224,133,226
         ,136,177,224,133,23
         6,136,177
PP 10030 DATA 224,133,237,24
         0,87,169,0,133,235,
         165,226,133,227,166
         ,228,169,0,133,234,
         164,231,177,224,240
         ,10
CN 10040 DATA 164,233,240,6,
         74,102,234,136,208,
         250,5,235,72,164,22
         7,49,236,240,2,133,
         212,104,81,236,145
BK 10050 DATA 236,165,234,13
         3,235,230,227,230,2
         31,202,208,209,200,
         165,234,72,49,236,2
         40,2,133,212,104,81
         ,236
GL 10060 DATA 145,236,165,23
         6,24,105,40,133,236
         ,144,2,230,237,198,
         229,208,169,165,222
         ,240,16,201,6,240,9
FA 10070 DATA 160,0,132,212,
         132,213,152,145,224
         ,96,208,149,165,220
         ,41,7,133,233,160,3
         ,145,224,70,221,165
JJ 10080 DATA 220,106,74,74,
         133,226,136,145,224
         ,169,0,133,212,133,
         213,6,219,42,6,219,
         42,6,219,42,133
NK 10090 DATA 237,166,219,13
         4,236,6,219,42,6,21
         9,42,170,165,219,24
         ,101,236,133,236,13
         8,101,237,133,237,1
         65
DJ 10100 DATA 88,24,101,236,
         133,236,136,145,224
         ,165,89,101,237,133
         ,237,136,145,224,16
         5,230,133,229,169,6
         ,133
LC 10110 DATA 231,133,222,20
         8,161
```

The Power Of ON-GOTO And ON-GOSUB

Ronald R. Lambert

This tutorial details the use of ON-GOTO and ON-GOSUB, two of the most powerful statements in BASIC. Although the program examples are written in Atari BASIC, the principles apply to most microcomputers with BASIC.

ON-GOTO and ON-GOSUB are among the last commands learned by many new BASIC programmers. Even intermediate-level programmers may pass them up because the same results can be achieved by IF-THEN statements and logical comparisons. But the power of ON-GOTO and ON-GOSUB is appreciable. Once you learn how to use these commands, you'll wonder how you got along without them for so long.

The ON commands are used in program branching in cases where the value of some expression (a variable, calculation, and so on) determines the program line where execution goes next. This seems straightforward enough; but there is more to these commands than meets the eye. We'll look at how to handle these commands in Atari BASIC, but the examples will work with little or no modification in almost any version of BASIC.

Multiple Branches With ON-GOSUB

Suppose you are writing a game program, and you want to move a shape in one of four directions around the screen. You might, typically, convert the keyboard input so that if a cursor key is pressed, the program assigns a variable different values representing the corresponding direction. The variable may be set to 1 when you press cursor up, 4 when you press cursor right, and so forth. Similarly, you might convert joystick input to a series of numeric values indicating directions.

In this case, you might have a separate subroutine to handle movement in each direction. A single ON-GOSUB statement can handle all the branches, directing execution to the proper subroutine. Here is a short program that demonstrates the multibranching feature of ON-GOSUB. When it prompts you for a number, enter a value from 1 to 4.

```
10 PRINT "ENTER NUMBER: "
   ;:INPUT D
20 PRINT "D=";D;": ";
30 ON D GOSUB 40,50,60,70
   :GOSUB 80:GOTO 10
40 PRINT "CAME TO LINE 40
   : ";:RETURN
50 PRINT "CAME TO LINE 50
   : ";:RETURN
60 PRINT "CAME TO LINE 60
   : ";:RETURN
70 PRINT "CAME TO LINE 70
   : ";:RETURN
80 PRINT "CAME TO LINE 80
   . ":RETURN
```

When you run this program, it asks you to enter a number. If you enter 1, the program displays the message *Came to line 40.* After it returns from the subroutine at line 40, notice that the program does not proceed to the routines in lines 40–70. Instead, it proceeds to the next BASIC statement in the line, GOSUB 80. This statement is always performed, allowing you to note the difference between ON-GOSUB and a normal GOSUB command.

Now try entering a number smaller than 1 or greater than 4. Notice how ON-GOSUB responds. In this case, the program skips all of the possible destination lines listed in the ON-GOSUB statement. ON-GOSUB does not cause any branches, and the program proceeds to the next statement (GOSUB 80). For example, if you enter 0 at the prompt, the program prints *Came to line 80.*

The Destination List

The program demonstrates some basic features which ON-GOTO and ON-GOSUB share. Both statements are followed by a list of destination line numbers. The first line number is used when the tested expression equals 1; the second line number is used when the expression equals 2, and so on. If the expression evaluates to zero, no branch is taken, and the program proceeds to the next statement.

In the simple example above, the expected number series 1-2-3-4 corresponds neatly to the needs of our program. In other cases, you might have an expression that doesn't evaluate in 1-2-3 order. Say, for instance, that your expression might produce the values 1, 3, 4, and 5 (the value 2 is missing).

Atari BASIC does not allow you to simply omit a line number from the destination list, although that does work in some other versions of BASIC. (The statement ON X GOSUB 100,,300,400,500 causes a syntax error in Atari BASIC, but it works in most other BASICs.) One solution is to create a do-nothing subroutine that consists of nothing but a RETURN statement. In the unlikely event that your program ever produces the unexpected value, the program will return without doing anything. For example, including the line 200 RETURN allows you safely to use the statement ON X GOSUB 100,200,300,400,500.

Branching With ON-GOTO

You can make multiple branches in much the same way with ON-GOTO. To see how this works, substitute GOTO for each GOSUB in line 30 of the example program, and replace each RETURN in lines 40–80 with the statement GOTO 10.

When you run the program with these changes, it branches to the same destinations as before, except that it never reaches line 80 unless you enter a value outside the range 1–4. The last command in line 30 (GOTO 10) is rendered superfluous and may be deleted.

The real difference, of course, between ON-GOSUB and ON-GOTO is in what happens after the branch occurs. In the case of ON-GOSUB, the program returns to the next BASIC statement after the ON-GOSUB statement when a RETURN is encountered. On the other hand, an ON-GOTO statement, like a regular GOTO, doesn't automatically return to the part of the program where it originated.

Priority Sifting

The basic use for ON-GOSUB and ON-GOTO, then, is to test a range of conditions and branch to any of several possible destinations. In the previous example, the condition was the value assigned to the variable D, and the range was a series of numbers from 1–4, inclusive. In addition to such basic uses, ON-GOTO and ON-GOSUB can be employed to make a series of decisions in order of priority—a technique which we'll explain with a practical example.

Suppose that you are writing a text-adventure program. In an adventure, or other simulation of real-life environments, one of the most basic problems is how to assign different properties to objects and discriminate intelligently among them. For example, say that you, as the hero or heroine of the adventure, try to pick up a castle and stuff it into a gunny sack containing other objects. To respond to this action, the program must first be able to tell what properties the castle possesses. You don't want to print an inappropriate message like *You must drop something else before picking up the castle*, because that would imply that the castle is something that can be picked up.

In a typical adventure program, an object might have any of several different properties. Some objects can be picked up, but others are immovable; some are valuable, while others are useless; some are dangerous; some can be eaten or drunk; some possess magical powers, and so forth. Certain properties exclude others, as well. If you can't pick up a castle, then it's not something which you can eat or use as a weapon, for instance. To deal sensibly with this wide range of possibilities, the program needs a subroutine that will consider certain factors in order of precedence and respond as fits the situation.

ON-GOTO and ON-GOSUB are ideally suited for this sort of work. While you could accomplish the same task with a series of IF-THEN commands, see how neatly and concisely ON-GOTO performs these tasks in the following example. In a text adventure, the program might use this routine whenever you try to pick something up.

```
700 TAKE=0:ON IMMOVE GOTO
     710:ON WEIGHT>STRENG
    TH GOTO 720:ON BURDEN
    >CAPACITY GOTO 730:TA
    KE=1:RETURN
710 PRINT "YOU CAN'T PICK
     UP A ";OBJECT$;"!":R
    ETURN
720 PRINT "THAT'S TOO HEA
    VY FOR YOU.";:ON  NOT
     INJURED GOTO 740:PRI
    NT CHR$(126);" IN YOU
    R PRESENT CONDITION."
    :RETURN
730 PRINT "YOU CAN'T CARR
    Y ALL THAT. YOU'LL HA
    VE TO DROP SOMETHING.
    ";
740 PRINT :RETURN
```

The variable IMMOVE shows whether an object can be picked up; IMMOVE equals 0 if an object is movable or 1 if it's not. Let's assume that the program sets IMMOVE appropriately before we enter this routine. If you try to pick up a castle, IMMOVE equals 1 when the program executes line 700. The first ON-GOTO statement in line 700 responds to that condition, causing a branch to line 710, which prints *You can't pick up a castle* and returns. On the other hand, if you decide to pick up something more sensible, such as a goblet or newspaper, IMMOVE will be zero, in which case no branch is taken at the first ON-GOTO.

At this point, we have established the most basic fact needed for a response—whether the object can be picked up at all. If an object can't be carried, then we needn't waste time deciding other questions such as whether you have sufficient strength to lift this particular object, or whether you already are carrying too many other things.

Let's assume that you pick up something sensible and pass the first ON-GOTO test. The next statement in line 700 happens to be another ON-GOTO statement which compares the weight of the object to your current strength, to determine whether you have enough vigor to grab it. Again, we'll assume that the variables WEIGHT and STRENGTH get assigned elsewhere in the program.

If you fail the strength test, the ON-GOTO statement diverts execution to line 720, which prints the message *That's too heavy for you.* This line also contains a secondary ON-GOTO which checks the variable INJURY to see if an injury is responsible for your inability to take the object. If so, the program prints *in your present condition* at the end of the last message. Notice the use of the logical operator NOT in this comparison. If the variable INJURED equals zero, the program branches to line 740, which simply prints a blank line and exits the routine with RETURN. (Note that the result of the NOT operator may

differ in other versions of BASIC.)

At this point, we have passed the second level of discrimination; we know that the object is both movable and light enough to pick up. The third test in line 700 tests whether you already are carrying so many objects that you can't pick up another, comparing the variables BURDEN and CAPACITY. We'll say that BURDEN represents the size of your current load, and CAPACITY indicates the maximum number of objects you can carry.

If your current burden exceeds the maximum capacity, the last ON-GOTO statement causes a branch to line 730, which advises you to drop something else before trying to grab the desired object. Only if all three conditions are met does the program allow you to pick up the object and add it to your inventory.

Notice how the logical tests are arranged in order of priority, so that we establish the most basic facts first. If the object is immovable, then there's no point in checking whether you can lift it or have room in your sack to stow it—and it would diminish the realism of the adventure to print messages about those secondary topics. Similarly, if you are too weak or injured to pick something up, then we don't care how much room is in your sack. This is quite different from the joystick-reading example mentioned earlier, where all of the possible program branches have equal weight. The ability to sift through a series of tests in order of priority is essential to complex decision making.

Emulating IF-THEN-ELSE

If you're familiar with other versions of BASIC, you may have noticed something about the ON-GOTO statements in the previous example. They operate something like an IF-THEN-ELSE structure, in that the computer continues to read additional statements in the same program line if the conditions for the first ON-GOTO are not satisfied.

In most versions of BASIC, program execution drops down to the next program line whenever the conditions for an IF statement are not fulfilled. But many newer versions of BASIC allow you to add an ELSE command to an IF-THEN structure on the same program line, so that in the event the first IF test fails, the computer goes on to perform the statements that appear after ELSE. When ELSE is followed by an additional IF-THEN construct (which may include another ELSE, in turn), you can create a single line that performs quite sophisticated logical tests.

While Atari BASIC does not support the ELSE statement, you can use ON-GOTO and ON-GOSUB to much the same effect. Even though these commands can only result in a decision whether or not to branch, they can be followed on the same program line by other BASIC statements. This is demonstrated in lines 720 and 740 of the previous example. The ON-GOTO statement in line 720 has exactly the same effect as an IF-THEN-ELSE statement.

Series Comparisons

Yet another use for ON-GOTO or ON-GOSUB is for making a series of logical comparisons, which would otherwise require a FOR-NEXT loop containing one or more IF-THEN statements. This is demonstrated in the following program. Despite its brevity, this is a complete program, which converts into Arabic equivalents Roman numerals you have entered.

```
10 DIM RN$(120),C$(1),R(6
   ):PRINT CHR$(125)
20 R(0)=1:R(1)=5:R(2)=10:
   R(3)=50:R(4)=100:R(5)=
   500:R(6)=1000
30 PRINT :PRINT "ENTER RO
   MAN NUMERALS: ";:INPUT
    RN$
40 Z=LEN(RN$):IF Z=0 THEN
    30
50 TRAP 110:L=0:S=0
60 FOR X=1 TO Z:RESTORE 1
   00:Y=-1
70 Y=Y+1:READ C$:ON C$<>R
   N$(X,X) GOTO 70
80 N=R(Y):S=S+L*(L>=N)-L*
   (L<N):L=N
90 NEXT X:PRINT "ARABIC E
   QUIVALENT: ";S+N:GOTO
   30
100 DATA I,V,X,L,C,D,M
110 PRINT CHR$(253);"ILLE
    GAL CHARACTER: ";RN$(
    X,X):GOTO 30
```

The heart of this program is the ON-GOTO command in line 70. Each Roman numeral you enter is compared to the legal characters in the DATA statement. As long as the characters don't match, the ON-GOTO command loops back to the beginning of line 70 to READ the next character from the DATA list. When there is a match, program execution proceeds to the next statement after the ON-GOTO, adding the value assigned to that numeral to the cumulative sum for the whole number.

There are two advantages to using ON-GOTO in this situation instead of a FOR-NEXT loop. First, the loop terminates as soon as the tested condition is met—no time is wasted performing meaningless tests while a FOR-NEXT loop completes its appointed rounds. (It is possible, of course, to exit a FOR-NEXT loop prematurely, but this a dangerous programming practice, since you may overflow the computer's stack and cause an error if you jump out of too many loops without performing the right number of POPs.)

Incidentally, the formula used in this program to compute Arabic equivalents for Roman numerals is a little different from the one you may have learned in school. Instead of subtracting a smaller number from a following larger number (as in IV, XL, and so on), the program subtracts the smaller number from the cummulative sum when it is followed by a larger number, and then adds the larger number. The results are the same in both cases.

Other Creative Uses

The final demonstration program exhibits a number of creative uses of ON-GOTO and ON-GOSUB. The program asks you to enter a date, and it then tells you what day of the week that date falls on. For instance, if you enter JULY followed by 4 followed by 1776, the program prints the message *JULY 4, 1776 is a Thursday*. To keep the code reasonably short, its responses are limited to dates following the calendar reform of 1752, when the modern Gregorian calendar was adopted in the English-speaking world.

This 1752 limit on date calculations is the result of the Gregorian calendar reform: To bring the calendar back into harmony with sun time so that the equinoxes and solstices would fall on their traditional dates, the day Wednesday, Septem-

ber 2, 1752 was followed by Thursday, September 14, 1752. As a result, computations that take the modern calendar back prior to that date will produce a false weekday result unless you make a specific adjustment. (The year 1752 is when England and its colonies, including the ones which later became the United States, made the change. Many European countries made the change earlier, beginning in 1582 when the Gregorian calendar system was introduced. Pope Gregory decreed that Thursday, October 4, 1582 was to be followed by Friday, October 15, 1582. For countries where the change was made on the original date, change the 639797 in line 110 to 577736.)

The same calendar reform changed the formula for calculating leap years. Under the modern Gregorian calendar, centenary years (years evenly divisible by 100) are *not* leap years unless they are also evenly divisible by 400. This creates the interesting category of *leap centuries*. This explains the length of the formula in line 90 for determining leap years.

Observe carefully how this program performs its functions, especially the ON-GOTO and ON-GOSUB commands. This example may suggest additional ways to use these commands in your own programs.

Calendar

For instructions on entering this program, please refer to "COMPUTE!'s Guide to Typing In Programs" elsewhere in this issue.

```
OB 10 DIM MONTH$(9),DAYNAME$
      (6),MO$(2),M(12)
EN 20 M(0)=0:X=0:PRINT CHR$(
      125)
BB 30 X=X+1:ON M(X-1)<>31 GO
      SUB 50:ON M(X-1)=31 GO
      SUB 40:ON X=8 GOSUB 50
      :ON X<12 GOTO 30:GOTO
      60
BE 40 M(X)=30:RETURN
BB 50 M(X)=31:RETURN
LH 60 PRINT :PRINT "Month na
      me: ";:INPUT MONTH$:TR
      AP 60:PRINT "Day: ";:I
      NPUT DAY:PRINT "Year:
      ";:INPUT YEAR
CE 70 RESTORE 190:X=0
MB 80 X=X+1:ON X>12 GOTO 160
      :READ MO$:ON MO$<>MONT
      H$(2,3) GOTO 80:M=X
GO 90 LEAP=(INT(YEAR/4)=YEAR
      /4)-(INT(YEAR/100)=YEA
      R/100)+(INT(YEAR/400)=
      YEAR/400):M(2)=28+LEAP
AG 100 ON DAY>M(M) GOTO 170
AH 110 DAYS=INT((YEAR-1)*365
      .2425):FOR X=0 TO M-1
      :DAYS=DAYS+M(X):NEXT
      X:DAYS=DAYS+DAY:IF DA
      YS<639797 THEN 180
EI 120 RESTORE 200:X=0:WEEKD
      AY=DAYS-INT(DAYS/7)*7
      :IF WEEKDAY=0 THEN WE
      EKDAY=7
JI 130 X=X+1:READ DAYNAME$:O
      N X<WEEKDAY GOTO 130
IA 140 PRINT MONTH$;" ";DAY;
      ", ";YEAR;" is a ";DA
      YNAME$;"day."
DF 150 GOTO 60
CM 160 PRINT "What month is
      ";MONTH$;"?":ZANY=ZAN
      Y+1:ON ZANY<2 GOTO 60
      :PRINT "Let's get ser
      ious!":GOTO 60
AC 170 PRINT MONTH$;" only h
      as ";M(M);" days.":ON
       M<>2 OR DAY<>29 GOTO
       60:PRINT YEAR;" is n
      ot a leap year.":GOTO
       60
CL 180 PRINT :PRINT "This da
      te was prior to the a
      doption ofthe modern
      (Gregorian) calendar.
      ":GOTO 60
LG 190 DATA AN,EB,AR,PR,AY,U
      N,UL,UG,EP,CT,OV,EC
GB 200 DATA Mon,Tues,Wednes,
      Thurs,Fri,Satur,Sun ©
```

Chaining Programs In Applesoft BASIC

Richard J. Kaufman

By appending a few lines to Applesoft BASIC programs, you can cause them to call each other and still keep their variables intact.

When a BASIC program gets so large that it exhausts all available memory, you have very few options. If you try to make the program more memory-efficient, the first step is usually to delete the REMarks; but this can prove a disaster when you come back to the program six months later. Even the best programmers forget the details of old programs, and can benefit from a few reminders. You can also split the program in two and save all the shared variables in a disk file, which the second program can read and use. This is slow at best. The old DOS manual shows a CHAIN command, which is just what you need in these situations. Unfortunately, CHAIN works only with Integer BASIC, which is no longer supported by Apple. Most high-level languages provide some means to call another program that does not remain memory resident, and include a means of passing variables. Applesoft BASIC, alas, does not. It's too basic.

There is a way, however, to chain (transfer control between) Applesoft programs, leaving the first program's variables and arrays intact. It requires an understanding of how Applesoft handles memory, and a little experimentation, but it will greatly increase the amount of BASIC code you can run at one session. Even if memory size is not a problem, you may find several small, modular programs easier to deal with than one huge one.

Table 1: Applesoft Pointer Locations

Pointer Location		Name	Description
Hex	Decimal		
$67 $68	103 104	TXTTAB	Start of program (usually $0801).
$69 $6A	105 106	VARTAB	Start of simple variable space.
$6B $6C	107 108	ARYTAB	Start of array space.
$6D $6E	109 110	STREND	End of variable space.
$6F $70	111 112	FRETOP	Start of string space.

Inside Applesoft BASIC

Table 1 summarizes the pairs of memory locations that tell BASIC where to find the program, its variables, arrays, and strings.

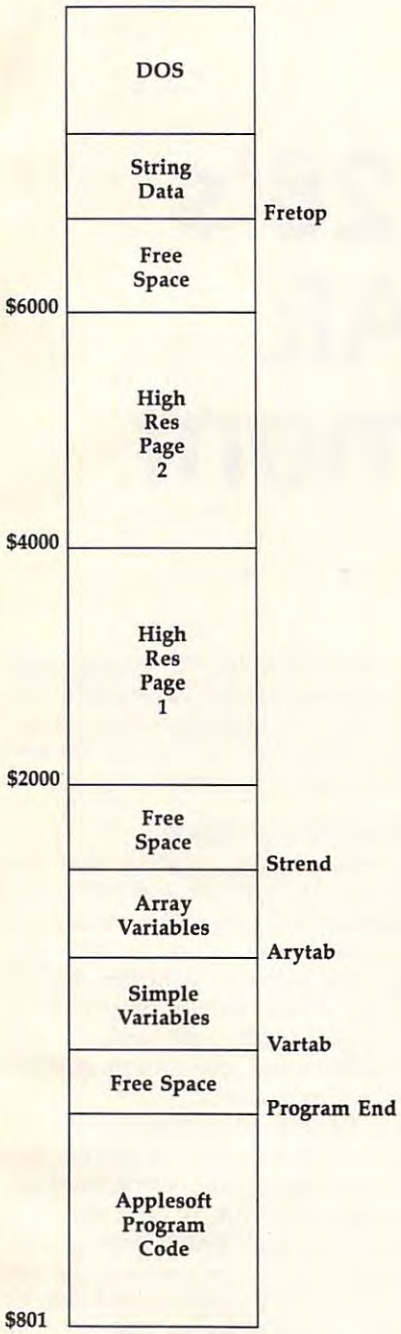
VARTAB, the start of simple variable space, usually begins right after the program text itself. However, it can be set to any desired value using the LOMEM command. This is the key to our chaining technique. Imagine that a program—which we'll call FIRST—has issued a command to run a second program named SECOND:

```
PRINT CHR$(4);"RUN SECOND"
```

When SECOND loads at TXTTAB (location $0801), it overwrites the previous contents of BASIC program memory, including at least part of FIRST. Assume that the program text for SECOND is shorter than the text of FIRST. In this case, the variable space is not yet destroyed, and it will not be until SECOND starts to use its variables. The figure helps you to visualize what is happening.

If SECOND sets VARTAB, ARYTAB, STREND, and FRETOP to the values in effect when FIRST was exited, all of FIRST's variables will be available to SECOND. The only exceptions are strings that were assigned values by literal statements in FIRST—for example, A$="THIS IS A STRING"—rather than by an INPUT statement, a disk read, or another string operation. Such strings must be reassigned. This is because of the way Applesoft handles string variables. The entry for a string variable in the variable table doesn't actually con-

Applesoft Memory Organization



tain the text of the string. Instead, it contains a pointer holding the address of the string's text. The text of strings assigned by INPUT or created by some string operation will be stored in the tables above FRETOP. However, for strings with literal assignments, the variable table entry will point to the literal statement in the program text. When another program is chained, the program text containing the literal definition will be overwritten, but the variable table entry will still point into the program text area. Thus, the string assignment is no longer valid.

There is a way to avoid this problem. Literal strings involved in string operations such as concatenation are given definitions in the string pool above FRETOP, even if the string isn't actually changed. Thus, if you replace the literal string definition A$="THIS IS A STRING" with A$="THIS IS A STRING" + "", the string definition will be preserved after chaining. Adding (concatenating) an empty (null) string to the literal assignment will cause a copy of the string definition to be created in the string pool above FRETOP.

Implementing An Applesoft Chain

The need to chain BASIC programs often is brought about by the use of high-resolution graphics. Program space starts at $0801, and hi-res graphics page 1 starts right in the middle of the program space, at $2000. Translated to decimal, this leaves a mere 6144 bytes for the program. You can use hi-res page 2 for graphics (it starts at $4000), but suppose you need both high-resolution screens for some special effects? Program 1 shows how to cause an Applesoft program to load at any desired location in memory. (Starting the program above hi-res page 2 will give you more room, unless you use many large arrays or lots of strings.)

Line 5 of Program 1 is the key to this technique. This line can be appended at the beginning of any program you wish. Note that the variable LOC is set to the one location beyond the last byte of hi-res screen 2. This is fine if you want the program to load at that location. Otherwise, set LOC to the address of the location where you want your program to start. Of course, if you're not doing graphics, you can leave the program loading address alone.

Line 6 sets LOMEM, the bottom of variable storage, to location 34817. This gives you 10,241 bytes of program text space above hi-res screen 2. The optimum value for LOMEM will be determined by the longest program to be chained. You can estimate this by examining a catalog of the programs to be chained.

To calculate the approximate length of a program, multiply the number of sectors shown in the catalog entry for the program by 256 for DOS 3.3 or by 512 for ProDOS. For instance, a program that occupies four sectors when saved under DOS 3.3 is about 1024 (4 * 256) bytes long. A four-sector program saved under ProDOS will be about 2048 (4 * 512) bytes long. A more precise method is to check the difference between the values of VARTAB and TXTTAB after loading the longest program in the chain and use that value. The following line shows how to do this:

