

November-December 1989 Edition

# RE RUN

## RUN Programs on Disk

For the C-64 and C-128



Plus: Extra Bonus Programs!

[www.Commodore.ca](http://www.Commodore.ca)

May Not Reprint Without Permission



# Introduction

*November–December '89 ReRUN*

I'M BACK, SPORTS FANS! After a six-month stint as Technical Editor of *RUN's* sister publication, *AmigaWorld*, I've returned to my roots as Technical Manager here at *RUN*. Thanks to the editions of *ReRUN* I continued to compile during my half-year hiatus, I feel as though I'd never really left.

I'm returning to the *RUN* staff at an exciting time in the life of the magazine. As many of you know, we recently acquired *Commodore Magazine*—a move guaranteed to solidify *RUN* as the prime source of support for Commodore 8-bit computers. Consolidating the two magazines also assures that *ReRUN* remains the best 8-bit Commodore software disk you'll find anywhere.

For an example of this first-rate software, look no further than *Friendly File Copier*, otherwise known as *RUN Copy*, the first program on this November–December edition of *ReRUN*. Bearing more resemblance to what you'd expect in a commercial disk-copying utility rather than a magazine type-in, *RUN Copy* lets anyone with access to two disk drives make backups of files directly between the drives.

Those interested in programming on their trusty C-64s will appreciate our *Enhanced Basic*, the next program on the disk. By adding commands such as *CASE*, *ELSE*, *WEND* and *WHILE*, the number of *Basic 2.0* commands is broadened significantly by this extended *Basic*.

Friday, the 13th of October, 1989, will long be remembered by investors as the second biggest day of losses in the stock market's history. Dire predictions by some investment analysts warn of more big-loss days to come. This high-finance excitement invades your C-128 with *Wall Street 128*, a 40-Column mode program that launches you with \$5000 on the road to financial success or failure. Remember the key to success: Buy low; sell high.

Then, turn your C-64's video monitor into an aquarium with *Deep C-64 World*, or *Aqua 64*, the next program on this disk. Written by talented programmer Ken Huebner, it truly turns your computer into a conversation piece when company arrives.



One of the most interesting and useful programs we've published in recent memory is Directory Speed Loader (DSL), entitled Get a Load of This! in December's issue. A fast disk loader, it takes advantage of both the 1571 and 1581 disk drives' high-speed file-loading capabilities in loading files up to 12 times faster than normal in 64 mode. Note that DSL must be copied to a work disk as the first file on the disk, and that either Make 1571 Usr or Make 1581 Usr must be run to write a machine language user file to that work disk. You can then save up to 20 fast-loading files to that same disk. While all this copying may take a few minutes, it's well worth the time spent.

Graphmaker, for either C-64 or C-128 mode, creates three-dimensional bar graphs for presentations, business and school reports. You can print the graphs and save them to disk for later recall. Graphmaker should meet just about everyone's graph-making needs.

Just in time for the holidays, Season's Greetings provides Beethoven's Ode to Joy, played with a variety of computerized instruments at various tempos, while a lovely winter scene slowly fills with snow. Complementing Season's Greetings is the first of our bonus programs, The Caroling Commodore, which allows your C-128 to play popular Christmas carols.

Tony Brantner, who has written many of *RUN*'s favorite entertainment programs, wrote the next bonus program, Wraparound, for the C-64. Plug in your joystick and try to arrange the squares in as few moves as possible.

Finally, Family Tree, a genealogy program for the C-64, is the last bonus program on the disk. Written by retired physician Hugh McMenamin, this program allows you to use your computer to categorize your entire family on disk. Just be sure to have a work disk handy when you run it, since the 336-block relative file that it creates for each family eats up more than half of a 1541-formatted disk!

Once again, it's time to wrap things up. Enjoy your holidays, and I'll be back early in 1990!



Technical Manager  
RUN Magazine

# Directory

PAGE	DOCUMENTATION	DISK FILENAME	FILE TYPE
		*MENU 128 _____	BASIC
1	FRIENDLY FILE COPIER	MENU 64 _____	BASIC
		RUNCOPY _____	ML
		RUNCOPY.BAS _____	BASIC
4	BUILD A BETTER BASIC	BOOT ENBASIC _____	BASIC
		ENHANCED BASIC _____	ML
		ENHANCED BAS.HEX _____	BASIC
9	WALL STREET 128	*WALL STREEET 128 _____	BASIC
11	DEEP C-64 WORLD	AQUA 64 _____	BASIC
11	GET A LOAD OF THIS!	DSL _____	BASIC
		MAKE 1571 USR _____	BASIC
		MAKE 1581 USR _____	BASIC
13	GRAPHMAKER	GRAPH MAKER _____	BASIC
20	SEASON'S GREETINGS	*SEASONS GREETING _____	BASIC
21	£ THE CAROLING COMMODORE	*XMAS CAROLS 128 _____	BASIC
22	£ WRAPAROUND	WRAPAROUND _____	BASIC
23	£ THE FAMILY TREE	FAMILY TREE _____	BASIC

\* — C-128 mode only

£ — Bonus program

Before you run a program, carefully read the documentation that pertains to it.