```
PRINT (PEEK(105) + 256 * PEEK(106)) -
  (PEEK(103) + 256 * PEEK(104))
```

Once you've determined the program length, add that value to the program load address to get the minimum value for LOMEM.

In practice, it's usually easiest to start with 35000 for LOMEM and adjust the value if you find that you have memory problems. If you are not using both graphics screens, you can use a much lower value for LOMEM, but this is not likely to be necessary unless you are using a lot of array space.

Lines 10–30 of Program 1 assign a literal value to a string, use that string in a concatenation operation, print it, accept a numeric value for the variable X, and assign the value 9999 to Y.

We now are ready to save the program pointers, a task which is done in lines 900–910. The top of hi-res page 1 is 16383. We work down from there, putting the pointer bytes into the top of the screen. They do not show up on the screen, since they are nondisplayed "slack bytes." If you are not doing graphics, and you need to chain, you may have to find some other safe place to store your pointers. Perhaps you can use space in one of the DOS buffers or else find some unused space in low memory. If waiting for the extra time required doesn't bother you, the pointers can even be saved to a file. Line 920 runs Program 2.

Lines 5 and 6 of Program 2 retrieve the program pointers stashed by Program 1. Lines 10–40 then print the strings and variables

assigned in Program 1. As you can see from looking at the output, the literal string assigned in line 10 of Program 1 didn't make it. The string table points to the location where the string resided in Program 1's text, but that area of memory now contains Program 2's text. It's necessary to reassign a value to the string. However, line 15 of Program 2 shows that the involving the literal string in a string operation—as in line 15 of Program 1—will create a string that can be successfully chained.

Remember, strings entered from the keyboard or from disk files will not need resetting, while those initialized with data statements will need to be reinitialized.

You can continue chaining from program to program. Just keep saving and retrieving the pointers.

For instructions on entering these programs, please refer to "COMPUTE!'s Guide to Typing In Programs" elsewhere in this issue.

Program 1: First

```
7E 5 LOC = 24576 + 1: IF PEEK (1
     Ø3) + PEEK (1Ø4) * 256 < >
     LOC THEN POKE LOC - 1,Ø: PO
     KE 1Ø3,LOC - INT (LOC / 256
     ) * 256: POKE 1Ø4, INT (LOC
      / 256): PRINT CHR$ (4)"RUN
      PROGRAM 1"
47 6 LOMEM: 34817
92 1Ø E1ND$ = "THIS IS PROGRAM 1
     "
9B 15 E2ND$ = E1ND$ + ""
ØE 2Ø PRINT "STRING = ";E1ND$
A9 3Ø PRINT "ENTER A NUMBER, X "
     : INPUT X:Y = 9999
9B 99 REM STORE POINTERS AT END
      OF HIGH-RES SCREEN 1
FB 9ØØ POKE 16383, PEEK (1Ø5): P
      OKE 16382, PEEK (1Ø6): PO
      KE 16381, PEEK (1Ø7): POK
      E 16380, PEEK (1Ø8): POKE
       16379, PEEK (1Ø9): POKE
      16378, PEEK (11Ø)
DB 91Ø POKE 16377, PEEK (111): P
      OKE 16376, PEEK (112)
BE 92Ø PRINT CHR$ (4);"RUN PROGR
      AM 2"
97 93Ø END
```

Program 2: Second

```
57 5 POKE 1Ø5, PEEK (16383): POK
     E 1Ø6, PEEK (16382): POKE 1
     Ø7, PEEK (16381): POKE 1Ø8,
      PEEK (16380)
2E 6 POKE 1Ø9, PEEK (16379): POK
     E 11Ø, PEEK (16378): POKE 1
     11, PEEK (16377): POKE 112,
      PEEK (16376)
78 1Ø PRINT "STRING =[";E1ND$;"]
     "
8A 15 PRINT "STRING =[";E2ND$;"]
     "
D3 2Ø E1ND$ = "THIS IS PROGRAM 2
     "
84 3Ø PRINT "X = ";X;" Y = ";Y
FE 4Ø PRINT "STRING = ";E1ND$; ©
```

The 128's CHAR Statement

Jim Butterfield, Associate Editor

In this article, associate editor Jim Butterfield examines the highly versatile but quirky CHAR statement in the Commodore 128's BASIC 7.0.

The CHAR statement of BASIC 7.0 can display characters on any type of screen—in 40 or 80 columns, text or high resolution. Whatever you have, CHAR will deliver the message. But the statement has its own special quirks, and at least one bug. This article explores CHAR in detail.

The format of the CHAR statement is described in the *Commodore 128 System Guide* as follows:

CHAR ***color source*****, x, y, string,** ***rvs***

The parameters shown in italics are optional. As a more easily understood example, this statement prints the word HELLO in column 9 of row 4:

```
CHAR ,9,4,"HELLO"
```

We'll describe all of CHAR's parameters later in this article.

Terminology

It's customary to call the high-resolution screen a *graphics* screen, and call the normal display screen a *text* screen. This can be confusing, since the 128's character set contains both text (letters, numerals, and so on) and what you might call graphics—special symbols such as hearts or diamonds. To prevent confusion, we'll use the term *bitmapped* to describe the high-resolution screen and *character* to designate the conventional text screen.

Supplying Values

CHAR always requires that you specify the column and row where printing will occur. Remember that the rows and columns are numbered starting at 0 rather than 1. Thus, if you want to print at what you might ordinarily consider column 10, row 5, use column and row values of 9 and 4.

On the 40-column screen, the row value must not be greater than 24 and the column value must not exceed 39. This is true for both character and bitmapped modes. On the 80-column screen, the column and row values must fit within the currently defined output window. If you haven't set a window, the default output window occupies the whole screen. When you're working in 80 columns, a GRAPHIC statement switches CHAR's output over to the bitmapped screen. In this case, GRAPHIC 5 (return to 80 columns) alone won't bring it back; you must cancel the bitmapped screen with GRAPHIC 0:GRAPHIC 5 (or with GRAPHIC CLR, which eliminates the bitmapped screen altogether).

Partly because of this confu-

sion, and partly because of a bug described later, I don't like to use CHAR on the 80-column screen. I'd much rather PRINT cursor-movement characters followed by the text I want to appear.

Color Source, Foreground, And Background

The first parameter in a CHAR statement is the color source. This value is optional. If you're in character mode (not high resolution), the number is ignored. If you are in bitmapped mode, a value of 0 selects the background color for printing, and a value of 1 selects the foreground color. Most programmers omit this parameter, which defaults to the foreground value (1).

It's hard to explain in words what it means to print in background and foreground colors. We'll do it the clearest way, with an example. From the 40-column screen, type this statement and press RETURN:

```
GRAPHIC 2
```

You'll end up with a lot of clutter on the upper part of the screen, but that's intentional in this case (it could be prevented with GRAPHIC 2,1). Now change the color scheme by typing these statements:

```
COLOR 0,5
COLOR 1,3
FOR J=7168 TO 7407:POKE J,48:NEXT J
```

(If you make a typing mistake, the 128 will drop out of bitmapped mode to display the appropriate error message. Entering another GRAPHIC 2 statement will get you back.)

The two COLOR statements set distinctive colors that you'll recognize when they crop up in unexpected places. Note that the COLOR 0,5 statement (the 0 specifies the background color source) sets the background to purple only on the character screen. The bitmapped screen is not affected at all. The COLOR 1,3 statement sets the foreground color to red. However, nothing appears to happen when we issue this statement or continue to type. It's odd that background affects only the character screen, and foreground, as it turns out, applies only to the bitmapped screen.

The FOR-NEXT loop sets the foreground and background colors on the first six rows of the bitmapped screen. I've chosen the colors cyan and black, and you'll see these colors sweep through the screen clutter as the loop executes. When we POKE colors, we use the color code 3 for cyan and the color code 0 for black. The value to POKE is computed as 3 * 16 + 0. If we were using a BASIC 7.0 statement such as COLOR, the corresponding color codes we'd use would be increased by 1: 4 for cyan and 1 for black.

Now we're ready to see exactly how the color code value works. Enter these lines:

```
CHAR 1,2,2,"TESTING"
CHAR 0,3,3,"MORE TESTS"
```

The screen colors were originally cyan on black. The first CHAR statement prints in a new foreground color (red) without changing the black background. The second CHAR statement prints in the original foreground color (cyan), but changes the background to the same color as the character screen background (purple). To change both, print twice with a different color source each time. Here's an example using a loop:

```
FOR J=0 TO 1:CHAR J,4,4,"HERE ARE
  BOTH COLORS":NEXT J
```

The rest of this article will skip the first parameter, assuming that the default value (1) will do the appropriate job.

Reverse Flag

If the reverse flag (*rvs* in the statement format shown above) is set to 0 or is omitted, the characters will print in the usual way—foreground color on background color. Setting the reverse flag to 1 (as in CHAR ,9,9,"GREETINGS",1) doesn't exactly reverse the color usage. Instead, it prints reversed characters. This is a fine point, and it may not make a difference in your case. But if you examine the various color combinations outlined in the previous section, you'll see that there is a difference.

You might like to repeat the above exercises, using an extra *,1* at the end of each CHAR statement. You'll find that you get new combinations: black letters against a red background, and purple letters against a cyan background.

Text Versus Graphics Mode

When you're in character mode, the rules are pretty simple. Whatever print mode the computer is in—lowercase/uppercase or uppercase/graphics—will be the mode used for the output of CHAR.

You can change modes during a CHAR display. Enter NEW and GRAPHIC CLR and then try this:

```
CHAR ,20,20,CHR$(142)+"HELLO"
  +CHR$(14)+"THERE"
```

For the 40-column screen, the entire screen makes the switch from uppercase/graphics to lowercase/uppercase mode. The 80-column screen can have both character sets displayed at the same time.

CHAR for the bitmapped screen seems at first to support only the graphics character set. But it can easily be made to support both. Try this:

```
GRAPHIC 2,1
CHAR ,10,10,CHR$(142)+"LOOK"
  +CHR$(14)+"HERE"
```

You may go back and forth between character sets, using CHR$(14) and CHR$(142), as often as you like. But you can't use programmed cursor movements or other control characters; the two just named are the only ones supported for CHAR usage on bitmapped screens. There is, however, an extra gimmick you can use on bitmapped screens. We'll talk about it, and give a program example, in a moment.

The Bug

When you use the CHAR statement on an 80-column screen, an error in the 128's logic causes two locations high in bank 0 to be changed. Bank 0 is where BASIC program text is kept. Your program won't be hurt unless it's very big (at least 38K in size)—but CHAR will corrupt a program that reaches the danger spot.

The problem is easy to get around. Just don't use the CHAR statement to print messages to the 80-column screen.

At the time of writing, Commodore is preparing a revised operating ROM for the Commodore 128. This new ROM may eliminate the bug. To test whether the bug exists on your machine, enter and run the following short program:

```
200 GRAPHIC CLR
210 BANK 0
220 POKE 54784,0
230 POKE 54785,0
240 CHAR ,1,1,"NOW TRY THIS"
```

If you have only a 40-column monitor, add the following two lines:

```
100 PRINT CHR$(27);"X"
300 PRINT CHR$(27);"X"
```

Do not add these lines if you are working in 80 columns. After you've run the program, check the contents of the two locations you POKEd to 0. You should still be set for bank 0, so just type:

```
PRINT PEEK(54784)
PRINT PEEK(54785)
```

If the values have changed from 0, your 128's ROMs have the bug. In the case of a very long BASIC program, those two locations might contain part of a line, in which case the 80-column version of the CHAR statement would corrupt two bytes of your program. Stay away.

Special Feature

When the 128 is in bitmapped graphics mode, there's an extra gimmick that CHAR can use. The character set can be switched to one that you create yourself. Normal printing to the screen won't be affected; only those characters produced by the CHAR command. Location 4588 ($11EC) controls this feature.

The address of the character set used by CHAR is kept in location 4588, which normally holds the value 208. That's a page number; multiply it by 256 and you get the address of the 128's character ROM, starting at location 53248 in the bank 14 configuration.

To create a custom character set, you don't have to design all the characters of the alphabet, plus numbers, punctuation, and so on. In the example given below, we define only three characters: one to replace the letter *A*, and the other two for *B* and *C*.

We'll replace these three letters with pictures of little people. They must fit into the space a single character occupies, so they will be quite small. If you want bigger pictures, it's not hard to hook together two or more characters so that they jointly depict some object.

As we switch from one character to another, using the CHAR command, the figures will be displayed in slightly different postures. They will look as though they are convulsing.

The procedure is simple. First, lines 110–160 POKE the new character descriptions into memory. The new character set starts at location 2816, but we skip character 0 (which would occupy the eight bytes from 2816 to 2823) and start with character 1, the letter *A*. After we've defined the new *A*, *B*, and *C*, we do some initial printing to the graphics screen (lines 170–200), and then switch the character set at line 210. The POKE to location 4588 does this job.

Lines 220–280 print the convulsing figures. You can change the rate at which the figure changes by pressing one of the number keys. Pressing any nonnumeric key causes the program to exit from this loop. The original character set is restored in line 290.

If you want to use this feature in your own programs, remember that the computer expects to find the custom character set in bank 14. That means you will usually want to set it up at an address below 16384, since bank 14 can't see RAM above that address.

```
EQ 110 FOR J=2824 TO 2847
SG 120 READ X$:POKE J,DEC(X$)
QS 130 NEXT J
MX 140 DATA 18,18,FF,18,3C,3C,
       66,81
MK 150 DATA 98,D8,7F,18,3C,3D,
       67,80
JH 160 DATA 19,1B,FE,18,3C,BC,
       E6,01
XA 170 S=100:GRAPHIC 1,1
MB 180 CHAR ,8,3,"A CHORUS LIN
       E"
JS 190 CHAR ,7,7,"(PRESS ANY K
       EY)"
BG 200 CHAR ,5,9,CHR$(14)+"(KE
       YS 1-9 FOR SPEED)"
BM 210 POKE 4588,11
FP 220 DATA AAAAA,BBBBB,AAAAA,
       CCCCC
DH 230 IF F$="CCCCC" THEN REST
       ORE 220
MK 240 READ F$
JB 250 CHAR 1,12,5,F$
AP 260 FOR J=1 TO S:NEXT J
JP 270 GET X$:IF X$="" GOTO 23
       0
AJ 280 IF X$>="0" AND X$<="9"
       {SPACE}THEN S=(10-VAL(X
       $))*25:GOTO 230
GE 290 POKE 4588,208
XR 300 GRAPHIC CLR
```

Comparing BASIC Programs

P. Kenneth Morse

This compact utility for the IBM PC/PCjr and compatibles compares two BASIC programs line by line, reporting any differences between the two files.

If you program in BASIC on MS-DOS/PC-DOS machines, it's not uncommon to find that you have two or more copies of a BASIC program with similar filenames. How can you tell for certain which is the most recent copy? DOS provides a utility called COMP.COM, but this program reports differences in the files in terms of offsets to mismatched bytes—not particularly useful information for BASIC programmers.

"COMPARE.BAS" compares two BASIC programs even if they are of different lengths, reporting any lines in one program that are not duplicated in another. It generates a complete printout of lines that are not the same in both line number and content. The program is written for IBM BASICA. It should run without modification in PCjr Cartridge BASIC, GW-BASIC, or other PC-compatible BASICs.

Enter the program and save a copy before you run it. Also before running the program, make sure that the two programs you wish to compare are saved in ASCII (nontokenized) format. To save a program in ASCII format, append ,*A* to the end of a normal SAVE command. For example, the command SAVE *"PROGRAM"*,A saves the file named PROGRAM in ASCII form.

When you run the program, it prompts you to enter the filespec (drive, path, and filename) for each of the two files you wish to compare. At the same time, make sure that your printer is turned on and ready to print. If you have a second drive or a RAMdisk, you can speed up the comparison by placing the files on separate drives.

COMPARE.BAS

For instructions on entering this program, please refer to "COMPUTE!'s Guide to Typing In Programs" elsewhere in this issue.

```
PJ 100 'Copyright 1987
PL 110 'Compute! Publications, I
       nc.
LC 120 'All Rights Reserved.
BA 125 CLS
CP 130 PRINT TAB(30);"Copyright
       1987"
BC 140 PRINT TAB(25);"Compute! P
       ublications, Inc."
IJ 150 PRINT TAB(28);"All Rights
        Reserved."
QN 160 PRINT:PRINT
MB 190 LINE INPUT "File name of
       #1 is ",FILE$(1)
AA 200 LINE INPUT "File name of
       #2 is ",FILE$(2)
QI 210 OPEN FILE$(1) FOR INPUT A
       S #1
DL 220 OPEN FILE$(2) FOR INPUT A
       S #2
FF 230 LPRINT "File comparison:
        [1] = ";FILE$(1);"     [
       2] = ";FILE$(2)
JC 240 LPRINT
IE 250 GOSUB 400: IF L1$="Ø" THE
       N 330
II 260 GOSUB 420: IF L2$="Ø" THE
       N 390
NH 270 IF L1$=L2$ THEN 250
KB 280 L1=VAL(L1$): L2=VAL(L2$)
MJ 290 ON SGN(L1-L2) + 2 GOTO 31
       Ø,350,370
FI 300 ' Lower line number in fi
       le 1
PH 310 LPRINT "[1] ";L1$: GOSUB
       400: IF L1$ <> "Ø" THEN 2
       70
EN 320 LPRINT "[2] ";L2$
DO 330 GOSUB 420: IF L2$ <> "Ø"
       THEN 320 ELSE CLOSE: END
FC 340 ' Same line number in bot
       h files
AF 350 LPRINT "[1] ";L1$: LPRINT
        "[2] ";L2$: GOTO 250
HF 360 ' Lower line number in fi
       le 2
JK 370 LPRINT "[2] ";L2$: GOSUB
       420: IF L2$ <> "Ø" THEN 2
       70
CH 380 LPRINT "[1] ";L1$
OB 390 GOSUB 400: IF L1$ <> "Ø"
       THEN 380 ELSE CLOSE: END
IM 400 IF EOF(1)=Ø THEN LINE INP
       UT#1,L1$ ELSE L1$="Ø"
MA 410 RETURN
QO 420 IF EOF(2)=Ø THEN LINE INP
       UT#2,L2$ ELSE L2$="Ø"
ME 430 RETURN                  ©
```

News & Products

Life Simulation

Life Simulation 100 is a program which simulates the possible consequences of various choices made in real life. It is based on the ripple theory to show how a decision will affect a person's life throughout his or her lifetime. *LS100* uses mathematical probabilities and actuary tables instead of artificial intelligence to create the what-if scenarios.

Thomas Life Systems designed the program for use in school, career, and personal decision making. Its purpose is to help individuals, especially high school and college students, anticipate how decisions may affect them in the future.

Life Simulation 100 runs on the IBM PC and any compatible which supports Microsoft BASIC.

The suggested retail price is $95. Support, updates, and registration are also available for $95 per year. School discounts are available.

Thomas Life Systems, 17408 NE 19th, Bellevue, WA 98008

Circle Reader Service Number 200.

Two Disk Managers

Lassen Software has released two IBM PC disk managers. *Diskette Manager Plus, Version 1.1* automatically reads the directory from the disk and prepares a catalog. Each catalog can store information from up to 200 disks and is updated each time a disk is read. Reports using the information from the catalog can also be generated.

With *Diskette Manager Plus*, you can also print labels that contain the name of the disk, creation date, number of files, amount of disk space remaining, and up to eight lines of comments.

Diskette Manager II performs the same functions as *Diskette Manager Plus*, and it creates a database from the catalog. You can use the wildcard function, cross-reference disks, and print the results to screen.

Both disk library managers require an IBM PC, AT, XT, or compatible with 80-character display, 128K RAM memory, PC-DOS 2.X or 3.X, and two double-sided disk drives or one double-sided and one fixed drive. To print labels, a dot-matrix printer that prints eight lpi and 16 or 17 cpi is required.

Diskette Manager Plus has a price of $59.95, and *Diskette Manager II* is $79.95. An upgrade from *Plus* to *II* is available for $20. Neither disk is copy-protected.

Lassen Software, P.O. Box 1190, Chico, CA 95927

Circle Reader Service Number 201.

What Happened The Day You Were Born?

Any one born since 1901 can find out what was happening in the world on his or her birthdate with *Time Scrolls*. This database application is contained on two 400K disks and runs on the Macintosh. Using the program, you can print out the birthdate information in a scroll format, designing your own scrolls or using those on the disk. For each date you choose, you'll see the person's name, day of the week, news events for that day, news events for that year, and three other people born on the same day. There is also entertainment information such as who won an Oscar, consumer prices for that date and the present, sports news, and political information.

Time Scrolls requires a Macintosh 512, Plus, SE, or Mac II with external drive. The program is hard-disk compatible and is not copy-protected. It also requires a LaserWriter or an ImageWriter printer.

Retail price for the program is $29.95. California residents add $1.80 for sales tax. Optional preprinted forms are also available for $20 for 100 forms.

Accurate Computer Search, 993 "C" S. Santa Fe, Dept. P, Vista, CA 92083

Circle Reader Service Number 202.

Business Software For The Mac

You can be the CEO of a company in the electronic industry with Venture magazine's *Business Simulator* for the Macintosh. This game simulates the day-to-day activities and strategic planning of a large corporation. You start out with $500,000 in capital and complete control over all decision making and resources. The object is to reach $1 billion in sales over 25 years. A business journal provides information on the economy, competition, and the market. You can use the program for what-if analyses and forecasting with no risk.

Retail price of the program is $69.95. *Business Simulator* is distributed by Electronic Arts.

Reality Technologies, 3624 Market St., Philadelphia, PA 19104

Circle Reader Service Number 203.

Toy Shop Price Reduction

Brøderbund has reduced the price of its critically acclaimed *The Toy Shop* to $39.95 for the Commodore 64 version. Users of this program can customize and print out designs for 20 mechanical models, paste the designs to cardstock, and then cut out and assemble models such as an antique truck and jet plane.

Brøderbund Software, 17 Paul Dr., San Rafael, CA 94903-2101

Circle Reader Service Number 204.

Submarine War Game

You can command a World War II submarine with ActionSoft's *Up Periscope!* and your Commodore 64. There are 14 different historical scenarios for surface or submerged

ATARI® ST SYSTEMS

520ST **1040ST**

$659 COLOR **$call**

$469 MONO **$call**

OR CALL FOR LATEST PRICES

STation

The one STop shop

ORDERS ONLY

CALL TOLL FREE

1-800-CALL TRU

(1-800-225-5878)

ORDER STATUS, INFORMATION, R.A.'s

CALL 1-313-595-0788

TRUE BLUE PC-XT SYSTEM $399

100% GUARANTEED

SATISFACTION OR YOUR MONEY BACK*

- IBM* COMPATIBLE
- 360k FLOPPY DRIVE
- FLOPPY DRIVE CAR
- 256k MEMORY
- XT STYLE KEYBOARD
- 8 EXPANSION SLOTS
- 150 WATTS P.S.
- RGB/COMPOSITE MONITOR OUTPUTS
- FCC APPROVED

TURBO MODEL W/AT KEYBOARD **$429**

1200 BAUD MODEM KIT FOR ST

- ★ AVATEX 1200 HC
- ★ ST TALK SOFTWARE
- ★ COMPUSERVE STARTER KIT
- ★ RS-232 CABLE M-F

COMPLETE KIT *JUST* **$139**

3½" Continuous Label Kit

- ★ 500 3½" Disk Labels
- ★ FREE Label Program
  - Automatically Reads Disk & Makes Label
  - Create Database

All Just $9.95

ST Monitor Riser

$19.95 Reg. $49.95

★ Metal Contstruction

PC-AT SYSTEMS $995

- PC-AT COMPATIBLE
- RGB/COMPOSITE OUTPUT
- FLOPPY DRIVE CARD
- 1.2 MEG DRIVE
- AT STYLE KEYBOARD
- FLOPPY CONTROLLER
- 512k RAM
- FCC APPROVED
- 6/8 MHZ

PANASONIC 1080i

- ★ 120 CPS
- ★ EPSON/IBM
- ★ NLQ MODE
- ★ FRICTION/ TRACTOR

$179 2 YEAR WARRANTY!

BULK DISKS

LIFETIME WARRANTY!

5¼ **$.29** DS/DD QTY. 1000

$.31 DS/DD QTY. 100

3½ **$.99** SS/DD QTY 25

$1.29 DS/DD QTY. 25

DATA CASE SPECIAL

50 DS/DD DISKETTES PACKED IN A FLIP TOP DATA CASE WITH SLEEVES, LABELS, & WRITE PROTECTS

All Just $24.95 5¼"

SUPER SPECIAL SEAGATE PC-XT HARD DRIVE KIT

$349

20 MEGABYTES

Drive w W.D. Controller

FREE CATALOG!

WITH ANY $10.00 PURCHASE – COMPLETE WITH PICTURES, DESCRIPTIONS AND MORE. THOUSANDS OF ITEMS AT LOW, LOW PRICES!

ATARI ST SOFTWARE

Alternate Reality	32.95
Animal Kingdom	26.95
Arctic Fox	27.95
Black Cauldron	26.95
Chessmaster 2000	30.95
Crystal Castles	19.95
Dac Easy Accounting	46.95
Data Manager ST	52.95
DB Man	119.95
DB Master One	39.95
Degas Elite	53.95
Dollars & Sense	66.95
Donald Duck's Playground	19.95
Easy Draw	53.95
First Cadd	32.95
First Shapes	32.95
First Word	34.95
Flight Simulator II	34.95
Fraction Action	26.95
GFA Basic	52.95
GFA Basic Compiler	52.95
High Roller	32.95
I.S. Talk	32.95
Isgur Portfolio	132.95
Joust	18.95
K-Switch	25.95
Kinderama	26.95
Kings Quest I, II or III	32.95
Lattice C	99.95
Leader Board	26.95
Mark Williams C	119.95
Math Talk	32.95
Music Studio	39.95
Phantasie	26.95
Phantasie II	26.95
PrintMaster Plus	26.95
Psion Chess	39.95
Publishing Partner	99.95
S.D.I.	32.95
Silent Service	26.95
Skyfox	31.95
Space Quest	32.95
Speller Bee	32.95
Swiftcalc ST	52.95
Tenth Frame	26.95
Time Bandits	26.95
VIP Professional	166.95
WordWriter ST	52.95

ST HARDWARE

Atari 1040ST	Call
Atari 314 Drive	199.00
Atari 354 Drive	105.00
Atari 520ST	259.00
Atari SC1224 Color Monitor	299.00
Atari SH204 Hard Drive	539.00
Atari SM124 Mono Monitor	119.00
Atari SMM804 Printer	169.00
Atari SX212 Modem	79.00
Supra 20 Meg Hard Drive	549.00
Upgrade, EZ-Ram	159.00
Z-Time 1040	49.95
Z-Time 520	39.95

PRINTERS

Okimate 20 w/Plug-N-Print	198.00
Panasonic KX-P1080	179.00
Panasonic KX-P1091i	249.00
Panasonic KX-P1092	309.00
Panasonic KX-P1592	399.00
Panasonic KX-P1595	435.00
Panasonic KX-P3151	259.00
Panasonic KX-P3131	409.00

DUST COVERS

1040 ST	9.95
520 ST	6.95
520 ST System	24.95
Gemini 10	6.95
IBM PC Keyboard	6.95
IBM PC System	23.95
Panasonic 1080/90/91	6.95
SF 314	5.95
SF 354	5.95
SM 1224	11.95
SM 124	11.95

COLOR PAPER

BLUE, YELLOW, GREEN, PINK
WHITE, SALMON & GOLD
500 Sheets (Any Color)

20# LZ Cut, 9.5x11	6.95
Rainbow Pack #1 (P-Y-B)	6.95
Rainbow Pack #2 (Gr-S-Go)	6.95

CONTINUOUS CARD STOCK

3 X 5	4.99
4 X 6	7.99
5 X 7	9.99

CONTINUOUS LABELS

Audio Cassette, White	9.99
Disk (2.75 X 2) 250 White	4.99
File Folder (3½ X 5/8) 750	9.99
Mail (3½ X 15/16) 1000	6.99

AB Switch Box Parallel	39.95
AB Switch Box RS-232	39.95
AB-X Crossover Box Par	49.95
IBM/ST Printer Cable 10'	9.95
IBM/ST Printer Cable 6'	6.95
Microstuffer	59.95
P R Connection	79.95
RS-232 F/F 6'	9.95
RS-232 M/F 10'	12.95
RS-232 M/F 6'	9.95
RS-232 M/M 6'	9.95
ST Drive Cable 6'	14.95
XE Serial 6'	9.95

MISCELLANEOUS ACCESSORIES

14" Twist-N-Tilt	14.95
5.25" Disk Notcher	7.95
Amaray Printer Stand	14.95
Mouse House	4.95
Mouse Mat	7.95
Printer Stand 2-Pc.	12.95
Six Outlet Surge Strip	14.95

DISKETTES

Bulk 3.5" SS/DD	.99
Bulk 3.5" DS/DD	1.29
Bulk 5.25" DS/DD	.39
BASF 3.5" SS/DD	13.95
BASF 3.5" DS/DD	19.95
BASF 5.25" DS/DD	9.95
BASF 5.25" DS/HD	19.95
Maxell 3.5" SS/DD	14.95
Maxell 3.5" DS/DD	22.95
Maxell 5.25" SS/DD	10.95
Maxell 5.25" DS/DD	11.95
Maxell 5.25" DS/HD	23.95
Sony 3.5" SS/DD	14.95
Sony 3.5" DS/DD	22.95
Sony 5.25" SS/DD	8.95
Sony 5.25" DS/DD	9.95
Sony 5.25" DS/HD	22.95
3M 3.5" SS/DD	15.95
3M 3.5" DS/DD	22.95
3M 5.25" SS/DD	10.95
3M 5.25" DS/DD	11.95
3M 5.25" DS/HD	29.95

DISK STORAGE

Disk Directory (3.5")	22.99
Disk File 40 - 3.5"	6.99
Disk File 60 - 5.25"	6.99
Library Case-3 (5.25")	.99
Library Case-5 (3.5")	.99
Pocket Pack (3.5")	14.99

RIBBONS

Epson FX/MX/RX-80	5.99
IBM Proprinter	9.95
Panasonic 1080/90i	9.99
Star/Okidata	2.99
XMM 801/Legend/BMC	7.99
SMM 804	7.99

IBM HARDWARE

PC-XT

1200 Baud Internal Modem	119.00
20 Meg Seagate Kit	389.00
C.G.V. Card	59.95
Color Graphic/Printer Card	75.95
Disk Drive Card	35.95
DS/DD ½ Ht. 360k Drive	89.95
E.G.A. Card	299.00
Game Controller Port	29.95
Hard Disk Control	110.00
I/O Plus II Card	75.95
Keyboard - AT Style	75.00
Keyboard - RT Style	95.00
Keyboard - XT Style	54.00
Main Board - Turbo	149.00
Main Board	116.00
Monographics Card	75.95
Multifunction Card for PC/XT	75.95
Multi I/O Card	86.95
Printer Card	29.95
RS-232 Card	38.95

PC-AT

1.2MB Drive Card	CALL
3 MB Multifunction Card	CALL
4 MB Ram Card	CALL
Hard Disk Controller	CALL
HD/Floppy Card for AT	CALL
Main Board	CALL
Serial Parallel Card	CALL

THOMSON MONITORS

4160 CGA Color	339.00
450A Monochrome - 132 CLM	159.00
4350 EGA Dual Scan	469.00
4375M Ultra Scan	629.00
TTL Monitor	119.00

CONTROLLERS

Atari Econorny	4.95
Epyx 500XJ	14.95
Replacement Mouse (ST)	49.95
Tac 2	8.95
TAC-10 IBM & APPLE	27.95
Wico 3-Way	23.95
Wico Boss	12.95

IBM SOFTWARE

Alternate Reality, City	26.95
Archon	10.95
Balance of Power	33.95
Black Cauldron	26.95
Championship Lode Runner	23.95
Chessmaster 2000	27.95
Clip Art Vol. 1-Newsroom	19.95
Clip Art Vol. 2-Newsroom	26.95
Clip Art Vol. 3-Newsroom	19.95
Crosstalk	129.95
Dbase II	333.95
Dbase III Plus	466.95
Dollars & $ense w/Forecast	120.95
Donald Duck's Playground	19.95
F-15 Strike Eagle	23.95
Flight Simulator	33.95
Gato	26.95
Graphics Library I or II	23.95
I.S. Talk	39.95
Jet	33.95
Karateka	23.95
King's Quest I, II or III	33.95
Lotus 1-2-3	333.95
Mark Williams C	333.95
Mastertype	26.95
Math Blaster	33.95
Mean 18	26.95
Newsroom	40.95
Pawn	29.95
Print Shop	40.95
Print Shop Companion	33.95
Radio Baseball	27.95
Sargon III	26.95
Shanghai	29.95
Sidekick	43.95
Silent Service	23.95
Space Quest	33.95
Summer Games II	26.95
Symphony	466.95
Temple of Apshai Trilogy	26.95
Ultima III	39.95
Winnie The Pooh	16.95
Winter Games	26.95
Wizardly Proving Grounds	40.95
Word Attack	33.95
Word Perfect	330.95
Wordstart 2000 +	396.95
World Games	26.95
Zork Trilogy	46.95

No surcharge for MasterCard or Visa

Ordering Information: Credit Card – Have number, expiration date, shipping address and **DAY** phone number ready with order. Personal and company checks: Call order in – you will receive an order number, send order number and check to the P.O. Box listed. Allow 14 Business Days for check to clear. Money Orders: Call order in – you will receive an order number, send number with money order to P.O. Box listed. Shipping: Continental USA – **All Orders add 3%,** minimum $3.00. HI, AK, Canada add 5%, minimum $5.00. All other order areas add 15% minimum $10.00. No C.O.D.'s over $100.00, add an additional $3.00 shipping charges for C.O.D. All areas not serviced by U.P.S. will be shipped first class insured mail. All shipping charges in excess of the above terms will be added. All goods are new and include factory warranty. **All defectives must have a return authorization number.** Please call (313)595-0788 to obtain an R.A. # or return will not be accepted. Prices and availability subject to change without notice. All correspondence should be sent to P.O. Box 75 – Wayne, MI 48184. * SATISFACTION REFUND – IS LESS SHIPPING & HANDLING – Call for Details.

Up Periscope!, *a World War II submarine simulation from ActionSoft.*

patrol. Your targets include freighters, tankers, troop ships, destroyers, and battleships. The fleet-class sub is equipped with Mark 10 and Mark 18 torpedoes, surface and attack radars, a variable-power periscope, complete instrument panel, and ocean charts.

Up Periscope! is available on disk only and requires one disk drive and either a monochrome or a color monitor.

Retail price is $29.95.

ActionSoft, 201 W. Springfield Ave., Ste. 711, Champaign, IL 61820

Circle Reader Service Number 205.

Alien Action Game

Energy is the most important commodity in Firebird's *The Sentry.* In this new game, the player and a robot try to defeat the Sentry and her Landgazers by absorbing their alien energy while at the same time maintaining their own energy levels. Once the player becomes ruler of the landscape, he or she moves on to another of the 10,000, three-dimensional landscapes for more advanced play.

The Sentry is available for the Commodore 64. The package includes a booklet, quick-key guide, and a pin-on button. The suggested retail price is $39.95.

Firebird Licensees, P.O. Box 49, Ramsey, NJ 07446

Circle Reader Service Number 206.

ST Products From MichTron

MichTron has released several new products for the Atari ST. *GoldRunner* is a futuristic game in which humanity deserts Earth and searches for another inhabitable planet. However, to reach a new planet, you must battle the Tritons of the Ring Worlds, a highly technical and violent race. Your computer is your companion, giving instructions as you go. Graphics and an original music score are other features of this game.

Retail price for *GoldRunner* is $39.95.

GFA-Vector is a three-dimensional graphics program to be used with the *GFA BASIC* interpreter. Using this program, you can create objects using coordinates or the graphics editor and then add the objects to other programs. The pictures are created in machine language for speed and can be revolved along the axes in one-degree increments. You can store up to 32 objects in one picture file.

GFA-Vector is available for $49.95 and requires the GFA-BASIC interpreter.

Make It Move is a graphics animation program designed for use with a paint program to create screen movement. The program is controlled by the mouse and is compatible with paint programs such as *NEOchrome* and *DEGAS.* You move predrawn objects from one picture to another using the objects program. With the script function, you can display the graphics and pictures in a sequence, cutting, fading, repeating the object, or scrolling it over another object.

Retail price for *Make It Move* is $49.95.

Twenty-one different utility programs are combined into one package with *STuff.* This package includes utilities such as a GEM-program autoboot, caps-lock key lock, a file security program, a text file search utility, hexadecimal display for binary data and program files, date and time set for selected files, disk-write speed-up, and four utilities for the auto folder programs.

STuff has a retail price of $39.95.

MichTron, 576 S. Telegraph, Pontiac, MI 48053

Circle Reader Service Number 207.

IBM Educational Software

Gamco Industries has released two new educational packages for the IBM PC. *Telling Time* combines an arcade with drills in learning to tell time. Students choose from four different time-telling lessons and four levels of difficulty. They have three tries to answer each problem correctly before the correct answer is displayed. At the end of the lesson, the student receives a performance summary and, if he or she scored a certain percentage, plays an arcade game.

The teacher can change the number of problems in a lesson and the percentage needed to play the game. The program holds up to 200 students names and their scores.

In *Money,* there are also four different lessons with four difficulty levels. The lessons in this package deal with money—pennies, nickels, dimes, quarters, and half-dollars. If the student achieves a set percentage of correct answers, he or she has the opportunity to play an arcade-style game.

Teachers can store up to 200 students' names and print all or individual files.

The IBM versions of both *Telling Time* and *Money* require a graphics card and 256K memory. Versions are also available for the Commodore 64 and Apple II series with 48K. Backup disks and class packages are also available.

Retail price for all versions is $44.95.

Gamco Industries, Box 1911, Big Spring, TX 79721

Circle Reader Service Number 208.

PC Accelerator

Prism Electronics has announced an accelerator board for the IBM PC XT and compatibles. PC-Bandit does not require an expansion slot and resides in the PC's current 8284 clock-chip position. A lead is then connected to the DMA chip and another lead is connected to the motherboard for software speed selection. The board uses software instead of a switch to toggle between speeds.

A disk can be formatted using the accelerated mode, and the mode does not interfere with the realtime clock.

PC-Bandit can boost the PC's processing speed by up to 60 percent. An additional 8 megahertz can be purchased for a total boost of 280 percent.

PRINTER PACKAGE SPECTACULAR

ALL PRINTER PACKAGES INCLUDES - NO SURCHARGES ON CREDIT CARDS - FREE SHIPPING IN CONTINENTAL USA
APPROPRIATE INTERFACE DEVICE FROM THE FAIR MAIL ORDER COMPANY

ComputAbility
Consumer Electronics

Panasonic

PANASONIC 1080i PRINTER PKG. WITH APPROPRIATE INTERFACE DEVICE

IBM PC/PARALLEL CABLE	$199
ATARI ST/PARALLEL CABLE	$199
AMIGA/PARALLEL CABLE	$199
C-64/128/XETEC SUPER GRAPHIC	$239
ATARI 8-BIT/SUPRA 1150	$229
APPLE 2E/PRO GRAPPLER	$269

NO SURCHARGES ON CREDIT CARDS
FREE SHIPPING IN CONTINENTAL USA

PANASONIC 1091i PRINTER PKG WITH APPROPRIATE INTERFACE DEVICE

IBM PC/PARALLEL CABLE	$259
ATARI ST/PARALLEL CABLE	$259
AMIGA/PARALLEL CABLE	$269
C-64/128/XETEC SUPER GRAPHIC	$309
ATARI 8-BIT/SUPRA 1150	$299
APPLE 2E/PRO GRAPPLER	$339

NO SURCHARGES ON CREDIT CARDS
FREE SHIPPING IN CONTINENTAL USA

PANASONIC 1092i PRINTER PKG WITH APPROPRIATE INTERFACE DEVICE

IBM PC/PARALLEL CABLE	$329
ATARI ST/PARALLEL CABLE	$329
AMIGA/PARALLEL CABLE	$339
C-64/128/XETEC SUPER GRAPHIC	$369
ATARI 8-BIT/SUPRA 1150	$359
APPLE 2E/PRO GRAPPLER	$399

NO SURCHARGES ON CREDIT CARDS
FREE SHIPPING IN CONTINENTAL USA

star

NX-10 PRINTER PKG WITH APPROPRIATE INTERFACE DEVICE

IBM PC/PARALLEL CABLE	$209
ATARI ST/PARALLEL CABLE	$209
AMIGA/PARALLEL CABLE	$219
C-64/128/XETEC SUPER GRAPHIC	$259
ATARI 8-BIT/SUPRA 1150	$249
APPLE 2E/PRO GRAPPLER	$279

NO SURCHARGES ON CREDIT CARDS
FREE SHIPPING IN CONTINENTAL USA

NX-15 PRINTER PKG WITH APPROPRIATE INTERFACE DEVICE

IBM PC/PARALLEL CABLE	$349
ATARI ST/PARALLEL CABLE	$349
AMIGA/PARALLEL CABLE	$359
C-64/128/XETEC SUPER GRAPHIC	$389
ATARI 8-BIT/SUPRA 1150	$379
APPLE 2E/PRO GRAPPLER	$409

NO SURCHARGES ON CREDIT CARDS
FREE SHIPPING IN CONTINENTAL USA

NB-24/10 PRINTER (24 WIRE) PKG WITH APPROPRIATE INTERFACE DEVICE

IBM PC/PARALLEL CABLE	$459
ATARI ST/PARALLEL CABLE	$459
AMIGA/PARALLEL CABLE	$469
C-64/128/XETEC SUPER GRAPHIC	$499
ATARI 8-BIT/SUPRA 1150	$489
APPLE 2E/PRO GRAPPLER	$529

NO SURCHARGES ON CREDIT CARDS
FREE SHIPPING IN CONTINENTAL USA

commodore

64C COMPUTER
1541C DISK DRIVE
1802C MONITOR

C-128 COMPUTER
1571 DISK DRIVE
1902 MONITOR

CALL FOR INDIVIDUAL & SUPER PACKAGE PRICES

1750 RAM EXPANDER
1764 RAM EXPANDER
1351 MOUSE
AVATEX 1200 HC MODEM
AVATEX 2400 HC MODEM

CALL FOR CURRENT PRICES

COMMODORE PC-10
PC COMPATIBLE COMPUTER

CALL FOR CURRENT PRICES

AMIGA

COMPUTER SYSTEM
CALL FOR CURRENT PRICES & AVAILABILITY

EPSON®

PRINTER PACKAGES
PRICES TO LOW TO ADVERTISE

ATARI

520 ST & 1040 ST
SYSTEM PACKAGES
CALL FOR LASTEST PRICES

IBM

Amnisia	29.95
Art Gallery 1	24.95
Art Gallery2	18.95
Balance of Power	31.95
Bop-N-Wrestle	19.95
Certificate Maker	38.95
Championship Baseball	25.95
Championship Golf	25.95
Chessmaster 2000	25.95
Clip Art 1,3	19.95
Clip Art 2	25.95
Commando	25.95
Conflict in Vietnam	25.95
Crusade in Europe	25.95
Decision in the Desert	25.95
Defender of the Crown	25.95
Destroyer	25.95
F-15 Strike Eagle	21.95
Flight Simulator II	38.95
Grand Slam Bridge	39.95
Graphics Library 1 or 2	21.95
Hitchhikers Guide to the Galaxy	19.95
Hollywood Hijinx	25.95
Infiltrator	19.95
Isqur Portfolio	149.95
Jet	34.95
Kings Quest I, II, or III	31.95
Leather Goddesses of Phobos	25.95
Newsmaster	62.95
Newsroom	37.95
One on One	10.95
Orbiter	31.95
Pawn	29.95
P.C. Powerpack	77.95
Phantasie	25.95
Police Quest	CALL
Printmaster	36.95
Print Shop	37.95
P.S. Companion	31.95
Radio Baseball	25.95
Roadwar 2000	25.95
Seven Cities of Gold	10.95
Shanghai	22.95
Silent Service	21.95
Space Quest	31.95
Starflight	32.95
Sub Battle	25.95
Sublogic Baseball	34.95
Sublogic Football	34.95
Tag Team Wrestling	25.95
Tenth Frame	29.95
Two on Two Basketball	25.95
Universe II	47.95
Where in the U.S.A is Carmen Sandiego	29.95
Where in the World is Carmen Sandiego	25.95
World Tour Golf	32.95

C64

Bard's Tale	25.95
Bard's Tale 2	25.95
Basic Compiler 64	25.95
Cad Pak 64	25.95
Carmen U.S.A.	29.95
Certificate Maker	30.95
Champ. Baseball	19.95
Chart Pak	25.95
Chessmaster 2000	25.95
Copy II 64	27.95
Data Manager II	25.95
Defender of the Crown	22.95
Desk Pak I	22.95
Desk Pak 2	32.95
Famous Courses	14.95
Fast Load	22.95
Final Cartridge	39.95
Flight Simulator II	33.95
Font Master II	29.95
Geocalc	34.95
Geo Chart	27.95
Geodex	25.95
Geofile	34.95
Geopublish	34.95
Geos	37.95
Geos Inside & Out	16.95
Gunship	22.95
Hardball	18.95
Hollywood Hijinx	22.95
Leaderboard	24.95
Leather Goddess	22.95
Legacy/Ancients	20.95
Mail Order Monster	10.95
Mind Mirror	10.95
Movie Maker	10.95
Newsroom	30.95
Partner 64	29.95
Pawn	25.95
Pegasus	23.95
Pirates	25.95
Printmaster Plus	21.95
Printshop	28.95
P.S. Companion	22.95
S.D.L	22.95
Skyfox	10.95
Starglider	25.95
Sub Battle	22,95
ST. Sports Baseball	25.95
ST. Sports Basketball	25.95
Super C Compiler	39.95
Super Pascal	39.95
Super Pak 2/64	59.95
Super Star Hockey	22.95
Swiftcalc Sideways	25.95
2 on 2 Basketball	22.95
Ultima 1	25.95
War in S. Pacific	39.95
Wordwriter/Spell	32.95
Wld Class Leader	25.95
World Tour Golf	23.95

C128

Abacus Books	CALL
Basic Compiler 128	39.95
Bureaucracy	22.95
Cad Pak 128	39.95
Chart Pak	25.95
Cobol 128	39.95
Datamanager128	42.95
Fleet System 4	46.95
Fontmaster 128	36.95
Goes 128	CALL
Mach 128	32.95
Merlin 128	69.95
Multplan 128	27.95
Paperclip II/128	CALL
Partner 128	42.95
Perfect Writer	49.95
Pocket Filer 2	36.95
Pocket Planner 2	36.95
Pocket Writer 2	36.95
Speed Term 128	25.95
Superbase 128	57.95
Super C Compiler	39.95
Super Pak2/128	59.95
Super Pascal 128	39.95
Superscript 128	46.95
Swiftcalc 128	42.95
Sylvia Porter 128	42.95
Trinity	25.95
Visastar 128	54.95
Visawrite 128	54.95
Wordwriter 128	42.95

ATARI 8-BIT

Animation Station	59.95
Archon	10.95
Archon II	10.95
Atariwriter Plus	39.95
Bop-N-Wrestle	19.95
Chessmaster 2000	25.95
F-15 Strike Eagle	22.95
Gettysburg	39.95
Hardball	18.95
Home Planetarium	27.95
Leaderboard	24.95
Leather Goddess	22.95
Mail Order Monsters	10.95
Megafont II	16.95
Movie Maker	10.95
Music Studio	22.95
Page Designer	18.95
Print Shop	28.95
P.S. Companion	22.95
P.S. Interface	18.95
Racing Destruction	10.95
Rubber Stamp	18.95
Silent Service	22.95
Syncalc	32.95
Synfile	32.95
Top Gunner	16.95
Typesetter	21.95
Ultima II	31.95
Ultima IV	38.95
Wizards Crown	27.95
Zork Trilogy	39.95

APPLE

Airheart	22.95
Animate	44.95
Autoduel	32.95
Bards Tale	29.95
Bards Tale II	32.95
Bop-N-Wrestle	19.95
Certificate Maker	31.95
Certificate Maker Lib #1	22.95
Clip Art 1 or 3	19.95
Clip Art 2	25.95
Destroyer	25.95
F-15 Strike Eagle	22.95
Fight Night	21.95
Graphics Lib 1,2or3	16.95
Hardball	21.95
Infiltrator	19.95
Jet	27.95
Newsroom	37.95
On Balance	24.95
Pawn	29.95
Pegasus	25.95
Phantasie 1,2 or 3	25.95
Print Shop	31.95
Silent Service	22.95
Starglider	29.95
Street Sports Baseball	25.95
Street Sports Basketball	25.95
Tag Team Wrestling	22.95
Where in the U.S.A. is Carmen Sandiego	29.95
Where in the World is Carmen Sandiego	25.95

ATARI ST

Abacus Books	CALL
Aegis Animator	48.95
Art Gallery 1or2	18.95
Artic Fox	26.95
Assempro	39.95
Back-up	24.95
Balance of Power	32.95
Certificate Maker	31.95
Chessmaster 2000	32.95
Colonial Conquest	25.95
CZ Droid	69.95
Dac Easy Accounting	44.95
Dac Payroll	32.95
Datamanager ST	48.95
Datatrieve	32.95
DB Man	96.95
Deep Space	29.95
Defender of the Crown	32.95
Degas Elite	52.95
Desk Cart	72.95
Easy Draw	48.95
EZ Track	39.95
Flash	27.95
Flight Simulator	33.95
GFA Basic	48.95
GFA Compiler	48.95
GFA Draft	52.95
GFA Vector	36.95
Gold Runner	24.95
Hardball	27.95
Karate Kid II	24.95
Kid Talk	31.95
Labelmaster Elite	27.95
LDW Basic	44.95
Major Motion	24.95
Mark Williams	114.95
Math Talk	31.95
Megafont ST	24.95
Music Studio	32.95
Pawn	29.95
Personal Pascal	49.95
Phantasie 1,2or3	25.95
PM Interface	18.95
Printmaster Plus	24.95
Publishing Partner	99.95
Roadwar 2000	25.95
Rubber Stamp	24.95
S.D.L	32.95
Shanghai	25.95
Silent Service	25.95
Skyfox	29.95
Speller Bee	31.95
Starglider	29.95
Sundog	24.95
Swiftcalc ST	48.95
Thunder	26.95
Time Bandits	24.95
Typesetter Elite	31.95
Wordwriter ST	48.95
Write 90	18.95

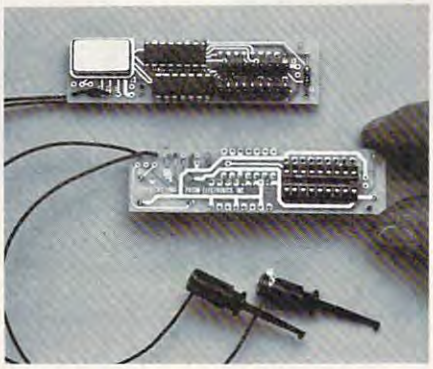
AMIGA

Aegis Animator	84.95
Aegis Draw	159.95
Analyze 2.0	94.95
Bards Tale	32.95
Bureaucracy	25.95
Cambridge Lisp	129.95
CBM Textcraft	69.95
Championship Baseball	29.95
Championship Golf	25.95
Chessmaster 2000	29.95
Defender of the Crown	32.95
Deja Vu	32.95
Deluxe Music 2.0	64.95
Deluxe Print II	84.95
Deluxe Paint	64.95
Deluxe Video 2.0	84.95
Earl Weaver Baseball	32.95
Fairy Tale Adventure	31.95
Firat Shapes	31.95
Flight Simulator II	34.95
Flip Side	31.95
Fortran 77	169.95
GFL Football	29.95
Goldspel	27.95
Grand Slam Tennis	31.95
Grid Iron	44.95
Impact	54.95
Instant Music	32.95
Kampgruppe	39.95
Kid Talk	31.95
Kinderama	31.95
Kings Quest 1,2 or 3	32.95
Logistix	89.95
Marble Madness	32.95
Math Talk	31.95
Mean 18	27.95
Microleague Baseball	39.95
Modula 2	64.95
Music Studio	32.95
On Line	42.95
Organize	62.95
Page Setter	89.95
Pawn	29.95
Phantasie	25.95
Print Master Plus	31.95
Publisher	119.95
Scribble	62.95
S.D.L	32.95
Shanghai	25.95
Silent Service	25.95
Sinbad	32.95
Skyfox	26.95
Sonix	54.95
Starglider	29.95
Superbase Personal	89.95
Tass Times	25.95
Two on Two Basketball	29.95
VIP Professional	169.95
Winter Games	25.95
World Games	25.95

SINCE 1982
ComputAbility
Consumer Electronics
P.O.BOX 17882, MILWAUKEE, WI 53217
ORDER LINES OPEN
Mon.-Fri. 9a.m.-9p.m. CST Sat.11a.m.-5p.m. CST
To Order Call Toll Free
800-558-0003
For Technical Info, Order Inquiries, or for Wisc. Orders
414-357-8181
TELEX NUMBER 9102406440
(ANSERBACK = COMPUT MILW UQ)

No surcharge for Mastercard or Visa

ORDERING INFORMATION: Please **specify system.** For fast delivery send cashier's check or money order. Personal and company checks allow 14 business days to clear. School P.O.'s welcome. **C.O.D. charges are $3.00.** In Continental U.S.A. include $3.00 for software orders. 5% shipping for hardware, minimum $5.00. MasterCard and Visa orders please include: card #, expiration date and signature. WI residents please include 5% sales tax. HI, AK, FPO, APO, Puerto Rico and Canadian orders, please add 6% shipping, **minimum $6.00.** All other **foreign orders add 15% shipping, minimum $15.00.** All orders shipped outside the Continental U.S.A. are shipped first class insured U.S. mail. If foreign shipping charges exceed the minimum amount, you will be charged the additional amount to get your package to you quickly and safely. All goods are new and include factory warranty. Due to our low prices all sales are final. **All defective returns must have a return authorization number.** Please call (414) 357-8181 to obtain an R.A. # or your return will not be accepted. Prices and availability subject to change without notice.



PC-Bandit from Prism Electronics.

There are two versions of PC-Bandit—7.4 MHz for the 150 nanosecond chip machines and a slower, 6.7 MHz for machines with 200 nanosecond chips.

Installation takes about 30 minutes unless the 8284 is soldered in place, in which case assistance may be necessary.

Retail price is $69.95. The 8MHz boost retails for an additional $19.95.

Prism Electronics, 14682 NE 95th St., Redmond, WA 98052
Circle Reader Service Number 209.

More Aegis Amiga Software

To add to Aegis' line of Amiga software such as the *Aegis Images* paint system and the *Aegis Sonix* music and MIDI software, the company also has introduced *Diga!*, desktop communications software for VT-52, VT-100 and Tektronix terminal emulation and for telecommunications at 300, 1200, or 2400 baud; and the *Amiga Desktop Design System*, composed of *Aegis Draw Plus* design software for the Amiga used with the Roland DXY-980 or DXY-880 series plotter.

Aegis Development, 2115 Pico Blvd., Santa Monica, CA 90403
Circle Reader Service Number 210.

New Action Software

Paragon Software's first entertainment package, *Master Ninja: Shadow Warrior of Death*, is a graphics-intensive action game for IBM PCs and compatibles available immediately and this fall for Commodore 64, Amiga, and Atari ST computers ($29.95–$34.95).

The company also has an accounting package for the Commodore 128, *Cash In-Cash Out* ($69.95), and plans to have more computer games available in late 1987 and early 1988.

Paragon Software, Plymouth Center, 521 Plymouth St., Greensburg, PA 15601
Circle Reader Service Number 211.

Avalon Hill's *Darkhorn, NBA*

With *Darkhorn*, Avalon Hill provides both strategic and arcade action in a high-fantasy setting as up to four players challenge the Darklord's dominance of the world. The game is available for Apple II and Commodore machines, and is priced at $30.

Licensed by the National Basketball Association, Avalon Hill's *NBA* is an animated, statistical basketball package that provides players with the chance to recreate 20 historical teams, with rosters including Larry Bird, Wilt Chamberlain, Michael Jordan, and Bob Cousey. The game can be played solitaire, two-person, or in autoplay mode. It is available for Commodore, the Apple II family, and PC compatibles for $39.95.

The Avalon Hill Game Co., 4517 Hartford Rd., Baltimore, MD 21214
Circle Reader Service Number 212.

Amiga Expansion

For the Amiga, Byte By Byte has announced a $99.95 powerful solid modeling and ray-tracing image production package, *Sculpt 3-D*; the Advantage 500, an Amiga memory expansion system that comes with 512K of RAM, and is user expandable to two megabytes of RAM ($249–$599, depending on amount of RAM); and the PAL JR, a two-slot, fully Zorro compatible autoconfigure expansion system and 20-meg hard disk for the Amiga, priced at $1,495.

Byte By Byte, Arboretum Plaza II, 9442 Capital of Texas Highway North, Suite 150, Austin, TX 78759
Circle Reader Service Number 213.

Apple *PrintMaster Plus*

The new Apple version of the *PrintMaster Plus* printing and graphics package has been announced by Unison World. Priced at $49.95, the package is also available as an update for $15 to current users of *PrintMaster, Print Shop, Stickybear Printer*, or similar programs.

Unison World, 2150 Shattuck Ave., Suite 902, Berkeley, CA 94704
Circle Reader Service Number 214.

Low-Cost Educational Programs

For $8.99 each, Vision Software offers a variety of budget-priced educational programs for PC-compatible, Apple, and Commodore 64 computers. The collection of programs, an educational series that's been used in the Arkansas school system, includes math, English grammar and composition, geography, and many other topics.

Vision Software, 5400 Taylor Rd., Suite 108, Naples, FL 33942
Circle Reader Service Number 215.

Tandy Expansion

This company offers a variety of user-installable expansion boards for the Tandy 1000, 1000SX, and 3000 PC-compatible computers; the latest board is a Hayes-compatible 1200-baud internal modem for $129.

Zuckerboard/ATD, 235 Santa Ana Ct., Sunnyvale, CA 94086
Circle Reader Service Number 216.

Budget Entertainment Software

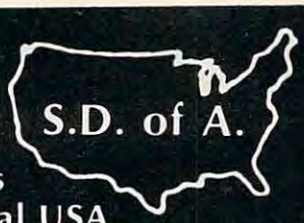
In addition to a growing line of budget—$9.95 range—entertainment software, Constellation Software has announced *Quiet Riot: The Silencer*, an earphone system for Commodore computers including Amiga. The earphones are being advertised as "Mom's Delight."

Constellation Software, 1300 N. Hagan St., Champaign, IL 61820
Circle Reader Service Number 217.

Passport Supports PC Music

Passport has announced three programs supporting IBM's new PC Music Feature, an eight-voice FM sound synthesizer and MIDI interface on a single card for under $500. The programs include *MIDIsoft Studio* ($149.95), a 32-track MIDI sequencer; the *Passport MIDI Voice Editor* ($149.95), a music editor; and

SOFTWARE DISCOUNTERS OF AMERICA



For Orders Only—1-800-225-7638
PA Orders—1-800-223-7784
Customer Service 412-361-5291

- Free shipping on orders over $100 in continental USA
- No Surcharge for VISA/MasterCard
- Your card is not charged until we ship

IBM SOFTWARE

ACCESS
10th Frame Bowling ... $29
ACCOLADE
Mean 18 Golf ... $30
Mean 18 Famous Course Disk ... $14
ACTIVISION
Hacker ... $9.88
Hacker 2 ... $24
Portal ... $27
Shanghai ... $23
Tass Times ... $23
ARTWORX
Bridge 5.0 ... $19
Strip Poker ... $25
Data Disk #1 Female ... $14
Data Disk #2 Male ... $14
Data Disk #3 Female ... $14
Linkword Languages:
Dutch ... $19
French ... $19
German ... $19
Italian ... $19
Russian ... $19
Spanish ... $19
Miniature Golf ... $19
AVALON HILL
Dr. Ruth's Game ... $19
Super Bowl Sunday ... $21
S.B.S. Champs Disk ... $14
S.B.S. '85 Team Disk ... $14
BRODERBUND
Ancient Art of War ... $27
Bank St. Writer Plus ... $49
Carmen Sandiego USA ... $29
Carmen Sandiego World ... $25
Karateka ... $21
Print Shop ... $36
P.S. Companion ... $32
P.S. Graphics Holiday Edition ... $21
P.S. Graphics Library #1 or #2 .. $21 Ea.
Toy Shop ... $32
Type! ... $30
CBS
Success with Algebra .. Call
Success with Math ... Call
CDA
Mr. Boston Bartender's Guide .. $9.88
America Cooks Series:
American ... $9.88
Chinese ... $9.88
French ... $9.88
Italian ... $9.88
Mexican ... $9.88
CENTRAL POINT
Copy 2 ... $23
PC Tools ... $23
DATA EAST
Commando ... $24
Tag Team Wrestling ... $24
DAVIDSON
Grammar Gremlins ... $30
Homeworker ... $59
Math Blaster ... $30
Spell It ... $30
Word Attack ... $30
ELECTRONIC ARTS
Software Classic Series:
Archon ... $9.88
Mind Mirror ... $9.88
Music Const. Set ... $9.88
One-on-One ... $9.88
Pinball Const. Set ... $9.88
Seven Cities Gold ... $9.88
Super Boulder Dash .. $9.88
ELECTRONIC ARTS
Amnesia ... $29
Arctic Fox ... $23
Chessmaster 2000 ... $26
Chuck Yeager's Advanced Flight Simulator ... $26
Empire ... $32
Grand Slam Bridge ... $39
Marble Madness ... $23
Murder Party ... $26
Scrabble ... $26
Starfleet 1 or 2 ... $32 Ea.
Starflight ... $32
World Tour Golf ... $32
EPYX
Create A Calendar ... $19
Pitstop 2 ... $9.88
P.S. Graphics Scrapbook:
#1 Sports ... $16
#2 Off the Wall ... $16
#3 School ... $16
Sub Battle Simulator .. $24
Summer Games 2 ... $24
World Games ... $24
GAMESTAR
Champ. Baseball ... $24
Champ. Basketball ... $24
Champ. Golf ... $24
GFL Ch. Football ... $24
HAYDEN
Sargon 3 Chess ... $25
HI-TECH EXPRESSIONS
Award Ware ... $9.88
Card Ware ... $6.88
Heart Ware ... $6.88
Party Ware ... $9.88
INFOCOM
Bureaucracy ... $24
Hitchhikers Guide ... $24
Hollywood Hijinx ... $24
Leather Goddesses ... $24
Stationfall ... $24
The Lurking Horror ... $24
Zork 1 ... $24
Zork Trilogy ... $44
MASTERTRONIC
Five a Side Soccer ... $6.88
Ninja ... $6.88
Pro Golf ... $6.88
MECA
Andrew Tobias: Managing Your Money 2.0 ... $129
MICROLEAGUE
Baseball ... $24
*General Mgr./Owner .. $19
*1986 Team Disk ... $12
*Add-on programs!
MICROPROSE
Crusade in Europe ... $24
F-15 Strike Eagle ... $21
Silent Service ... $21
MINDSCAPE
American Challenge .. $24
Balance of Power ... $30
Bop & Wrestle ... $19
Crossword Magic ... $30
Defender of the Crown $24
Infiltrator ... $19
Perfect Score SAT ... $44
Sub Mission ... $24
Super Star Ice Hockey . $24
PERSONAL CHOICE
Term Paper Writer ... $32
SIERRA
Black Cauldron ... $24
King's Quest 1, 2, or 3 ... $30 Ea.
Space Quest ... $30
SIMON & SHUSTER
Chem Lab ... $24
Star Trek: Kobayashi Alternative ... $24
Promethean Prophecy . $24
Typing Tutor 3 ... $30
Websters:
Spelling Checker ... $39
Thesaurus ... $44
Wine Cellar ... $44
SPRINGBOARD
Certificate Maker ... $36
C.M. Library #1 ... $21
Early Games ... $21
Easy as ABC ... $24
Newsroom ... $36
Newsroom Pro ... $89
N.R. Clip Art #1 ... $19
N.R. Clip Art #2 ... $24
N.R. Clip Art #3 ... $19
P.S. Graphics Expander $24
SSI
Battle of Antietam ... $30
Gettysburg ... $36
Kampfgruppe ... $36
Phantasie ... $24
Rings of Zilfin ... $24
Roadwar 2000 ... $24
Shard of Spring ... $24
SUBLOGIC
Football ... $26
Jet ... $32
Pure Stat Baseball ... $32
TELARIUM
Amazon ... $9.88
Dragonworld ... $9.88
Fahrenheit 451 ... $9.88
Nine Princes ... $9.88
Perry Mason ... $9.88
Rendezvous ... $9.88
THUNDER MOUNTAIN
Top Gun ... $6.88
TIMEWORKS
*Data Manager ... $39
Evelyn Wood Reader .. $39
Partner ... $39
*Swiftcalc w/Sideways . $39
Sylvia Porter's Financial Planner ... $59
*Word Writer ... $39
*All 3 in 1 Power Pack $79
UNISON WORLD
Fonts & Borders ... $23
Newsmaster ... $59
Printmaster Plus ... $36
Art Gallery 1 or 2 .. $19 Ea.
WEEKLY READER
Stickybear Math ... $23
Stickybear Numbers .. $23
Stickybear Reading ... $23
Stickybear Spellgrabber $23
Stickybear Typing ... $23

APPLE II SERIES

ACCESS
Triple Pack: BH1, BH2, Raid Over Moscow .. $14
World Class Leader Board ... $25
ACCOLADE
Comics ... $19
Hardball ... $19
ACTIVISION
Draw GS ... $47
Gamemaker Library Disks: Sci-Fi or Sports .. $16 Ea.
Gamemaker ... $30
Ghostbusters ... $9.88
Hacker 2 ... $24
Hacker 2 GS ... $29
Might & Magic ... $30
Music Studio GS ... $49
Paintworks Clip Art Gallery GS ... $19
Paintworks GS ... $47
Shanghai ... $21
Shanghai GS ... $29
Tass Times GS ... $24
Writer's Choice Elite GS $47
ARTWORX
Bridge 5.0 ... $19
International Hockey . $19
Strip Poker ... $19
Data Disk #1 (Female) . $14
Data Disk #2 (Male) ... $14
AVALON HILL
Dr. Ruth ... $19
Super Sunday ... $21
BAUDVILLE
Award Maker Plus ... $24
French Paper Pack .. $9.88
Video Vegas ... $19
BRODERBUND
Airheart 128K ... $21
Animate 128K ... $42
Bank St. Writer Plus 128K ... $47
Carmen San Diego USA ... $27
Carmen San Diego World ... $24
Dazzle Draw 128K ... $34
Fantavision ... $30
Karateka ... $21
On Balance 128K ... $59
Print Shop ... $27
Print Shop Companion $23
P.S. Graphics Library #1, 2 or 3 ... $14 Ea.
P.S. Graphics Library Holiday Edition ... $14
Science Tool Kit ... $39
Toy Shop ... $32
Type! ... $27
CENTRAL POINT
Copy 2 ... $21
DAVIDSON
Algeblaster ... $27
Grammar Gremlins ... $27
Math Blaster ... $27
Spell It ... $27
Word Attack ... $27
DATA EAST
Commando ... $21
Tag Team Wrestling ... $21
ELECTRONIC ARTS
Age of Adventure ... $9.88
Amnesia ... $29
Archon ... $9.88
Archon 2: Adept ... $9.88
Arctic Fox ... $26
Bard's Tale GS ... $32
Bard's Tale 2 ... $32
Chessmaster 2000 ... $29
Deluxe Paint II GS ... $65
D. Paint Art #1 GS ... $19
D. Paint Art #2 GS ... $19
Earth Orbit Station ... $23
Lords of Conquest ... $26
Murder Party ... $26
Marble Madness 128K . $23
Mind Mirror ... $9.88
Moebius ... $36
Movie Maker ... $9.88
Music Const. Set ... $9.88
Music Const. Set GS ... $32
One-on-One ... $9.88
Pegasus ... $23
Pinball Const. Set ... $9.88
Russia (The Great War) $26
Scrabble ... $26
Seven Cities of Gold . $9.88
Skyfox ... $9.88
Star Fleet 1 ... $32
Super Boulder Dash . $9.88
Ultima 4 ... $36
EPYX
Champ. Wrestling ... $23
Create A Calendar ... $19
Destroyer ... $23
P.S. Graphics Scrapbook:
#1 Sports ... $14
#2 Off the Wall ... $14
#3 School ... $14
Sub Battle Simulator .. $23
Summer Games 2 ... $23
Winter Games ... $23
GAMESTAR
Champ. Baseball 128K . $24
Ch. Basketball 128K ... $24
Champ. Football 128K . $24
HI-TECH EXPRESSIONS
Award Ware ... $9.88
Card Ware ... $6.88
Heart Ware ... $6.88
Party Ware ... $9.88
INFOCOM
Bureaucracy 128K ... $24
Hitchhiker's Guide ... $19
Hollywood Hijinx ... $24
Leather Goddesses ... $24
Zork Trilogy ... $44
MECA
Andrew Tobias: Managing Your Money 2.0 ... $95
MICROLEAGUE
Baseball ... $24
Box Score Stats ... $14
General Mgr./Owner .. $19
1986 Team Disk ... $12
MICROPROSE
F-15 Strike Eagle ... $21
Silent Service ... $21
MINDSCAPE
American Challenge .. $24
Bop & Wrestle ... $19
Crossword Magic ... $30
Infiltrator ... $19
Perfect Score SAT ... $44
MONOGRAM
Dollars & Sense 128K .. $69
PERSONAL CHOICE
Term Paper Writer 128K $30
SHARE DATA
Family Feud ... $8.88
Jeopardy ... $8.88
Wheel of Fortune ... $8.88
SIERRA
Black Cauldron ... $24
King's Quest 1 or 2 . $30 Ea.
Wrath of Denethonor . $16
SPINNAKER
T-Shirt Shop ... $30
T-Shirt Art #1 ... $14
SPRINGBOARD
Certificate Maker ... $29
C.M. Library Vol. 1 ... $21
Newsroom ... $32
N.R. Clip Art #1 or #3 .. $17
N.R. Clip Art #2 ... $23
STYLEWARE
Multiscribe 2.0 128K .. $44
Multiscribe 2.0 GS ... $65
SSI
Battlecruiser ... $35
Colonial Conquest ... $24
Gettysburg ... $35
Phantasie 1, 2 or 3 . $24 ea.
Realms of Darkness ... $24
Rings of Zilfin ... $24
Roadwar 2000 ... $24
Shard of Spring ... $24
War in S. Pacific ... $35
Wizard's Crown ... $24
SUBLOGIC
Baseball ... $32
Flight Simulator 2 ... $30
F.S. Scenery Disks .. $14 Ea.
Jet ... $26
TELARIUM
Amazon ... $9.88
Dragonworld ... $9.88
Fahrenheit 451 ... $9.88
Perry Mason ... $9.88
VIP TECHNOLOGIES
VIP Professional GS .. $179
WEEKLY READER
Stickybear ABC's ... $23
Stickybear Math 1 ... $23
Stickybear Math 2 ... $23
Stickybear Numbers .. $23
Stickybear Printer ... $23
Stickybear Shapes ... $23
Stickybear Spellgrabber $23
Stickybear Typing ... $23
WINDHAM CLASSICS
Alice in Wonderland . $9.88
Below the Root ... $9.88
Swiss Family Robinson $9.88
Treasure Island ... $9.88
Wizard of Oz ... $9.88

Commodore 64 & 128 Owners: Please read our full page ad in this months Compute's Gazette!

P.O. BOX 111327—DEPT. CP—BLAWNOX, PA 15238

*Please Read The Following Ordering Terms & Conditions Carefully Before Placing Your Order: Orders with cashiers check or money order shipped immediately on in stock items. Personal & Company checks, allow 3 weeks clearance. No C.O.D.'s! Shipping: Continental U.S.A.—Orders under $100 add $3; free shipping on orders over $100. AK, HI, FPO, APO—add $5 on all orders. Canada & Puerto Rico—add $10 on all orders. Sorry, no other International orders accepted! PA residents add 6% sales tax on the total amount of order including shipping charges! REASONS FOR CALLING CUSTOMER SERVICE—412-361-5291 (1) Status of order or back order (2) If any merchandise purchased within 60 days from S.D. of A. is defective, please call for a return authorization number. Defective merchandise will be replaced with the same merchandise only. Other returns subject to a 15% restocking charge. After 60 days please refer to the manufacturers warranty included with the merchandise & return directly to the manufacturer. Customer service will not accept collect calls or calls on S.D. OF A.'s 800# order lines! Have you seen our on line catalog of 1000 software titles for Commodore, Atari, Apple, IBM and Amiga? It's on Compuserve's Electronic Mall—just type GO SDA and shopping for software will never be the same again! HOURS: Mon.-Fri. 9AM-5:30 PM, Sat. 10AM-4PM Eastern Time. Because this ad had to be written 2-3 mos. before it was published, prices & availability are subject to change! When sending a mail order, please specify make & model# of your computer!

Score ($495), a desktop music publishing system for composing and printing high-quality sheet music.

Passport now has more than 50 music products available for seven different computer systems.

Passport Designs, 625 Miramontes St., Half Moon Bay, CA 94019
Circle Reader Service Number 218.

Byte Size Software

This new company's Byte Size line of low-cost applications software is targeted to meet specific consumer needs, with each package designed to do one thing well and to be easy to use. Initial products, available for PCs and compatibles, will include such programs as *Business Graphics, Word Processor, Stock Portfolio, Telecommunications, Labeler, Home Inventory, Calculator, Time Keeper,* and more.

Software Resources International, 1209 W. Knickerbocker Dr., Sunnyvale, CA 94087
Circle Reader Service Number 219.

IBM Programs On The ST

Avant-Garde Systems has introduced *pc-ditto*, a software package that allows IBM PC software to run on the Atari 520ST and 1040ST computers. The $89.95 program features up to 703K usable memory (with 1040ST), is not copy-protected (hard disk installable), supports 3½-inch 80-track formats and an optional 5¼-inch 40-track drive, imitates monochrome and color graphics adapters, and provides adjustable palette colors.

The company states that *pc-ditto* enables an ST to run virtually all of the major IBM programs, such as *Lotus 1-2-3, Multiplan, Symphony, Flight Simulator, Dollars & Sense, Sidekick, DAC Easy Accounting, Turbo Pascal,* and many others.

Avant-Garde Systems, 381 Pablo Point Dr., Jacksonville, FL 32225
Circle Reader Service Number 220.

AwardWare From Hi Tech

Hi Tech Expressions has introduced *AwardWare*, a $14.95 design and printing package for the creation of awards, certificates, letterheads, ribbons, tickets, coupons, checks, and other printed items. The program has 20 graphics, 20 graphic borders, unlimited text borders, five seals, and five fonts.

The program is available for IBM PCs and compatibles, the Apple II family (64K minimum), the Commodore 64, and the Atari XL and XE computers. A color graphics adapter and 256K are necessary to run the program's animation function on the IBM and compatibles. A dot-matrix printer is required.

Hi Tech Expressions, 1700 N.W. 65th Ave., Suite 9, Plantation, FL 33313
Circle Reader Service Number 221.

Amiga Image Capture

NewTek has announced the release of *Digi-View 2.0*, a major upgrade of the company's color image capture system for the Amiga 500, 1000, and 2000 computers.

The $199.95 program uses advanced graphics routines and dithering techniques to display images with up to 4096 colors at once in resolutions to 640 × 400. The *Digi-View* system works with any home color or black-and-white video camera, using NewTek's three-color filter wheel system to emulate a broadcast video camera. The package captures 21 bitplanes, or over two million colors.

New features include support of all resolution modes, the new enhanced hold and modify mode for increased color resolution, and compatibility with the new Amiga 500 and 2000 computers.

NewTek, 115 West Crane St., Topeka, KS 66603
Circle Reader Service Number 222.

Thesaurus For MS-Word 3.0

Users of *Microsoft Word 3.0* on the Macintosh now have access to a 28K desk accessory thesaurus. The *Merriam Webster Thesaurus for Microsoft Word 3.0* runs only with *Microsoft Word 3.0* and contains 45,000 whole root words with up to 1.4 million combinations of synonyms. All the root words are listed with definitions. The virtual cross-referencing feature enables a user to continuously cross-reference from one synonym to another. When a synonym is located, it is automatically inserted into the text.

Retail price for the desk accessory thesaurus from Target Software is $59.95.

Target Software has also upgraded *MacLightning* to include the Merriam Webster *Ninth New Collegiate Dictionary.* There are two versions of this upgrade—*Mentor* and *Mentor Plus*—both of which contain an abridged version of the *Merriam Webster Ninth New Collegiate Dictionary.* This 100,000-word electronic dictionary occupies approximately one megabyte and requires a hard disk drive.

Mentor Plus also includes the 75,000-word *Merriam Webster Pocket Dictionary* with definitions. It is designed for desktop publishing and large word processing use.

Retail price for *Mentor* is $99.95, and the upgrade is available for $25. *Mentor Plus* retails for $199.95 and the upgrade is available for $100. The upgrades are available for *MacLightning,* which retails for $99.95.

Target Software, 14206 SW 136th St., Miami, FL 33186
Circle Reader Service Number 223.

Power Up! Products

Organize all your addresses into one file and then print out an address book with Power Up!'s *Pocket Address Book.* Each entry may contain a name, two phone numbers, a company name, a two-line address, city, state, zip code, a profession, and a comment; each entry can also be listed and sorted by name, company, or profession.

With the program, you can print out by any of the categories in the form of an address book, rotary-file cards, labels, and envelopes. Envelopes and mailing labels can be sorted by name or zip code.

Paper and a black leather cover are included so you can create an address book that's 6 × 3⅜ inches. A pen and calender are also included.

The *Financial Reporter* helps you prepare columnar reports by automatically aligning columns and performing mathematical functions.

Financial Reporter can work with up to eight numeric columns and up to 200 lines per report. You can enter data in a column or row, and the program will calculate up to five levels of subtotals plus the total. You can move complete rows at

MONTGOMERY GRANT

115 Nassau St. Between Beekman & Ann Sts N.Y., N.Y. 10038 (212) 732-4500
open Mon-Fri 8:30am - 6:30pm Sat 9:30am - 6pm Sun 9:30am - 5:30pm

Penn Station/Amtrack Level - Beneath Madison Sq. Garden, N.Y., N.Y. 10001
(212) 594-7140 open Mon-Fri 8:30am - 8pm Sat & Sun 9:30am - 7pm

PHONE and MAIL ORDERS CALL TOLL FREE **1(800)345-7058** IN NEW YORK STATE CALL (212)732-4500

ALL COMPUTERS & PERIPHERALS COME COMPLETE WITH AC ADAPTOR, OWNER'S MANUAL AND ALL CONNECTING CABLES.

ALL MERCHANDISE IS BRAND NEW AND FACTORY FRESH, COMPLETE WITH MFR. U.S.A. WARRANTY.

EPSON

EQUITY I PACKAGE
IBM PC/XT COMPATIBLE
• Equity I Computer • 256K RAM Power • Dual 360K Disk Drives • 12" High Resolution Monitor • Serial and Parallel Ports

$599.95

EQUITY II PACKAGE
IBM PC/XT COMPATIBLE
• Equity II Computer • 640K RAM Power • 360K Disk Drive • 12" High Resolution Monitor • Serial and Parallel Ports

$999.95
With 360K Drive & 20MB Hard Drive $1299

IBM®

IBM XT HARD DRIVE PACKAGE
• IBM XT Computer • IBM Keyboard • 256K RAM Expandable to 640K • 360K Disk Drive • 20MB Hard Drive • Monitor Optional • Package of 10 Diskettes

$1299.95

IBM XT PACKAGE with Two 360K Disk Drives **$999.95**

IBM PERSONAL SYSTEM II
MODEL 30
• IBM Computer • Two 720K Drives • 640K RAM • Built-in Graphics • Serial and Parallel Ports

$1499.95

apple

APPLE IIE PACKAGE
• Apple IIE Keyboard • 5.25" Disk Drive • 12" Monitor • All AC Adaptors and Hook-up Cables • Package of 10 Diskettes

NOW ONLY $969.95

APPLE IIGS PACKAGE
• Apple IIGS Computer • 5.25" Disk Drive • 12" Monitor • Package of 10 Diskettes • AC Adaptors • All Hook-up Cables

$1195

COMPUTER PRINTER BONANZA

EPSON HOMEWRITER 10	$189.95	STAR SG-10C	$179.95	STAR SD-15	$399.95	STAR NB24-15	$669.95
EPSON LX-86	$219.95	STAR NX-10	$189.95	STAR ND-15	$419.95	STAR NB-15	$869.95
EPSON FX-86	$349.95	STAR NX-10C	$199.95	STAR SR-10	$429.95		
EPSON FX-286E	$469.95	STAR SD-10	$249.95	STAR SR-15	$469.95	OKIDATA 120	$199.95
EPSON LQ-800	$469.95	STAR NL-10	$249.95	STAR NR-15	$499.95	OKIMATE 20 w/Plug'n'Print	$189.95
PANASONIC 1081I	$189.95	STAR NX-15	$309.95	STAR NB24-10	$469.95		

BLUE CHIP

IBM PC/XT COMPATIBLE PCKG.
• Blue Chip Keyboard • 512K Expandable to 640K • 6 Expansion Slots • Optional 20MB Hard Drive • All Hook-up Cables & Adaptors • Package of 10 Diskettes • 12" Monitor

SPECIAL
$579.95

commodore

PC10-1 PACKAGE
IBM PC/XT Compatible
• PC10-1 Computer • 512K Expandable to 640K • 360K Disk Drive • Enhanced Keyboard • Serial & Parallel Ports • All Hook-up Cables & Adaptors • 10 Diskettes

$649.95

PC10-2 640K Computer w/2 360K Floppy Drives **$769.95**

AMIGA

AMIGA 512K SYSTEM
512K Computer with Built-in Disk Drive and 1080 Color Monitor

$1099.95

AMIGA 2000.....CALL
ATARI 1040 Color System....$869
ATARI 520ST Color System.$669

apple

APPLE IIC PACKAGE
• Apple IIC Keyboard • Built-in Disk Drive • Apple Monitor • Apple Monitor Stand • All Cables & Adaptors • 10 Diskettes

$899.95

APPLE IMAGEWRITER PRINTER NOW ONLY.....$489.95

commodore

COMMODORE 64-C PERSONAL COMPUTER	149.95
COMMODORE 128 PERSONAL COMPUTER	235
COMMODORE MPS-1200 Computer Printer	229.95
COMMODORE MPS-803C Computer Printer	129.95
COMMODORE MPS-1000 Computer Printer	199.95
COMMODORE 1351 MOUSE for any Commodore Computer	44.95

COMPUTER PACKAGES
Get it all together & save!

C64 COMPLETE PACKAGE
• Commodore 64 Computer
• Commodore 1541 Disk Drive
• Commodore Color Printer
• 12" Monitor

$369.95

Same Pckg. with C64/C $399

Color Monitor ADD $130

C64C COLOR PACKAGE
• Commodore 64C Computer
• Commodore 1541 Disk Drive
• Commodore Color Printer
• Color Monitor
With Geos Program!

$529.95

C128 COMPLETE PACKAGE
• Commodore 128 Computer
• Commodore 1541 Disk Drive
• 12" Monitor
• Commodore Color Printer

$499.95

To substitute 1571 for 1541 add **$65**

C128 DELUXE PACKAGE
• Commodore 128 Computer
• Commodore 1571 Disk Drive
• Commodore 1902 Color Monitor
• Commodore 80 Column Printer

$759.95

commodore

COMMODORE 1541C NEW Disk Drive	165.95
COMMODORE 1541 Disk Drive - SPECIAL	149.95
COMMODORE 1571 Computer Disk Drive	214.95
COMMODORE 1902 80 Column RGB Color Monitor	249.95
COMMODORE 1802 Color Computer Monitor	189.95
COMMODORE "MAGIC VOICE" Module	49.95
COMMODORE 1670 AutoModem	109.95
COMMODORE 1700 - 128K Expansion Module	99.95
COMMODORE 1750 - 512K Expansion Module	169.95

MasterCard DISCOVER VISA

WE INVITE CORPORATE & EDUCATIONAL CUSTOMERS

Certified check, Bank check, Mastercard, Visa, Am-Ex, Diner's Club, Carte-Blanche, Discover Card and C.O.D.s accepted. * No additional surcharge for credit card orders. Non-certified checks must wait 4-6 weeks clearance. Money orders are non-certified checks. N.Y. residents add applicable sales tax. Prices and availability subject to change without notice-so call before placing your order. Not responsible for typographic errors. Return of defective merchandise must have prior return authorization number, or returns will not be accepted. IBM PC/XT are registered trademarks of International Business Machine Corp. All orders can be shipped Air Express-call for details.

one time or move, swap, combine, or clear whole columns. The sort option helps you reformat your report to choose the correct form. The screen shows the report as it will print out.

There are built-in math functions so you can perform calculations without entering the formulas. A pop-up calculator sends the results to any location in the report.

Pocket Address Book and *Financial Reporter* run on the IBM PC, Compaq, and compatibles with 256K memory. The printer must be able to print at 15 to 17 cpi.

Retail price for the address book is $59.95 and $49.95 for *Financial Reporter*. Both have a 30-day money-back guarantee and are available only from the Power Up! catalog.

Power Up! Catalog, Channelmark Corporation, P.O. Box 7600, 2929 Campus Dr., San Mateo, CA 94403

Circle Reader Service Number 224.

Amiga Spelling Checker

LexCheck, a spelling-checking program for the Amiga, works with *Textcraft, Scribble!, Notepad,* and ASCII text files. The master dictionary contains over 100,000 words, and there are auxiliary dictionaries that enable you to add your own words. *LexCheck* is disk-based and uses less than 100K of RAM, so you can run it with your word processor. This spelling checker recognizes proper names, place names, and technical terms.

LexCheck requires *Workbench 1.2*. Retail price is $42.95.

Complete Data Automation, P.O. Box 1052, Yreka, CA 96097

Circle Reader Service Number 225.

Mystery Reading Program From Mindplay

Students in grades two through eight can use clues and intuitive thinking to write their own solutions to 60 mystery stories in *Ace Detective*. This educational program is designed to help improve reading comprehension, cause-and-effect reasoning, and creative writing skills. The players visit the scene of the crime, interview witnesses, and check mug shots to determine the motive, opportunity, means, and suspect. Players must read and draw conclusions to get search warrants and to print out mug shots of the suspects.

New mysteries can be created using the story-builder feature which has over 60 mug-shot graphics and a telephone directory. The *Challenge Upgrade* option includes sound, time, level, text speed, story selection, story creation, and performance summaries.

The teacher's software package includes a disk, a backup, a teacher's guide and a user's guide. It is available for the Apple II series, including the IIGS.

The home edition is priced at $39.99, the teacher's edition at $49.99, and a six-disk lab pack is available for $120.

Mindplay, 82 Montvale Ave., Stoneham, MA 02180

Circle Reader Service Number 226.

Your Family History On Disk

All About Us helps you record your family and personal records. It contains many entries where you can fill in information such as birthdates, memorable events, geneology, estate planning, personal loans, and medical history. You can enter information on as many family members as you wish and then print out the information on individuals or groups.

The software package includes a 220-page user's guide, a glossary of terms, a keyboard template, and an appendix on wallet-sized cards.

All About Us runs on the IBM PC, XT, AT, and compatibles with 192K DOS 2.0 or 256K DOS 3.0. An 80-column printer is optional.

The price of the package is $60.

IlliNet Software, 123 Mumford Hall, 1301 W. Gregory Dr., Urbana, IL 61801

Circle Reader Service Number 227.

New Adventure/ Simulation Game

Pirates! combines the action of an adventure game with battle and sailing simulations to recreate the seventeenth-century pirating era. Players travel throughout the Caribbean, taking part in land battles, ship duels, and sword fights. Actual sailing experiences are simulated by changing weather conditions and the characteristics of early barques, sloops, frigates, and galleons. There's also the constant threat of mutiny and panic.

Play takes place in six different time periods from 1560 to 1680. Players take the roles of English, French, Dutch, or Spanish pirates; the expeditions of Francis Drake, Piet Heyn, and Henry Morgan are also recreated. Difficulty of the game varies so beginners and advanced players are challenged.

Pirates! is stored on both sides of a 5¼-inch floppy disk for the Commodore 64.

Suggested retail price is $39.95.

Microprose Software, 120 Lakefront Dr., Hunt Valley, MD 21030

Circle Reader Service Number 228.

Arcade Adventure For Commodore 64/128

Cauldron is a new, two-part adventure game from Brøderbund consisting of *Cauldron* and *Cauldron II*. In *Cauldron*, you are a witch queen whose broom has been stolen by the pumpking. In *Cauldron II*, you play the part of a pumpking warrior trying to destroy the witch queen. The object in both games is to collect ingredients for a magic potion to brew in the cauldron. Along the way, you meet spiders and skeletons plus a host of other ghouls in over 190 different scenes.

Suggested retail price is $29.95.

Brøderbund Software, 17 Paul Dr., San Rafael, CA 94903-2101

Circle Reader Service Number 229. ©

COMPUTE!'s Author's Guide

Most of the following suggestions serve to improve the speed and accuracy of publication. COMPUTE! is primarily interested in new and timely articles on the Commodore 64/128, Atari, Apple, IBM PC/PCjr, Amiga, and Atari ST. We are much more concerned with the content of an article than with its style, but articles should be clear and well-explained.

The guidelines below will permit your good ideas and programs to be more easily edited and published:

1. The upper left corner of the first page should contain your name, address, telephone number, and the date of submission.

2. The following information should appear in the upper right corner of the first page: If your article is specifically directed to one make of computer, please state the brand name and, if applicable, the BASIC or ROM or DOS version(s) involved. In addition, *please indicate the memory requirements of programs.*

3. The underlined title of the article should be placed about ⅔ of the way down the first page.

4. Following pages should be typed normally, except that in the upper right corner there should be an abbreviation of the title, your last name, and the page number—for example: Memory Map/Smith/2.

5. All lines within the text of the article must be double- or triple-spaced. A one-inch margin should be left at the right, left, top, and bottom of each page. No words should be divided at the ends of lines. And please do not right-justify. Leave the lines ragged.

6. Standard typing paper should be used (no erasable, onionskin, or other thin paper), and typing should be on one side of the paper only (upper- and lowercase).

7. If you are submitting more than one article, send each one in a separate mailer with its own tape or disk.

8. Short programs (under 20 lines) can easily be included within the text. Longer programs should be separate listings. *It is essential that we have a copy of the program, recorded twice, on a tape or disk.* If your article was written with a word processor, we request that you include a copy of the text file on the tape or disk. If you include a copy of your article on disk, please save the article as plain text, without any special formatting characters or control codes. Most word processors provide an option for saving a document as plain ASCII text or in unformatted form. Please use high-quality 10- or 30-minute tapes with the program recorded on both sides. The tape or disk should be labeled with your name, the title of the article, and, if applicable, the BASIC/ROM/DOS version(s). Tapes are fairly sturdy, but disks need to be enclosed within plastic or cardboard mailers (available at photography, stationery, or computer supply stores). If possible, programs written in machine language or a compiled language should include source code (or an annotated disassembly if the program was written with a machine language monitor).

9. A good general rule is to spell out the numbers zero through ten in your article and write higher numbers as numerals (1024). The exceptions to this are: Figure 5, Table 3, TAB(4), and so on. Within ordinary text, however, the zero through ten should appear as words, not numbers. Also, symbols and abbreviations should not be used within text: Use *and* (not &), *reference* (not ref.), *through* (not thru).

10. For greater clarity, use all capitals when referring to keys (RETURN, TAB, ESC, SHIFT), BASIC words (LIST, RND, GOTO), and three languages (BASIC, APL, PILOT). Headlines and subheads should, however, be initial caps only, and emphasized words are not capitalized. If you wish to emphasize, underline the word; then it will be italicized during typesetting.

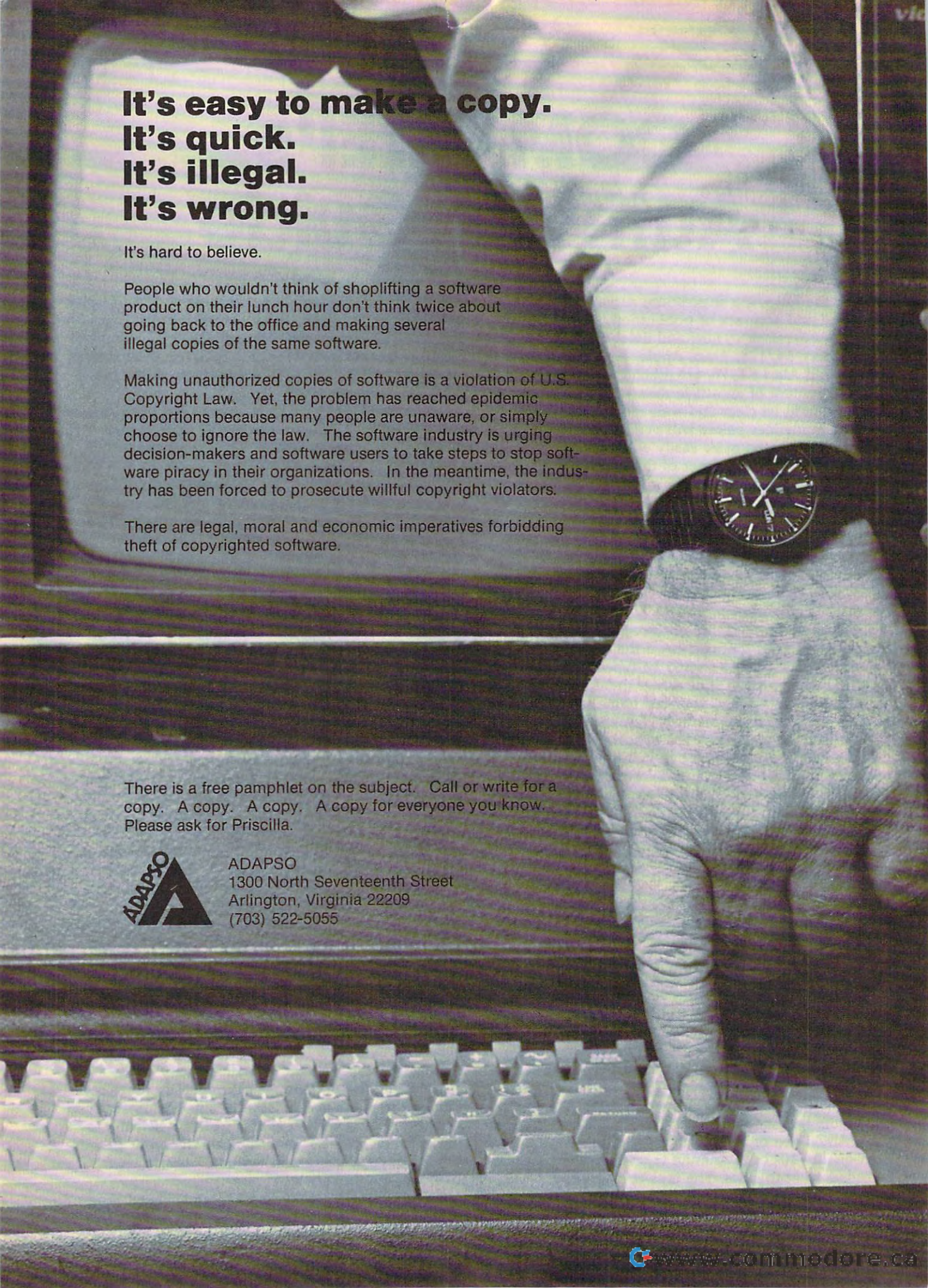
11. Articles can be of any length—from a single-line routine to a multiple-issue series. The average article is about four to eight double-spaced, typed pages.

12. We do not consider articles which are submitted simultaneously to other publishers. If you wish to send an article to another magazine for consideration, please do not submit it to us.

13. COMPUTE! pays between $70 and $800 for published articles. In general, the rate reflects the length and quality of the article. Payment is made upon acceptance. Following submission (to Editorial Department, COMPUTE! Magazine, P.O. Box 5406, Greensboro, NC 27403), it will take from four to eight weeks for us to reply. If your work is accepted, you will be notified by a letter which will include a contract for you to sign and return. *Rejected manuscripts are returned to authors who enclose a self-addressed, stamped envelope.*

14. If your article is accepted and you have since made improvements to the program, please submit an entirely new tape or disk and a new copy of the article reflecting the update. We cannot easily make revisions to programs and articles. It is necessary that you send the revised version as if it were a new submission entirely, but be sure to indicate that your submission is a revised version by writing *Revision* on the envelope and the article.

15. COMPUTE! does not accept unsolicited product reviews. If you are interested in serving on our panel of reviewers, contact the Features Editor for details.



COMPUTE!'s Guide To Typing In Programs

Computers are precise—type the program *exactly* as listed, including necessary punctuation and symbols, except for special characters noted below. We have provided a special listing convention as well as a program to check your typing—"The Automatic Proofreader."

Programs for the IBM, TI-99/4A, and Atari ST models should be typed exactly as listed; no special characters are used. Programs for Commodore, Apple, and Atari 400/800/XL/XE computers may contain some hard-to-read special characters, so we have a listing system that indicates these control characters. You will find these Commodore and Atari characters in curly braces; *do not type the braces.* For example, {CLEAR} or {CLR} instructs you to insert the symbol which clears the screen on the Atari or Commodore machines. A complete list of these symbols is shown in the tables below. For Commodore, Apple, and Atari, a single symbol by itself within curly braces is usually a control key or graphics key. If you see {A}, hold down the CONTROL key and press A. This will produce a reverse video character on the Commodore (in quote mode), a graphics character on the Atari, and an invisible control character on the Apple.

Graphics characters entered with the Commodore logo key are enclosed in a special bracket: [<A>]. In this case, you would hold down the Commodore logo key as you type A. Our Commodore listings are in uppercase, so shifted symbols are underlined. A graphics heart symbol (SHIFT-S) would be listed as S. One exception is {SHIFT-SPACE}. When you see this, hold down SHIFT and press the space bar. If a number precedes a symbol, such as {5 RIGHT}, {6 S}, or [<8 Q>], you would enter five cursor rights, six shifted S's, or eight Commodore-Q's. On the Atari, inverse characters (white on black) should be entered with the inverse video

Atari 400/800/XL/XE

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

Commodore PET/CBM/VIC/64/128/16/+4

When You Read:	Press:	See:
{CLR}	SHIFT + CLR/HOME	
{HOME}	CLR/HOME	
{UP}	SHIFT + ↑ CRSR ↓	
{DOWN}	↑ CRSR ↓	
{LEFT}	SHIFT + ← CRSR →	
{RIGHT}	← CRSR →	
{RVS}	CTRL 9	
{OFF}	CTRL 0	
{BLK}	CTRL 1	
{WHT}	CTRL 2	
{RED}	CTRL 3	
{CYN}	CTRL 4	
{PUR}	CTRL 5	
{GRN}	CTRL 6	
{BLU}	CTRL 7	
{YEL}	CTRL 8	

When You Read:	Press:	See:
[<1>]	COMMODORE 1	
[<2>]	COMMODORE 2	
[<3>]	COMMODORE 3	
[<4>]	COMMODORE 4	
[<5>]	COMMODORE 5	
[<6>]	COMMODORE 6	
[<7>]	COMMODORE 7	
[<8>]	COMMODORE 8	
{ F1 }	f1	
{ F2 }	SHIFT + f1	
{ F3 }	f3	
{ F4 }	SHIFT + f3	
{ F5 }	f5	
{ F6 }	SHIFT + f5	
{ F7 }	f7	
{ F8 }	SHIFT + f7	
←	←	

key (Atari logo key on 400/800 models).

Whenever more than two spaces appear in a row, they are listed in a special format. For example, {6 SPACES} means press the space bar six times. Our Commodore listings never leave a single space at the end of a line, instead moving it to the next printed line as {SPACE}.

Amiga program listings contain only one special character, the left arrow (←) symbol. This character marks the end of each program line. Wherever you see a left arrow, press RETURN or move the cursor off the line to enter that line into memory. Don't try to type in the left arrow symbol; it's there only as a marker to indicate where each program line ends.

The Automatic Proofreader

Type in the appropriate program listed below, then save it for future use. The Commodore Proofreader works on the Commodore 128, 64, Plus/4, 16, and VIC-20. Don't omit any lines, even if they contain unfamiliar commands or you think they don't apply to your computer. When you run the program, it installs a machine language program in memory and erases its BASIC portion automatically (so be sure to save several copies before running the program for the first time). If you're using a Commodore 128, Plus/4 or 16, do *not* use any GRAPHIC commands while the Proofreader is active. You should disable the Commodore Proofreader before running any other program. To do this, either turn the computer off and on or enter SYS 64738 (for the 64), SYS 65341 (128), SYS 64802 (VIC-20), or SYS 65526 (Plus/4 or 16). To reenable the Proofreader, reload the program and run it as usual. Unlike the original VIC/64 Proofreader, this version works the same with disk or tape.

On the Atari, run the Proofreader to activate it (the Proofreader remains active in memory as a machine language program); you must then enter NEW to erase the BASIC loader. Pressing SYSTEM RESET deactivates the Atari Proofreader; enter PRINT USR(1536) to reenable it.

The Apple Proofreader erases the BASIC portion of itself after you run it, leaving only the machine language portion in memory. It works with either DOS 3.3 or ProDOS. Disable the Apple Proofreader by pressing CTRL-RESET before running another BASIC program.

The IBM Proofreader is a BASIC program that simulates the IBM BASIC line editor, letting you enter, edit, list, save, and load programs that you type. Type RUN to activate. Be sure to leave Caps Lock on, except when typing lowercase characters.

Once the Proofreader is active, try typing in a line. As soon as you press RETURN, either a hexadecimal number (on the Apple) or a pair of letters (on the Commodore, Atari, or IBM) appears. The number or pair of letters is called a *checksum*.

Compare the value displayed on the screen by the Proofreader with the checksum printed in the program listing in the magazine. The checksum is given to the left of each line number. Just type in the program a line at a time (without the printed checksum), press RETURN or Enter, and compare the checksums. If they match, go on to the next line. If not, check your typing; you've made a mistake. Because of the checksum method used, do not type abbreviations, such as ? for PRINT. On the Atari and Apple Proofreaders, spaces are not counted as part of the checksum, so be sure you type the right number of spaces between quote marks. The Atari Proofreader does not check to see that you've typed the characters in the right order, so if characters are transposed, the checksum still matches the listing. The Commodore Proofreader catches transposition errors and ignores spaces unless they're enclosed in quotation marks. The IBM Proofreader detects errors in spacing and transposition.

IBM Proofreader Commands

Since the IBM Proofreader replaces the computer's normal BASIC line editor, it has to include many of the direct-mode IBM BASIC commands. The syntax is identical to IBM BASIC. Commands simulated are LIST, LLIST, NEW, FILES, SAVE, and LOAD. When listing your program, press any key (except Ctrl-Break) to stop the listing. If you enter NEW, the Proofreader prompts you to press Y to be especially sure you mean yes.

Two new commands are BASIC and CHECK. BASIC exits the Proofreader back to IBM BASIC, leaving the Proofreader in memory. CHECK works just like LIST, but shows the checksums along with the listing. After you have typed in a program, save it to disk. Then exit the Proofreader with the BASIC command, and load the program as usual (this replaces the Proofreader in memory). You can now run the program, but you may want to resave it to disk. This will shorten it on disk and make it load faster, but it can no longer be edited with the Proofreader. If you want to convert an existing BASIC program to Proofreader format, save it to disk with SAVE *"filename"*,A.

Program 1: Atari Proofreader

By Charles Brannon

```
100 GRAPHICS 0
110 FOR I=1536 TO 1700:REA
    D A:POKE I,A:CK=CK+A:N
    EXT I
120 IF CK<>19072 THEN ? "E
    rror in DATA Statement
    s.  Check Typing.":END

130 A=USR(1536)
140 ? :? "Automatic Proofr
    eader Now Activated."
150 END
160 DATA 104,160,0,185,26,
    3,201,69,240,7
170 DATA 200,200,192,34,20
    8,243,96,200,169,74
180 DATA 153,26,3,200,169,
    6,153,26,3,162
190 DATA 0,189,0,228,157,7
    4,6,232,224,16
200 DATA 208,245,169,93,14
    1,78,6,169,6,141
210 DATA 79,6,24,173,4,228
    ,105,1,141,95
220 DATA 6,173,5,228,105,0
    ,141,96,6,169
230 DATA 0,133,203,96,247,
    238,125,241,93,6
240 DATA 244,241,115,241,1
    24,241,76,205,238
250 DATA 0,0,0,0,0,32,62,2
    46,8,201
260 DATA 155,240,13,201,32
    ,240,7,72,24,101
270 DATA 203,133,203,104,4
    0,96,72,152,72,138
280 DATA 72,160,0,169,128,
    145,88,200,192,40
290 DATA 208,249,165,203,7
    4,74,74,74,24,105
300 DATA 161,160,3,145,88,
    165,203,41,15,24
310 DATA 105,161,200,145,8
    8,169,0,133,203,104
320 DATA 170,104,168,104,4
    0,96
```

Program 2: IBM Proofreader

By Charles Brannon

```
10 'Automatic Proofreader Vers
   ion 3.0 (Lines 205,206 adde
   d/190 deleted/470,490 chang
   ed from V2.0)
100 DIM L$(500),LNUM(500):COLO
    R 0,7,7:KEY OFF:CLS:MAX=0:
    LNUM(0)=65536!
110 ON ERROR GOTO 120:KEY 15,C
    HR$(4)+CHR$(70):ON KEY(15)
     GOSUB 640:KEY (15) ON:GOT
    O 130
120 RESUME 130
130 DEF SEG=&H40:W=PEEK(&H4A)
140 ON ERROR GOTO 650:PRINT:PR
    INT"Proofreader Ready."
150 LINE INPUT L$:Y=CSRLIN-INT
    (LEN(L$)/W)-1:LOCATE Y,1
160 DEF SEG=0:POKE 1050,30:POK
    E 1052,34:POKE 1054,0:POKE
     1055,79:POKE 1056,13:POKE
     1057,28:LINE INPUT L$:DEF
     SEG:IF L$="" THEN 150
170 IF LEFT$(L$,1)=" " THEN L$
    =MID$(L$,2):GOTO 170
```

```
180 IF VAL(LEFT$(L$,2))=0 AND
    MID$(L$,3,1)=" " THEN L$=M
    ID$(L$,4)
200 IF ASC(L$)>57 THEN 260 'no
     line number, therefore co
    mmand
205 BL=INSTR(L$," "):IF BL=0 T
    HEN BL$=L$:GOTO 206 ELSE B
    L$=LEFT$(L$,BL-1)
206 LNUM=VAL(BL$):TEXT$=MID$(L
    $,LEN(STR$(LNUM))+1)
210 IF TEXT$="" THEN GOSUB 540
    :IF LNUM=LNUM(P) THEN GOSU
    B 560:GOTO 150 ELSE 150
220 CKSUM=0:FOR I=1 TO LEN(L$)
    :CKSUM=(CKSUM+ASC(MID$(L$,
    I))*I) AND 255:NEXT:LOCATE
     Y,1:PRINT CHR$(65+CKSUM/1
    6)+CHR$(65+(CKSUM AND 15))
    +" "+L$
230 GOSUB 540:IF LNUM(P)=LNUM
    THEN L$(P)=TEXT$:GOTO 150
    'replace line
240 GOSUB 580:GOTO 150 'insert
     the line
260 TEXT$="":FOR I=1 TO LEN(L$
    ):A=ASC(MID$(L$,I)):TEXT$=
    TEXT$+CHR$(A+32*(A>96 AND
    A<123)):NEXT
270 DELIMITER=INSTR(TEXT$," ")
    :COMMAND$=TEXT$:ARG$="":IF
     DELIMITER THEN COMMAND$=L
    EFT$(TEXT$,DELIMITER-1):AR
    G$=MID$(TEXT$,DELIMITER+1)
     ELSE DELIMITER=INSTR(TEXT
    $,CHR$(34)):IF DELIMITER T
    HEN COMMAND$=LEFT$(TEXT$,D
    ELIMITER-1):ARG$=MID$(TEXT
    $,DELIMITER)
280 IF COMMAND$<>"LIST" THEN 4
    10
290 OPEN "scrn:" FOR OUTPUT AS
     #1
300 IF ARG$="" THEN FIRST=0:P=
    MAX-1:GOTO 340
310 DELIMITER=INSTR(ARG$,"-"):
    IF DELIMITER=0 THEN LNUM=V
    AL(ARG$):GOSUB 540:FIRST=P
    :GOTO 340
320 FIRST=VAL(LEFT$(ARG$,DELIM
    ITER)):LAST=VAL(MID$(ARG$,
    DELIMITER+1))
330 LNUM=FIRST:GOSUB 540:FIRST
    =P:LNUM=LAST:GOSUB 540:IF
    P=0 THEN P=MAX-1
340 FOR X=FIRST TO P:N$=MID$(S
    TR$(LNUM(X)),2)+" "
350 IF CKFLAG=0 THEN A$="":GOT
    O 370
360 CKSUM=0:A$=N$+L$(X):FOR I=
    1 TO LEN(A$):CKSUM=(CKSUM+
    ASC(MID$(A$,I))*I) AND 255
    :NEXT:A$=CHR$(65+CKSUM/16)
    +CHR$(65+(CKSUM AND 15))+"
     "
370 PRINT #1,A$+N$+L$(X)
380 IF INKEY$<>"" THEN X=P
390 NEXT :CLOSE #1:CKFLAG=0
400 GOTO 130
410 IF COMMAND$="LLIST" THEN O
    PEN "lpt1:" FOR OUTPUT AS
    #1:GOTO 300
420 IF COMMAND$="CHECK" THEN C
    KFLAG=1:GOTO 290
430 IF COMMAND$<>"SAVE" THEN 4
    50
440 GOSUB 600:OPEN ARG$ FOR OU
    TPUT AS #1:ARG$="":GOTO 30
    0
450 IF COMMAND$<>"LOAD" THEN 4
    90
460 GOSUB 600:OPEN ARG$ FOR IN
    PUT AS #1:MAX=0:P=0
470 WHILE NOT EOF(1):LINE INPU
    T #1,L$:BL=INSTR(L$," "):B
    L$=LEFT$(L$,BL-1):LNUM(P)=
    VAL(BL$):L$(P)=MID$(L$,LEN
    (STR$(VAL(BL$)))+1):P=P+1:
    WEND
480 MAX=P:CLOSE #1:GOTO 130
490 IF COMMAND$="NEW" THEN INP
    UT "Erase program - Are yo
    u sure";L$:IF LEFT$(L$,1)=
    "y" OR LEFT$(L$,1)="Y" THE
    N MAX=0:LNUM(0)=65536!:GOT
    O 130:ELSE 130
500 IF COMMAND$="BASIC" THEN C
    OLOR 7,0,0:ON ERROR GOTO 0
    :CLS:END
510 IF COMMAND$<>"FILES" THEN
    520
515 IF ARG$="" THEN ARG$="A:"
    ELSE SEL=1:GOSUB 600
517 FILES ARG$:GOTO 130
520 PRINT"Syntax error":GOTO 1
    30
540 P=0:WHILE LNUM>LNUM(P) AND
     P<MAX:P=P+1:WEND:RETURN
560 MAX=MAX-1:FOR X=P TO MAX:L
    NUM(X)=LNUM(X+1):L$(X)=L$(
    X+1):NEXT:RETURN
580 MAX=MAX+1:FOR X=MAX TO P+1
     STEP -1:LNUM(X)=LNUM(X-1)
    :L$(X)=L$(X-1):NEXT:L$(P)=
    TEXT$:LNUM(P)=LNUM:RETURN
600 IF LEFT$(ARG$,1)<>CHR$(34)
     THEN 520 ELSE ARG$=MID$(A
    RG$,2)
610 IF RIGHT$(ARG$,1)=CHR$(34)
     THEN ARG$=LEFT$(ARG$,LEN(
    ARG$)-1)
620 IF SEL=0 AND INSTR(ARG$,".
    ")=0 THEN ARG$=ARG$+".BAS"
630 SEL=0:RETURN
640 CLOSE #1:CKFLAG=0:PRINT"St
    opped.":RETURN 150
650 PRINT "Error #";ERR:RESUME
     150
```

Program 3: Commodore Proofreader

By Philip Nelson, Assistant Editor

```
10 VEC=PEEK(772)+256*PEEK(773)
   :LO=43:HI=44
20 PRINT "AUTOMATIC PROOFREADE
   R FOR ";:IF VEC=42364 THEN
   {SPACE}PRINT "C-64"
30 IF VEC=50556 THEN PRINT "VI
   C-20"
40 IF VEC=35158 THEN GRAPHIC C
   LR:PRINT "PLUS/4 & 16"
50 IF VEC=17165 THEN LO=45:HI=
   46:GRAPHIC CLR:PRINT"128"
60 SA=(PEEK(LO)+256*PEEK(HI))+
   6:ADR=SA
70 FOR J=0 TO 166:READ BYT:POK
   E ADR,BYT:ADR=ADR+1:CHK=CHK
   +BYT:NEXT
80 IF CHK<>20570 THEN PRINT "*
    ERROR* CHECK TYPING IN DATA
    STATEMENTS":END
90 FOR J=1 TO 5:READ RF,LF,HF:
   RS=SA+RF:HB=INT(RS/256):LB=
   RS-(256*HB)
100 CHK=CHK+RF+LF+HF:POKE SA+L
   F,LB:POKE SA+HF,HB:NEXT
110 IF CHK<>22054 THEN PRINT "
   *ERROR* RELOAD PROGRAM AND
   {SPACE}CHECK FINAL LINE":EN
   D
120 POKE SA+149,PEEK(772):POKE
    SA+150,PEEK(773)
130 IF VEC=17165 THEN POKE SA+
   14,22:POKE SA+18,23:POKESA+
   29,224:POKESA+139,224
140 PRINT CHR$(147);CHR$(17);"
   PROOFREADER ACTIVE":SYS SA
150 POKE HI,PEEK(HI)+1:POKE (P
   EEK(LO)+256*PEEK(HI))-1,0:N
   EW
160 DATA 120,169,73,141,4,3,16
   9,3,141,5,3
170 DATA 88,96,165,20,133,167,
   165,21,133,168,169
180 DATA 0,141,0,255,162,31,18
   1,199,157,227,3
190 DATA 202,16,248,169,19,32,
   210,255,169,18,32
200 DATA 210,255,160,0,132,180
   ,132,176,136,230,180
210 DATA 200,185,0,2,240,46,20
   1,34,208,8,72
220 DATA 165,176,73,255,133,17
   6,104,72,201,32,208
230 DATA 7,165,176,208,3,104,2
   08,226,104,166,180
240 DATA 24,165,167,121,0,2,13
   3,167,165,168,105
250 DATA 0,133,168,202,208,239
   ,240,202,165,167,69
260 DATA 168,72,41,15,168,185,
   211,3,32,210,255
270 DATA 104,74,74,74,74,168,1
   85,211,3,32,210
280 DATA 255,162,31,189,227,3,
   149,199,202,16,248
290 DATA 169,146,32,210,255,76
   ,86,137,65,66,67
300 DATA 68,69,70,71,72,74,75,
   77,80,81,82,83,88
310 DATA 13,2,7,167,31,32,151,
   116,117,151,128,129,167,136
   ,137
```

Program 4: Apple Proofreader

By Tim Victor, Editorial Programmer

```
10 C = 0: FOR I = 768 TO 768 +
   68: READ A:C = C + A: POKE I
   ,A: NEXT
20 IF C < > 7258 THEN PRINT "ER
   ROR IN PROOFREADER DATA STAT
   EMENTS": END
30 IF PEEK (190 * 256) < > 76 T
   HEN POKE 56,0: POKE 57,3: CA
   LL 1002: GOTO 50
40 PRINT CHR$ (4);"IN#A$300"
50 POKE 34,0: HOME : POKE 34,1:
    VTAB 2: PRINT "PROOFREADER
   INSTALLED"
60 NEW
100 DATA 216,32,27,253,201,141
110 DATA 208,60,138,72,169,0
120 DATA 72,189,255,1,201,160
130 DATA 240,8,104,10,125,255
140 DATA 1,105,0,72,202,208
150 DATA 238,104,170,41,15,9
160 DATA 48,201,58,144,2,233
170 DATA 57,141,1,4,138,74
180 DATA 74,74,74,41,15,9
190 DATA 48,201,58,144,2,233
200 DATA 57,141,0,4,104,170
210 DATA 169,141,96
```

MLX Machine Language Entry Program For Commodore 64

Ottis Cowper, Technical Editor

"MLX" is a labor-saving utility that allows almost fail-safe entry of Commodore 64 machine language programs.

Type in and save some copies of MLX—you'll want to use it to enter future machine langauge (ML) programs from COMPUTE!. When you're ready to enter an ML program, load and run MLX. It asks you for a starting address and an ending address. These addresses appear in the article accompanying the MLX-format program listing you're typing.

If you're unfamiliar with machine language, the addresses (and all other values you enter in MLX) may appear strange. Instead of the usual decimal numbers you're accustomed to, these numbers are in *hexadecimal*—a base 16 numbering system commonly used by ML programmers. Hexadecimal—hex for short—includes the numerals 0–9 and the letters A–F. But don't worry—even if you know nothing about ML or hex, you should have no trouble using MLX.

After you enter the starting and ending addresses, you'll be offered the option of clearing the workspace. Choose this option if you're starting to enter a new listing. If you're continuing a listing that's partially typed from a previous session, don't choose this option.

A functions menu will appear. The first option in the menu is ENTER DATA. If you're just starting to type in a program, pick this. Press the E key, and type the first number in the first line of the program listing. If you've already typed in part of a program, type the line number where you left off typing at the end of the previous session (be sure to load the partially completed program before you resume entry). In any case, make sure the address you enter corresponds to the address of a line in the listing you are entering. Otherwise, you'll be unable to enter the data correctly. If you pressed E by mistake, you can return to the command menu by pressing RETURN alone when asked for the address. (You can get back to the menu from most options by pressing RETURN with no other input.)

Entering A Listing

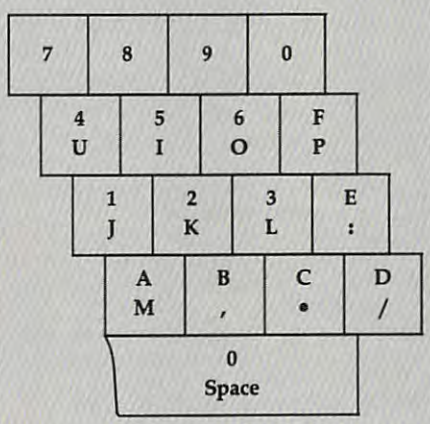
Once you're in Enter mode, MLX prints the address for each program line for you. You then type in all nine numbers on that line, beginning with the first two-digit number after the colon (:). Each line represents eight data bytes and a checksum. Although an MLX-format listing appears similar to the "hex dump" listings from a machine language monitor program, the extra checksum number on the end allows MLX to check your typing.

When you enter a line, MLX recalculates the checksum from the eight bytes and the address and compares this value to the number from the ninth column. If the values match, you'll hear a bell tone, the data will be added to the workspace area, and the prompt for the next line of data will appear. But if MLX detects a typing error, you'll hear a low buzz and see an error message. The line will then be redisplayed for editing.

Invalid Characters Banned

Only a few keys are active while you're entering data, so you may have to unlearn some habits. You *do not* type spaces between the columns; MLX automatically inserts these for you. You *do not* press RETURN after typing the last number in a line; MLX automatically enters and checks the line after you type the last digit.

Only the numerals 0–9 and the letters A–F can be typed in. If you press any other key (with some exceptions noted below), you'll hear a warning buzz. To simplify typing, a numeric keypad is now incorporated in the listing. The keypad is active only while entering data. Addresses must be entered with the normal letter and number keys. The figure below shows the keypad configuration:



MLX checks for transposed characters. If you're supposed to type in A0 and instead enter 0A, MLX will catch your mistake. There is one error that can slip past MLX: Because of the checksum formula used, MLX won't notice if you accidentally type FF in place of 00, and vice versa. And there's a very slim chance that you could garble a line and still end up with a combination of characters that adds up to the proper checksum. However, these mistakes should not occur if you take reasonable care while entering data.

Editing Features

To correct typing mistakes before finishing a line, use the INST/DEL key to delete the character to the left of the cursor. (The cursor-left key also deletes.) If you mess up a line really badly, press CLR/HOME to start the line over. The RETURN key is also active, but only before any data is typed on a line. Pressing RETURN at this point returns you to the command menu. After you type a character of data, MLX disables RETURN until the cursor returns to the start of a line. Remember, you can press CLR/HOME to quickly get to a line number prompt.

More editing features are available when correcting lines in which MLX has detected an error. To make corrections in a line that MLX has redisplayed for editing, compare the line on the screen with the one printed in the listing, then move the cursor to the mistake and type the correct key. The cursor left and right keys provide the normal cursor controls. (The INST/DEL key now works as an alternative cursor-left key.) You cannot move left beyond the first character in the line. If you try to move beyond the rightmost character, you'll reenter the line. During editing, RETURN is active; pressing it tells MLX to recheck the line. You can press the CLR/HOME key to clear the entire line if you want to start from scratch, or if you want to get to a line number prompt to use RETURN to get back to the menu.

Display Data

The second menu choice, DISPLAY DATA, examines memory and shows the contents in the same format as the program listing (including the checksum). When you press D, MLX asks you for a starting address. Be sure that the starting address you give corresponds to a line number in the listing. Otherwise, the checksum display will be meaningless. MLX displays program lines until it reaches the end of the program, at which point the menu is redisplayed. You can pause the display by pressing the space bar. (MLX finishes printing the current line before halting.) Press space again to

restart the display. To break out of the display and get back to the menu before the ending address is reached, press RETURN.

Other Menu Options

Two more menu selections let you save programs and load them back into the computer. These are SAVE FILE and LOAD FILE; their operation is quite straightforward. When you press S or L, MLX asks you for the filename. You'll then be asked to press either D or T to select disk or tape.

You'll notice the disk drive starting and stopping several times during a load or save. Don't panic; this is normal behavior. MLX opens and reads from or writes to the file instead of using the usual LOAD and SAVE commands. Disk users should also note that the drive prefix 0: is automatically added to the filename (line 750), so this should *not* be included when entering the name. This also precludes the use of @ for Save-with-Replace, so remember to give each version you save a different name.

Remember that MLX saves the entire workspace area from the starting address to the ending address, so the save or load may take longer than you might expect if you've entered only a small amount of data from a long listing. When saving a partially completed listing, make sure to note the address where you stopped typing so you'll know where to resume entry when you reload.

MLX reports the standard disk or tape error messages if any problems are detected during the save or load. (Tape users should bear in mind that Commodore computers are never able to detect errors during a save to tape.) MLX also has three special load error messages: INCORRECT STARTING ADDRESS, which means the file you're trying to load does not have the starting address you specified when you ran MLX; LOAD ENDED AT *address*, which means the file you're trying to load ends before the ending address you specified when you started MLX; and TRUNCATED AT ENDING ADDRESS, which means the file you're trying to load extends beyond the ending address you specified when you started MLX. If you see one of these messages and feel certain that you've loaded the right file, exit and rerun MLX, being careful to enter the correct starting and ending addresses.

The QUIT menu option has the obvious effect—it stops MLX and enters BASIC. The RUN/STOP key is disabled, so the Q option lets you exit the program without turning off the computer. (Of course, RUN/STOP–RESTORE also gets you out.) You'll be asked for verification; press Y to exit to BASIC, or any other key to return to the menu. After quitting, you can type RUN again and reenter MLX without losing your data, as long as you don't use the clear workspace option.

The Finished Product

When you've finished typing all the data for an ML program and saved your work, you're ready to see the results. The instructions for loading and using the finished product vary from program to program. Some ML programs are designed to be loaded and run like BASIC programs, so all you need to type is LOAD *"filename"*,8 for disk or LOAD *"filename"* for tape, and then RUN. Such programs will usually have a starting address of 0801 for the 64. Other programs must be reloaded to specific addresses with a command such as LOAD *"filename"*,8,1 for disk or LOAD *"filename"*,1,1 for tape, then started with a SYS to a particular memory address. On the Commodore 64, the most common starting address for such programs is 49152, which corresponds to MLX address C000. In either case, you should always refer to the article which accompanies the ML listing for information on loading and running the program.

An Ounce Of Prevention

By the time you finish typing in the data for a long ML program, you may have several hours invested in the project. Don't take chances—use our "Automatic Proofreader" to type the new MLX, and then test your copy *thoroughly* before first using it to enter any significant amount of data. Make sure all the menu options work as they should. Enter fragments of the program starting at several different addresses, then use the Display option to verify that the data has been entered correctly. And be sure to test the Save and Load options several times to ensure that you can recall your work from disk or tape. Don't let a simple typing error in the new MLX cost you several nights of hard work.

MLX For Commodore 64

```
SS 10 REM VERSION 1.1: LINES 8
      30,950 MODIFIED, LINES 4
      85-487 ADDED
EK 100 POKE 56,50:CLR:DIM IN$,
      I,J,A,B,A$,B$,A(7),N$
DM 110 C4=48:C6=16:C7=7:Z2=2:Z
      4=254:Z5=255:Z6=256:Z7=
      127
CJ 120 FA=PEEK(45)+Z6*PEEK(46)
      :BS=PEEK(55)+Z6*PEEK(56
      ):H$="0123456789ABCDEF"
SB 130 R$=CHR$(13):L$="{LEFT}"
      :S$=" ":D$=CHR$(20):Z$=
      CHR$(0):T$="{13 RIGHT}"
CQ 140 SD=54272:FOR I=SD TO SD
      +23:POKE I,0:NEXT:POKE
      {SPACE}SD+24,15:POKE 78
      8,52
FC 150 PRINT"{CLR}"CHR$(142)CH
      R$(8):POKE 53280,15:POK
      E 53281,15
EJ 160 PRINT T$" {RED}{RVS}
      {2 SPACES}[8 @]
      {2 SPACES}"SPC(28)"
      {2 SPACES}{OFF}{BLU} ML
      X II {RED}{RVS}
      {2 SPACES}"SPC(28)"
      {12 SPACES}{BLU}"
FR 170 PRINT"{3 DOWN}
      {3 SPACES}COMPUTE!'S MA
      CHINE LANGUAGE EDITOR
      {3 DOWN}"
JB 180 PRINT"{BLK}STARTING ADD
      RESS[4]";:GOSUB300:SA=A
      D:GOSUB1040:IF F THEN18
      0
GF 190 PRINT"{BLK}{2 SPACES}EN
      DING ADDRESS[4]";:GOSUB
      300:EA=AD:GOSUB1030:IF
      {SPACE}F THEN190
KR 200 INPUT"{3 DOWN}{BLK}CLEA
      R WORKSPACE [Y/N][4]";A
      $:IF LEFT$(A$,1)<>"Y"TH
      EN220
PG 210 PRINT"{2 DOWN}{BLU}WORK
      ING...";:FORI=BS TO BS+
      EA-SA+7:POKE I,0:NEXT:P
      RINT"DONE"
DR 220 PRINTTAB(10)"{2 DOWN}
      {BLK}{RVS} MLX COMMAND
      {SPACE}MENU {DOWN}[4]":
      PRINT T$"{RVS}E{OFF}NTE
      R DATA"
BD 230 PRINT T$"{RVS}D{OFF}ISP
      LAY DATA":PRINT T$"
      {RVS}L{OFF}OAD FILE"
JS 240 PRINT T$"{RVS}S{OFF}AVE
       FILE":PRINT T$"{RVS}Q
      {OFF}UIT{2 DOWN}{BLK}"
JH 250 GET A$:IF A$=N$ THEN250
HK 260 A=0:FOR I=1 TO 5:IF A$=
      MID$("EDLSQ",I,1)THEN A
      =I:I=5
FD 270 NEXT:ON A GOTO420,610,6
      90,700,280:GOSUB1060:GO
      TO250
EJ 280 PRINT"{RVS} QUIT ":INPU
      T"{DOWN}[4]ARE YOU SURE
       [Y/N]";A$:IF LEFT$(A$,
      1)<>"Y"THEN220
EM 290 POKE SD+24,0:END
JX 300 IN$=N$:AD=0:INPUTIN$:IF
      LEN(IN$)<>4THENRETURN
KF 310 B$=IN$:GOSUB320:AD=A:B$
      =MID$(IN$,3):GOSUB320:A
      D=AD*256+A:RETURN
PP 320 A=0:FOR J=1 TO 2:A$=MID
      $(B$,J,1):B=ASC(A$)-C4+
      (A$>"@")*C7:A=A*C6+B
JA 330 IF B<0 OR B>15 THEN AD=
      0:A=-1:J=2
GX 340 NEXT:RETURN
CH 350 B=INT(A/C6):PRINT MID$(
      H$,B+1,1);:B=A-B*C6:PRI
      NT MID$(H$,B+1,1);:RETU
      RN
RR 360 A=INT(AD/Z6):GOSUB350:A
      =AD-A*Z6:GOSUB350:PRINT
      ":";
BE 370 CK=INT(AD/Z6):CK=AD-Z4*
      CK+Z5*(CK>Z7):GOTO390
PX 380 CK=CK*Z2+Z5*(CK>Z7)+A
JC 390 CK=CK+Z5*(CK>Z5):RETURN
QS 400 PRINT"{DOWN}STARTING AT
      [4]";:GOSUB300:IF IN$<>
      N$ THEN GOSUB1030:IF F
      {SPACE}THEN400
EX 410 RETURN
HD 420 PRINT"{RVS} ENTER DATA
      {SPACE}":GOSUB400:IF IN
      $=N$ THEN220
JK 430 OPEN3,3:PRINT
SK 440 POKE198,0:GOSUB360:IF F
```

```
      THEN PRINT IN$:PRINT"
      {UP}{5 RIGHT}";
GC 450 FOR I=0 TO 24 STEP 3:B$
      =S$:FOR J=1 TO 2:IF F T
      HEN B$=MID$(IN$,I+J,1)
HA 460 PRINT"{RVS}"B$L$;:IF I<
      24THEN PRINT"{OFF}";
HD 470 GET A$:IF A$=N$ THEN470
FK 480 IF(A$>"/"ANDA$<":")OR(A
      $>"@"ANDA$<"G")THEN540
GS 485 A=-(A$="M")-2*(A$=",")-
      3*(A$=".")-4*(A$="/")-5
      *(A$="J")-6*(A$="K")
FX 486 A=A-7*(A$="L")-8*(A$=":
      ")-9*(A$="U")-10*(A$="I
      ")-11*(A$="O")-12*(A$="
      P")
CM 487 A=A-13*(A$=S$):IF A THE
      N A$=MID$("ABCD123E456F
      0",A,1):GOTO 540
MP 490 IF A$=R$ AND((I=0)AND(J
      =1)OR F)THEN PRINT B$;:
      J=2:NEXT:I=24:GOTO550
KC 500 IF A$="{HOME}" THEN PRI
      NT B$:J=2:NEXT:I=24:NEX
      T:F=0:GOTO440
MX 510 IF(A$="{RIGHT}")ANDF TH
      ENPRINT B$L$;:GOTO540
GK 520 IF A$<>L$ AND A$<>D$ OR
      ((I=0)AND(J=1))THEN GOS
      UB1060:GOTO470
HG 530 A$=L$+S$+L$:PRINT B$L$;
      :J=2-J:IF J THEN PRINT
      {SPACE}L$;:I=I-3
QS 540 PRINT A$;:NEXT J:PRINT
      {SPACE}S$;
PM 550 NEXT I:PRINT:PRINT"{UP}
      {5 RIGHT}";:INPUT#3,IN$
      :IF IN$=N$ THEN CLOSE3:
      GOTO220
QC 560 FOR I=1 TO 25 STEP3:B$=
      MID$(IN$,I):GOSUB320:IF
       I<25 THEN GOSUB380:A(I
      /3)=A
PK 570 NEXT:IF A<>CK THEN GOSU
      B1060:PRINT"{BLK}{RVS}
      {SPACE}ERROR: REENTER L
      INE [43]":F=1:GOTO440
HJ 580 GOSUB1080:B=BS+AD-SA:FO
      R I=0 TO 7:POKE B+I,A(I
      ):NEXT
QQ 590 AD=AD+8:IF AD>EA THEN C
      LOSE3:PRINT"{DOWN}{BLU}
      ** END OF ENTRY **{BLK}
      {2 DOWN}":GOTO700
GQ 600 F=0:GOTO440
QA 610 PRINT"{CLR}{DOWN}{RVS}
      {SPACE}DISPLAY DATA ":G
      OSUB400:IF IN$=N$ THEN2
      20
RJ 620 PRINT"{DOWN}{BLU}PRESS:
       {RVS}SPACE{OFF} TO PAU
      SE, {RVS}RETURN{OFF} TO
       BREAK[43]{DOWN}"
KS 630 GOSUB360:B=BS+AD-SA:FOR
      I=BTO B+7:A=PEEK(I):GOS
      UB350:GOSUB380:PRINT S$
      ;
CC 640 NEXT:PRINT"{RVS}";:A=CK
      :GOSUB350:PRINT
KH 650 F=1:AD=AD+8:IF AD>EA TH
      ENPRINT"{DOWN}{BLU}** E
      ND OF DATA **":GOTO220
KC 660 GET A$:IF A$=R$ THEN GO
      SUB1080:GOTO220
EQ 670 IF A$=S$ THEN F=F+1:GOS
      UB1080
AD 680 ONFGOTO630,660,630
CM 690 PRINT"{DOWN}{RVS} LOAD
      {SPACE}DATA ":OP=1:GOTO
      710
PC 700 PRINT"{DOWN}{RVS} SAVE
      {SPACE}FILE ":OP=0
RX 710 IN$=N$:INPUT"{DOWN}FILE
      NAME[43]";IN$:IF IN$=N$
      {SPACE}THEN220
PR 720 F=0:PRINT"{DOWN}{BLK}
      {RVS}T{OFF}APE OR {RVS}
      D{OFF}ISK: [43]";
FP 730 GET A$:IF A$="T"THEN PR
      INT"T{DOWN}":GOTO880
HQ 740 IF A$<>"D"THEN730
HH 750 PRINT"D{DOWN}":OPEN15,8
      ,15,"I0:":B=EA-SA:IN$="
      0:"+IN$:IF OP THEN810
SQ 760 OPEN 1,8,8,IN$+",P,W":G
      OSUB860:IF A THEN220
FJ 770 AH=INT(SA/256):AL=SA-(A
      H*256):PRINT#1,CHR$(AL)
      ;CHR$(AH);
PE 780 FOR I=0 TO B:PRINT#1,CH
      R$(PEEK(BS+I));:IF ST T
      HEN800
FC 790 NEXT:CLOSE1:CLOSE15:GOT
      O940
GS 800 GOSUB1060:PRINT"{DOWN}
      {BLK}ERROR DURING SAVE:
      [43]":GOSUB860:GOTO220
MA 810 OPEN 1,8,8,IN$+",P,R":G
      OSUB860:IF A THEN220
GE 820 GET#1,A$,B$:AD=ASC(A$+Z
      $)+256*ASC(B$+Z$):IF AD
      <>SA THEN F=1:GOTO850
RX 830 FOR I=0 TO B:GET#1,A$:P
      OKE BS+I,ASC(A$+Z$):IF(
      I<>B)AND ST THEN F=2:AD
      =I:I=B
FA 840 NEXT:IF ST<>64 THEN F=3
FQ 850 CLOSE1:CLOSE15:ON ABS(F
      >0)+1 GOTO960,970
SA 860 INPUT#15,A,A$:IF A THEN
       CLOSE1:CLOSE15:GOSUB10
      60:PRINT"{RVS}ERROR: "A
      $
GQ 870 RETURN
EJ 880 POKE183,PEEK(FA+2):POKE
      187,PEEK(FA+3):POKE188,
      PEEK(FA+4):IFOP=0THEN92
      0
HJ 890 SYS 63466:IF(PEEK(783)A
      ND1)THEN GOSUB1060:PRIN
      T"{DOWN}{RVS} FILE NOT
      {SPACE}FOUND ":GOTO690
CS 900 AD=PEEK(829)+256*PEEK(8
      30):IF AD<>SA THEN F=1:
      GOTO970
SC 910 A=PEEK(831)+256*PEEK(83
      2)-1:F=F-2*(A<EA)-3*(A>
      EA):AD=A-AD:GOTO930
KM 920 A=SA:B=EA+1:GOSUB1010:P
      OKE780,3:SYS 63338
JF 930 A=BS:B=BS+(EA-SA)+1:GOS
      UB1010:ON OP GOTO950:SY
      S 63591
AE 940 GOSUB1080:PRINT"{BLU}**
       SAVE COMPLETED **":GOT
      O220
XP 950 POKE147,0:SYS 63562:IF
      {SPACE}ST>0 THEN970
FR 960 GOSUB1080:PRINT"{BLU}**
       LOAD COMPLETED **":GOT
      O220
DP 970 GOSUB1060:PRINT"{BLK}
      {RVS}ERROR DURING LOAD:
      {DOWN}[43]":ON F GOSUB98
      0,990,1000:GOTO220
PP 980 PRINT"INCORRECT STARTIN
      G ADDRESS (";:GOSUB360:
      PRINT")":RETURN
GR 990 PRINT"LOAD ENDED AT ";:
      AD=SA+AD:GOSUB360:PRINT
       D$:RETURN
FD 1000 PRINT"TRUNCATED AT END
       ING ADDRESS":RETURN
RX 1010 AH=INT(A/256):AL=A-(AH
      *256):POKE193,AL:POKE1
      94,AH
FF 1020 AH=INT(B/256):AL=B-(AH
      *256):POKE174,AL:POKE1
      75,AH:RETURN
FX 1030 IF AD<SA OR AD>EA THEN
      1050
HA 1040 IF(AD>511 AND AD<40960
      )OR(AD>49151 AND AD<53
      248)THEN GOSUB1080:F=0
      :RETURN
HC 1050 GOSUB1060:PRINT"{RVS}
      {SPACE}INVALID ADDRESS
       {DOWN}{BLK}":F=1:RETU
      RN
AR 1060 POKE SD+5,31:POKE SD+6
      ,208:POKE SD,240:POKE
      {SPACE}SD+1,4:POKE SD+
      4,33
DX 1070 FOR S=1 TO 100:NEXT:GO
      TO1090
PF 1080 POKE SD+5,8:POKE SD+6,
      240:POKE SD,0:POKE SD+
      1,90:POKE SD+4,17
AC 1090 FOR S=1 TO 100:NEXT:PO
      KE SD+4,0:POKE SD,0:PO
      KE SD+1,0:RETURN
```

©

Attention Programmers

COMPUTE! magazine is currently looking for quality articles on Commodore, Atari, Apple, and IBM computers (including the Commodore Amiga and Atari ST). If you have an interesting home application, educational program, programming utility, or game, submit it to COMPUTE!, P.O. Box 5406, Greensboro, NC 27403. Or write for a copy of our "Writer's Guidelines."

All the programs in this issue are available on the ready-to-load *COMPUTE! Disk*. To order a one-year (four-disk) subscription, call toll free **800-247-5470** (in IA 800-532-1272). Please specify which computer you are using.

MLX Machine Language Entry Program For Apple

Tim Victor, Editorial Programmer

To make it easier to enter machine language programs into your computer without typos, COMPUTE! is introducing its "MLX" entry program for the Apple II series. It's our best MLX yet. It runs on the II, II+, IIe, and IIc, and with either DOS 3.3 or ProDOS.

A machine language (ML) program is usually listed as a long series of numbers. It's hard to keep your place and even harder to avoid making mistakes as you type in the listing, since an incorrect line looks almost identical to a correct one. To make error-free entry easier, COMPUTE! generally lists ML programs for Commodore and Atari computers in a format designed to be typed in with a utility called "MLX." The MLX program uses a checksum system to catch typing errors almost as soon as they happen.

Apple MLX checks your typing on a line-by-line basis. It won't let you enter invalid characters or let you continue if there's a mistake in a line. It won't even let you enter a line or digit out of sequence. Best of all, you don't have to know anything about machine language to enter ML programs with MLX. Apple MLX makes typing ML programs almost foolproof.

Using Apple MLX

Type in and save some copies of Apple MLX on disk (you'll want to use MLX to enter future ML programs in COMPUTE!). It doesn't matter whether you type it in on a disk formatted for DOS 3.3 or ProDOS. Programs entered with Apple MLX, however, must be saved to a disk formatted with the same operating system as Apple MLX itself.

If you have an Apple IIe or IIc, make sure that the key marked CAPS LOCK is in the down position. Type RUN. You'll be asked for the starting and ending addresses of the ML program. These values vary for each program, so they're given at the beginning of the ML program listing and in the program's accompanying article. Find them and type them in.

The next thing you'll see is a menu asking you to select a function. The first is (E)NTER DATA. If you're just starting to type in a program, pick this. Press the E key, and the program asks for the address where you want to begin entering data. Type the first number in the first line of the program listing if you're just starting, or the line number where you left off if you've already typed in part of a program. Hit the RETURN key and begin entering the data.

Once you're in Enter mode, Apple MLX prints the address for each program line for you. You then type in all nine numbers on that line, beginning with the first two-digit number after the colon (:). Each line represents eight bytes and a checksum. When you enter a line and hit RETURN, Apple MLX recalculates the checksum from the eight bytes and the address. If you enter more or less than nine numbers, or the checksum doesn't exactly match, Apple MLX erases the line you just entered and prompts you again for the same line.

Invalid Characters Banned

Apple MLX is fairly flexible about how you type in the numbers. You can put extra spaces between numbers or leave the spaces out entirely, compressing a line into 18 keypresses. Be careful not to put a space between two digits in the middle of a number. Apple MLX will read two single-digit numbers instead of one two-digit number (F 6 means F and 6, not F6).

You can't enter an invalid character with Apple MLX. Only the numerals 0–9 and the letters A–F can be typed in. If you press any other key (with some exceptions noted below), nothing happens. This safeguards against entering extraneous characters. Even better, Apple MLX checks for transposed characters. If you're supposed to type in A0 and instead enter 0A, Apple MLX will catch your mistake.

Apple MLX also checks to make sure you're typing in the right line. The address (the number to the left of the colon) is part of the checksum recalculation. If you accidentally skip a line and try to enter incorrect values, Apple MLX won't let you continue. Just make sure you enter the correct starting address; if you don't, you won't be able to enter any of the following lines. Apple MLX will stop you.

Editing Features

Apple MLX also includes some editing features. The left- and right-arrow keys allow you to back up and go forward on the line that you are entering, so you can retype data. Pressing the CONTROL (CTRL) and D keys at the same time (*delete*) removes the character under the cursor, shortening the line by one character. Pressing CTRL-I (*insert*) puts a space under the cursor and shifts the rest of the line to the right, making the line one character longer. If the cursor is at the right end of the line, neither CTRL-D nor CTRL-I has any effect.

When you've entered the entire listing (up to the ending address that you specified earlier), Apple MLX automatically leaves Enter mode and redisplays the functions menu. If you want to leave Enter mode before then, press the RETURN key when Apple MLX prompts you with a new line address. (For instance, you may want to leave Enter mode to enter a program listing in more than one sitting; see below.)

Display Data

The second menu choice, (D)ISPLAY DATA, examines memory and shows the contents in the same format as the program listing. You can use it to check your work or to see how far you've gotten. When you press D, Apple MLX asks you for a starting address. Type in the address of the first line you want to see and hit RETURN. Apple MLX displays program lines until you press any key or until it reaches the end of the program.

Save And Load

Two more menu selections let you save programs on disk and load them back into the computer. These are (S)AVE FILE and (L)OAD FILE. When you press S or L, Apple MLX asks you for the filename. The first time you save an ML program, the name you assign will be the program's filename on the disk. If you press L and specify a filename that doesn't exist on the disk, you'll see a disk error message.

If you're not sure why a disk error has occurred, check the drive. Make sure there's a formatted disk in the drive and that it was formatted by the same operating system you're using for Apple MLX (ProDOS or DOS 3.3). If you're trying to save a file and see an error message, the disk might be full. Either save the file on another disk or quit Apple MLX (by pressing the Q key), delete an old file or two, then run Apple MLX again. Your typing should still be safe in memory.

Apple MLX: Machine Language Entry Program

For instructions on entering this program, please refer to "COMPUTE!'s Guide to Typing In Programs" elsewhere in this issue.

```
100 N = 9: HOME : NORMAL : PRIN
    T "APPLE MLX": POKE 34,2: O
    NERR  GOTO 610
110 VTAB 1: HTAB 20: PRINT "STA
    RT ADDRESS";: GOSUB 530: IF
     A = 0 THEN  PRINT  CHR$ (7
    ): GOTO 110
120 S = A
130 VTAB 2: HTAB 20: PRINT "END
     ADDRESS  ";: GOSUB 530: IF
     S >  = A OR A = 0 THEN  PR
    INT  CHR$ (7): GOTO 130
140 E = A
150 PRINT : PRINT "CHOOSE: (E)NT
    ER DATA";: HTAB 22: PRINT "
    (D)ISPLAY DATA": HTAB 8: PR
    INT "(L)OAD FILE  (S)AVE FI
    LE  (Q)UIT": PRINT
160 GET A$: FOR I = 1 TO 5: IF
    A$ < > MID$ ("EDLSQ",I,1) T
    HEN  NEXT : GOTO 160
170 ON I GOTO 270,220,180,200:
    POKE 34,0: END
180 INPUT "FILENAME: ";A$: IF A
    $ <  > "" THEN  PRINT  CHR$
     (4);"BLOAD";A$;",A";S
190 GOTO 150
200 INPUT "FILENAME: ";A$: IF A
    $ <  > "" THEN  PRINT  CHR$
     (4);"BSAVE";A$;",A";S;",L"
    ;E - S
210 GOTO 150
220 GOSUB 590: IF B = 0 THEN 15
    0
230 FOR B = B TO E STEP 8:L = 4
    :A = B: GOSUB 580: PRINT A$
    ;": ";:L = 2
240 FOR F = 0 TO 7:V(F + 1) = P
    EEK (B + F): NEXT : GOSUB 5
    60:V(9) = C
250 FOR F = 1 TO N:A = V(F): GO
    SUB 580: PRINT A$" ";: NEXT
     : PRINT : IF  PEEK (49152)
     < 128 THEN  NEXT
260 POKE 49168,0: GOTO 150
270 GOSUB 590: IF B = 0 THEN 15
    0
280 FOR B = B TO E STEP 8
290 HTAB 1:A = B:L = 4: GOSUB 5
    80: PRINT A$;": ";: CALL 64
    668:A$ = "":P = 0: GOSUB 33
    0: IF L = 0 THEN 150
300 GOSUB 470: IF F < > N THEN
    PRINT CHR$ (7);: GOTO 290
310 IF N = 9 THEN GOSUB 560: IF
     C < > V(9) THEN PRINT CHR$
    (7);: GOTO 290
320 FOR F = 1 TO 8: POKE B + F
    - 1,V(F): NEXT : PRINT : NE
    XT : GOTO 150
330 IF LEN (A$) = 33 THEN A$ =
    O$:P = O: PRINT CHR$ (7);
340 L = LEN (A$):O$ = A$:O = P:
    L$ = "": IF P > 0 THEN L$ =
      LEFT$ (A$,P)
350 R$ = "": IF P < L - 1 THEN
    R$ =  RIGHT$ (A$,L - P - 1)
360 HTAB 7: PRINT L$;: FLASH :
    IF P < L THEN PRINT MID$ (A
    $,P + 1,1);: NORMAL : PRINT
    R$;
370 PRINT " ";: NORMAL
380 K = PEEK (49152): IF K < 12
    8 THEN 380
390 POKE 49168,0:K = K - 128
400 IF K = 13 THEN HTAB 7: PRIN
    T A$;" ";: RETURN
410 IF K = 32 OR K > 47 AND K <
     58 OR K > 64 AND K < 71 TH
    EN A$ = L$ + CHR$ (K) + R$:
    P = P + 1
420 IF K = 4 THEN A$ = L$ + R$
430 IF K = 9 THEN A$ = L$ + " "
     +  MID$ (A$,P + 1,1) + R$
440 IF K = 8 THEN P = P - (P >
    0)
450 IF K = 21 THEN P = P + (P <
    L)
460 GOTO 330
470 F = 1:D = 0: FOR P = 1 TO L
    EN (A$):C$ = MID$ (A$,P,1):
     IF F > N AND C$ < > " " TH
    EN  RETURN
480 IF C$ < > " " THEN  GOSUB 5
    20:V(F) = J + 16 * (D = 1)
    * V(F):D = D + 1
490 IF D > 0 AND C$ = " " OR D
    = 2 THEN D = 0:F = F + 1
500 NEXT : IF D = 0 THEN F = F
    - 1
510 RETURN
520 J = ASC (C$):J = J - 48 - 7
    * (J > 64): RETURN
530 A = 0: INPUT A$:A$ = LEFT$
    (A$,4): IF LEN (A$) = 0 THE
    N RETURN
540 FOR P = 1 TO LEN (A$):C$ =
    MID$ (A$,P,1): IF C$ < "0"
    OR C$ > "9" AND C$ < "A" OR
     C$ > "Z" THEN A = 0: RETUR
    N
550 GOSUB 520:A = A * 16 + J: N
    EXT : RETURN
560 C = INT (B / 256):C = B - 2
    54 * C - 255 * (C > 127):C
    = C - 255 * (C > 255)
570 FOR F = 1 TO 8:C = C * 2 -
    255 * (C > 127) + V(F):C =
    C - 255 * (C > 255): NEXT :
     RETURN
580 I = FRE (0):A$ = "": FOR I
    = 1 TO L:T =  INT (A / 16):
    A$ =  MID$ ("0123456789ABCD
    EF",A - 16 * T + 1,1) + A$:
    A = T: NEXT : RETURN
590 PRINT "FROM ADDRESS ";: GOS
    UB 530: IF S > A OR E < A O
    R A = 0 THEN B = 0: RETURN
600 B = S + 8 * INT ((A - S) /
    8): RETURN
610 PRINT "DISK ERROR": GOTO 15
    0
```

©

Classified

SOFTWARE

COMMODORE: TRY BEFORE YOU BUY. Best selling games, utilities, educational, + classics and new releases. 100's of titles. Visa/MC. Free brochure. RENT-A-DISC, Frederick Bldg. #345, Hunt'n, WV 25701 (304) 529-3232

FREE APPLE SOFTWARE
Over 1000 Public Domain Programs on 50 diskettes. $5 each plus $1 for shipping per order. Send $1 for catalog, refundable with order.
C&H ENTERPRISES
PO Box 29243, Memphis, TN 38127

TI-99/4A QUALITY SOFTWARE for Business, Home and Entertainment ** Bonus Software Offer! ** Send for FREE catalog to MICRO-BIZ HAWAII, BOX 1108, PEARL CITY, HI 96782

DISCOUNT SOFTWARE for most computers. FREE CATALOG. Sale: 5.25" DSDD Disks 25 for $13.95 ppd. WMJ DATA SYSTEMS-C, 4 Butterfly Dr., Hauppauge, NY 11788

Cheap Software/IBM,PCjr, Compatibles. 5.25 or 3.5 disk. For catalog of diskettes available for education, games, business, etc., write: Morning Star Industry, P.O. Box 3095, Ann Arbor, MI 48106

FREE! PUBLIC DOMAIN SOFTWARE! MS-DOS, IBM & Compatibles - Save $$$ @ $3.50 per disk! Free flyer: AP-JP, Inc, PO Box 1155, W. Babylon, NY 11704

FREE SOFTWARE for C64, C128, IBM & CPM send SASE for info (specify computer) to: PUBLIC DOMAIN USERS GROUP PO Box 1442-A1, Orange Park, FL 32067

MILLION $ LOTTO BUSTER PROGRAM GUARANTEED. SCIENTIFIC. HITS MILLIONS. AUTHOR HITS WEEKLY. NOT A R/N GEN. Z-WAY, PO BOX 9017, CANTON, OH 44711

IBM PUBLIC DOMAIN SOFTWARE $3 PER DISK. Send SASE for catlg. Hundreds of disks to choose from. Excellent service. Two disk sizes now available: 5.25" @ $3.00 and 3.5" @ $5.00. Send for your list.
JDX/C, PO BOX 1561, CORONA, CA 91718

HEY AMIGO! PD software for AMIGA!
Games, Graphics, Utilities, More! Over 70 disks available, only $5.95 ea! SASE for cat. AMYWARE, POB 19474, Jacksonville, FL 32245

FREE PUBLIC DOMAIN SOFTWARE - Request free catalog or send $2 for sample disk & catalog (refundable). Apple, C64, C128 (specify) CALOKE IND., Box 18477, KC, MO 64133

$$ WIN $$
with our
SPORTS
Handicapping
Software

THOROUGHBRED/HARNESS Handicapping..$29.95, enhanced..$49.95. GREYHOUND Handicapping..$29.95, enhanced..$49.95. Up to 13 entries analyzed: class, speed, times, odds, track, post, stretch, & finish positions. Pro Football System..$39.95. Pro Basketball System.. $49.95. Specify disk/tape: Apple II+ce, Atari, IBM PC, COM 64/128, TI, TRS-80 Mod 3/4, Mod 100/200, Color. Add $2 p&h. MC/VISA accepted. Free info. SOFTWARE EXCHANGE, P.O. Box 5382-CP, W. Bloomfield, MI 48033. (313) 626-7208. Orders: 1-800-527-9467.

SOFTWARE AT WHOLESALE PRICES!
Please specify computer type. Send $1 for catalog to: ACE SOFTWARE CO. 2702 So. 25th., St. Joseph, MO 64503 $2 discount on first purchase!

More than 200 great ML routines for 64 and 128, ready to add to your own programs, in COMPUTE! Books' MACHINE LANGUAGE ROUTINES FOR THE COMMODORE 64/128. Explanations, uses, commented source code. 585 pages, $18.95. Check your local bookstore or call (800) 346-6767

FREE SOFTWARE FOR IBM PC COMPATIBLE SEND S.A.S.E. FOR CATALOG G&D DISTRIBUTORS, PO BOX 1901 BAY CITY, MI 48706

FREE LIST: CARTRIDGES/SOFTWARE FOR: Coleco, Atari 2600, 5200, 7800, Nintendo, Adam, XL, XE, Intellivision: Modem: 414-265-5149, 24 hrs, Villa Video, Box 17085, Milw, WI 53217

MUTUAL FUND PROGRAM (MFP) saves trouble and money by figuring lowest taxable capital gain on withdrawals. Keeps funds, bank accounts, IRA's, etc, all in one record. Disk and manual, C64:$50, IBM PC:$100. Bernhardt MFP Software, P.O. Box 124, Great Neck, NY 11022

MATH MADE EASY
IBM PC & COMPATIBLE. For grades 1–7
Easy online instruction, complete user manual.
ARITH I: Add, Subtract
ARITH II: Multiply, Divide
FRACTIONS I: Simplify, Compare
FRACTIONS II: Add, Subtract, Multiply, Divide
One-time offer: 4 for $50, 2 for $27, 1 for $15
SOFTWARE CONSULTANTS, INC.
P.O. Box 4370, Denver, CO 80204 USA

TANDY 1000 PROGRAMS AND NEWSLETTER
Send for free information on educational & entertainment programs & newsletter. Soda Pop Software, POB 653, Kenosha, WI 53141

***** NUTRITIONAL GUIDANCE SYSTEM *****
Atari 8-Bit (Apple, IBM, ST in prog/call) 1300 common foods + Custom food entry, tracks 9 food values, easy & fast $24.95. Also avail: GREENHOUSE (houseplants DB) & GRADE-STAR $19.95 ORDER/?'s: LJ Software PO Box 7190 SLC, UT 84107 (801)268-6468. Free US Shipping, $5 backup, Prof. Quality.

MISCELLANEOUS

SAFEWARE INSURES COMPUTERS against fire, theft, & power surges for as little as $39. Call Safeware, The Insurance Agency Inc. at 800/848-3469, Columbus, Ohio.

SX-64 OWNERS! STAND UP AND BE COUNTED! Join the USA SX-64 USERS GROUP! Monthly newsletter, discounts on products and free software. Yearly dues $15. USA SX-64 USERS GROUP, 10 Cornwall St., Bost, MA 02130

COMMODORE & TI-99/4A CATALOGS! Hardware-Software-Supplies-Accessories Send $2.00, specify computer to ORDER 99, P.O. Box 791497, San Antonio, TX 78279

$ WIN $
THE LOTTO
With Your Computer!

Forget random numbers. This program for home computers does an actual analysis of the past winning numbers. This amazing program will quickly provide you with all the data you need to predict which numbers will likely come up in subsequent drawings. All consistent lottery winners use some kind of system based on the past winners. Using the real power of your computer gives you a definite edge. It's menu driven and all you do is add the latest winners each week and the program does the rest in seconds. On screen or printer it shows hot and cold numbers, frequency, groups, sums-of-digits, odd/even, wheels numbers and more. No thick manual to read. It even has a built-in tutorial.

Ask your software dealer or call or write:

SOFT BYTE

SOFT-BYTE
P.O. Box 556 F. Park
Dayton, Ohio 45405
(513) 233-2200

THE LOTTO PROGRAM is designed for all 6 & 7 draw lotto games (up to 49 numbers)!

DON'T PLAY LOTTO WITHOUT IT!

APPLE & M/S DOS IBM 24.95
COMMODORE & ATARI 21.95
TRS-80 & MOD III, IV 21.95
MACINTOSH (super version) 29.95

Please add $2.00 shipping/ handling. Fast service on charge cards.

MasterCard VISA

COMPUTE! Classified is a low-cost way to tell over 350,000 microcomputer owners about your product or service.

Rates: $25 per line, minimum of four lines. Any or all of the first line set in capital letters at no charge. Add $15 per line for boldface words, or $50 for the entire ad set in boldface (any number of lines.) Inquire about display rates.
Terms: Prepayment is required. Check, money order, American Express, Visa, or MasterCard is accepted. Make checks payable to COMPUTE! Publications.
Form: Ads are subject to publisher's approval and must be either typed or legibly printed. One line equals 40 letters and spaces between words. Please underline words to be set in boldface.
General Information: Advertisers using post office box numbers in their ads must supply permanent address and telephone numbers. Ad will appear in next available issue after receipt.
Closing: 10th of the third month preceding cover date (e.g., June issue closes March 10th). Send order and remittance to: Harry Blair, Classified Manager, COMPUTE!, P.O. Box 5406, Greensboro, NC 27403. To place an ad by phone, call Harry Blair at (919) 275-9809.
Notice: COMPUTE! Publications cannot be responsible for offers or claims of advertisers, but will attempt to screen out misleading or questionable copy.

Advertisers Index

Reader Service Number/Advertiser	Page
102 Abacus	25
103 Acorn of Indiana	130
104 Adapso	122
105 Air Force	1
C.O.M.B. Direct Marketing Corp.	80
106 CompuServe	9
107 ComputAbility	115
108 Computer Direct	28–29
109 Computer Mail Order	18–19
110 Covox, Inc.	132
111 Dresselhaus	45
112 Electronic Arts	2
113 Electronic Arts	IBC
114 GE Information Services	5
115 Heuristic Research, Inc.	130
116 Indus Tool	109
117 Lyco Computer	32–35
118 Marathon Software	105
119 Montgomery Grant	119
120 National Scouting Museum	39
121 Near-Future Computer	75
NRI Schools	48–49
122 Origin Systems, Inc.	IFC
123 Parsons Technology	73
124 PC-SIG	75
125 Precision Data Products	75
126 Precision Images, Inc.	109
127 Soft Byte	131
128 Softronics	12
129 Software Discounters of America	117
130 ST Station	113
131 subLOGIC Corporation	7
132 Video Technology Computers, Inc.	BC

Classified Ads	131
COMPUTE! Apple Applications and Programming Contest Announcement	22–23
COMPUTE! Books' Commodore 128 & 64 Selection	31
COMPUTE! Books' Flight Simulator Books	41
COMPUTE! Books' Programming Books for the Atari ST	27
COMPUTE! Books' Quick & Easy Books	111
COMPUTE! Disk Subscription	65
COMPUTE!'s PC Magazine and Programming Contest Announcement	68–69
COMPUTE! Subscription	17

COMPUTE!

A WORTHWHILE INVESTMENT

You may have easily spent thousands on your computer. Whatever you spent, it was a wise and worthwhile investment! Now make your computer even more valuable by subscribing to Compute!, the leading magazine of home, educational and recreational computing.

☐ **1 Year/$24** ☐ **2 Years/$45** J8721

Name _____

Address _____

City _____ State _____ Zip _____

☐Payment Enclosed ☐ Bill Me Later

For faster service call: 1-800-247-5470

For Foreign & Canadian Subscribers, please add $6 (U.S.) per year postage.

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY MAIL
FIRST CLASS PERMIT NO. 7478 DES MOINES, IA

POSTAGE WILL BE PAID BY ADDRESSEE

COMPUTE!

P.O. Box 10955
Des Moines, IA 50347-0955

COMPUTE!'s Disk

SPECIAL INTRODUCTORY OFFER

YES! I want to save time and money. Please enter my quarterly subscription to the following COMPUTE! Disk:

☐ Commodore ☐ Apple ☐ Atari ☐ IBM

☐ Save 33% off the single issue price. 1 year subscription, $39.95

☐ Save even more! 2 year subscription, $69.95

☐ Payment enclosed (check or money order)
☐ Charge ☐ MasterCard ☐ Visa

Or Call: 1-800-247-5470

Acct. No. _____ Exp. Date _____

Signature _____

Name _____

Address _____

City _____ State _____ Zip _____

Save 33%

(Outside U.S. and Canada, add $9.00 per year for shipping and handling.)

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY MAIL

FIRST CLASS PERMIT NO. 7551 DES MOINES, IA

POSTAGE WILL BE PAID BY ADDRESSEE

COMPUTE!'s DISK
P.O. BOX 10036
DES MOINES, IA 50347-0036

RUSH POSTAGE-PAID CARD FOR YOUR FREE CATALOG

- **COMPUTER ELECTRONICS** training prepares you to service all computers as you build your own 16-bit IBM PC compatible computer. Total system program includes disk drive, test equipment, bundled software, and NRI Discovery Lab.
- **TV/VIDEO/AUDIO SERVICING** includes training with a state-of-the-art 27" high resolution broadcast stereo TV for learning troubleshooting and professional bench techniques.
- **SATELLITE ELECTRONICS** training gives you the skills to service both consumer and commercial satellite earth station equipment as you assemble your own home satellite TV system.
- **ROBOTICS** training features remote-controlled, mobile, fully programmable robot you build, experiment with, and keep along with other test equipment.

CHECK ☑ ONE

Send me your free catalog on NRI Training in:

- ☐ Computer Electronics
- ☐ TV/Video/Audio Servicing
- ☐ Robotics
- ☐ Satellite Electronics
- ☐ Digital Electronics Servicing
- ☐ Data Communications
- ☐ Electronic Design Technology
- ☐ Industrial Electronics
- ☐ Communications Electronics
- ☐ Basic Electronics
- ☐ Building Construction and Remodeling
- ☐ Automotive Servicing
- ☐ Small Engine Servicing
- ☐ Electrician
- ☐ Air Conditioning, Heating, and Refrigeration
- ☐ Locksmithing and Electronic Security
- ☐ Telephone Servicing
- ☐ Appliance Servicing
- ☐ Photography
- ☐ Bookkeeping and Accounting

Name *(Please Print)* Age

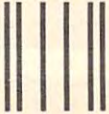
Street

City State Zip

For career courses approved under G.I. Bill ☐ Check for details.

Accredited by the National Home Study Council

198-087



NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY MAIL

FIRST CLASS MAIL PERMIT NO. 10008 WASHINGTON, D.C.

POSTAGE WILL BE PAID BY ADDRESSEE

NRI Schools
McGraw Hill Continuing
Education Center
3939 Wisconsin Avenue
Washington, D.C. 20077-9265

COMPUTE!'s
FREE Reader Information Service

Use these cards to request FREE information about the products advertised in this issue. Clearly print or type your full name and address. Only one card should be used per person. Circle the numbers that correspond to the key number appearing in the advertisers index.

Send in the card and the advertisers will receive your inquiry. Although every effort is made to insure that only advertisers wishing to provide product information have reader service numbers, COMPUTE! cannot be responsible if advertisers do not provide literature to readers.

Please use these cards *only* for subscribing or for requesting product information. Editorial and customer service inquiries should be addressed to: COMPUTE!, P.O. Box 5406, Greensboro, NC 27403. Check the expiration date on the card to insure proper handling.

Use these cards and this address only for COMPUTE!'s Reader Information Service. Do not send with payment in any form.

COMPUTE!

101	102	103	104	105	106	107	108	109	110	111	112	113	114	115	116	117
118	119	120	121	122	123	124	125	126	127	128	129	130	131	132	133	134
135	136	137	138	139	140	141	142	143	144	145	146	147	148	149	150	151
152	153	154	155	156	157	158	159	160	161	162	163	164	165	166	167	168
169	170	171	172	173	174	175	176	177	178	179	180	181	182	183	184	185
186	187	188	189	190	191	192	193	194	195	196	197	198	199	200	201	202
203	204	205	206	207	208	209	210	211	212	213	214	215	216	217	218	219
220	221	222	223	224	225	226	227	228	229	230	231	232	233	234	235	236
237	238	239	240	241	242	243	244	245	246	247	248	249	250	251	252	253

Circle 101 for a one year new U.S. subscription to COMPUTE!: you will be billed for $24.

Please let us know. Do you own: plan to buy:

- ☐ 270 Apple ☐ 271
- ☐ 272 Atari ☐ 273
- ☐ 274 Commodore ☐ 275
- ☐ 276 IBM ☐ 277
- ☐ 278 TI-99/4A ☐ 279
- ☐ 280 Other (specify model) ☐ 281

Please print or type name and address.
Limit one card per person.

Name _____

Address _____

City _____

State/Province _____ Zip _____

Country _____

Phone _____

Please Include ZIP Code

Expiration Date 10/31/87 CO887

SUBSCRIBE TO COMPUTE!

For Fastest Service, Call Our **Toll-Free** US Order Line
800-247-5470

☐ $24.00 One Year US Subscription
☐ $45.00 Two Year US Subscription

Name _____

Address _____

City _____ State _____ Zip _____

☐ Payment Enclosed ☐ Bill me
Charge my: ☐ VISA ☐ MasterCard ☐ American Express
Account No. _____ Expires ____/____

Your subscription will begin with the next available issue. Please allow 4-6 weeks for delivery of first issue. Subscription rate subject to change at any time. Outside the U.S.A. please add $6 for each subscription.

J8747

Place
Stamp
Here

COMPUTE! Reader Service
P.O. Box 2141
Radnor, PA 19089

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY MAIL
FIRST CLASS PERMIT NO. 7478 DES MOINES, IA

POSTAGE WILL BE PAID BY ADDRESSEE

COMPUTE!

P.O. Box 10955
Des Moines, IA 50347-0955

"The Bard is Back!"

From impossible dungeons and split-second snares, the Bard and his party emerge. The Sceptre, so long forgotten, gleams with power like an exploding sun. Even Phenglei Kai, the ancient archmage, bows his head in awe.

"I smell serpents!" Slipfinger squeals, stealing away like the thief he is. Two archdragons slither out of the ground, their eyes burning with the relentless fury of treasure lost.

Protected behind the flame lizards, beyond the reach of normal weapons, a cackling wizard begins the eerie chants of a death spell. A spell that can finish the Bard and his party.

The time has come to battle-test the magic of the Destiny Wand – and reveal the awesome powers of The Destiny Knight.™

The Best Ever Dungeon Role-Playing Game

- 50% bigger than Bard's Tale.™
- An all-new story line.
- Six cities and a huge overland wilderness to explore.
- Dozens of new spells – 79 spells in all.
- New real-time dungeon puzzles. You have to get through them before the clock stops ticking.
- Summon and name monsters to become a permanent part of your party.
- More strategy in combat encounters – the weapons and spells you choose depend on the enemy's distance.
- A bank and casino.
- A starter-dungeon for building up your low-level characters.
- 6 guilds for easier game saving.
- Optional use of Bard's Tale characters. Bard's Tale experience not required.
- Cluebooks available for both Bard's Tale and Bard's Tale II.™

You get a new class of magic user – the Archmage. With 8 powerful spells like Heal All, Fanskar's Night Lance, and the awesome Mangar's Mallot.

There are over 100 monsters, like this Kner Drone. Many animated. All dangerous.

25 scrolling dungeon levels. All in color. All 3-D. Including 7 different Snares of Death, a new kind of real-time puzzle.

The Bard's Tale II

The Destiny Knight

from

ELECTRONIC ARTS™

Optional use of characters from Ultima III™ or Bard's Tale™. Apple version uses Wizardry™ characters, too!

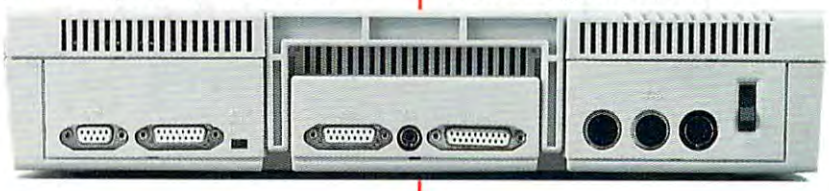
HOW TO GET IT: Visit your retailer, or call 800-245-4525 (in CA call 800-562-1112) for VISA or Mastercard orders. **To buy by mail,** send a check, money order, or VISA or Mastercard information to Electronic Arts, P.O. Box 7530, San Mateo, CA 94403. The price is $39.95 for the Commodore 64 version, and $49.95 for the Apple version. Add $5 for shipping and handling ($7 Canadian). Allow 1-4 weeks for delivery. The Bard's Tale II and Electronic Arts are registered trademarks of Electronic Arts. Ultima is a registered trademark of Richard Garriott. Commodore is a trademark of Commodore Electronics Ltd. Wizardry is a trademark of Sir-Tech Software, Inc. For a copy of our complete catalog, send 50¢ and a stamped, self-addressed envelope to Electronic Arts Catalog, 1820 Gateway Drive, San Mateo, CA 94404.

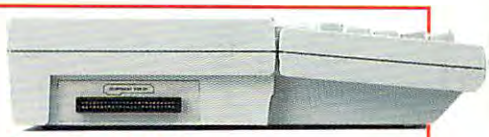
The standard equipment on a Laser 128 is optional on most computers

Take a look at the Laser 128 Apple®-compatible computer, and beginning with its built-in disk drive and 128K RAM of memory, you start to feel that this machine was designed with you in mind. The Laser 128 is ready-to-run – plug it in, insert your program and go. Everything is built-in for you.

Choose from the largest software library in the world. The Laser 128 runs Apple IIe and IIc software, which makes it nice for you and your family, because children can use the same programs at home that they learn on in school. The Laser 128 is a computer for the whole family.

The Laser 128 has built-in interfaces for all your peripherals, including serial and parallel printer interfaces, modem and mouse interfaces, a game port and you can hookup a color or monochrome monitor, or even a TV to it. Only the Laser has this much built-in as standard equipment.





It's expandable, too. The Laser 128 includes an expansion slot to let the computer grow with your increasing demands.

The Laser 128 is priced to keep money in your pocket. The Laser is about half the price of an Apple, and is even less than a comparably equipped Commodore unit. You get a lot of computer for a little money.

Look for the attractive Laser packaging at a store near you. For more information on the Laser 128 and the name of your nearest dealer, contact Video Technology Computers, Inc., 400 Anthony Trail, Northbrook, IL 60062, or call (312) 272-6760.

LASER 128®

Manufactured by
VIDEO TECHNOLOGY COMPUTERS, INC.

MAKING COMPUTERS AFFORDABLE