# How To Load

## LOADING FROM MENU

To get started, C-64 users should type LOAD "MENU 64",8 and press the return key. When you get the Ready prompt, the menu is loaded and you should type RUN to see a list of the programs on your disk. C-128 users need only press the shift and run-stop keys. When all the programs are displayed on the screen, you can run the one you select by pressing a single key.

## LOADING FROM KEYBOARD

If you do not wish to use the menu program, follow these instructions.

**C-64:** To load a C-64 program written in Basic, type: LOAD "DISK FILENAME",8 and then press the return key. The drive will whirl while the screen prints LOADING and then READY, with a flashing cursor beneath. Type RUN and press the return key. The program will then start running. To load a C-64 program written in machine language (ML), type: LOAD "DISK FILENAME",8,1

**C-128:** All C-64 programs can be run on the C-128 as long as your computer is in C-64 mode. All C-128 programs are clearly labeled on the directory page. Your C-128 *must* be in C-128 mode to run these programs. To load a C-128 mode program, press the F2 key, type the disk filename and then press the return key. When the program has loaded, type RUN.

## MAKING COPIES OF ReRUN FILES

Many programs on your ReRUN disk have routines that require a separate disk onto which the program writes or saves subfiles. To use these programs, you must first make a copy of the original program onto another disk that has enough free space on it to hold these newly written subfiles.

It's simple to make a copy of a Basic program. Just load it into your computer as outlined above, and then save the program back onto a separate disk that has plenty of free space for extra files.

Copying an ML program is not so simple. You cannot simply load and save an ML program; you'll need to use a disk-backup utility program, such as the one on your Commodore Test Demo disk.



# Friendly File Copier

By Ed Parry

RUN COPY IS A versatile and easy-to-use C-64 multifile/multidrive file copier. It's written completely in machine language for speed, can copy up to 200 files at a time and works with any two C-64-compatible disk drives.

## PRELIMINARIES

To use RUN Copy, reset your computer (to make sure memory is clear), then load and run it like any Basic program. It loads quickly and will soon display its self-explanatory main screen. The screen lists various fields, including filename, file type, file size, files chosen to copy, files actually copied, and so forth, and offers a choice of three program options, which are explained below. The screen also contains a window that displays messages as you use the program.

While using RUN Copy, you can reset the program at any input prompt by pressing the left-arrow key. To exit RUN Copy, press the run-stop key when no option is pending.

## THE COPY FILES OPTION

Copy Files, the main RUN Copy option, is invoked by pressing the C key. When the program prompts you for the source and destination drive numbers, specify each by using the cursor-right key to increase the default number, or the cursor-down key to decrease it, and then pressing the return key to enter your choice.

Automatic drive limits range from device number 6 (if you're using a hard drive or RAM disk program) through 30. After you have pressed the return key, RUN Copy automatically checks for the presence of the drive, and, if it doesn't find it there, displays an error message.

Note that when it asks for your input, RUN Copy always offers a logical default. To accept the default, just press the return key.

After you've chosen drive numbers, the program will prompt you for a copy pattern. Any standard directory pattern is acceptable,



with the default being \* (all files). Examples of acceptable copy patterns are shown in Table 1.

Next, RUN Copy will read the source directory, display the first filename that fits in the pattern you chose and ask if you want it copied. Press Y, for yes, to place the filename in the program's internal list of files to be copied, or N, for no, to exclude the file from the list. Pressing A will abort the copy option and reset the program.

Once you've specified the files, copying begins. The program uses the standard disk read/write routines, so you can copy files easily from any standard Commodore disk drive to any other. Slower than some commercial copiers with customized DOS routines, RUN Copy compensates through its maximum disk-drive compatibility.

Utilities (such as JiffyDOS) that speed up read/write operations work well with RUN Copy. Also, when you're using a C-128 and 1571, you can speed things up by putting the drive in 1571 mode via the RUN Copy wedge. Type `u0>m1` and then press return to activate this mode.

During copying, the R/W field indicators flash back and forth to indicate which phase, read or write, the program is in. Filenames

**Table 1. Examples of acceptable copy patterns.**

<b>Command</b>	<b>Pattern</b>
*	All files in the directory
*=P	PRG files only
*=S	SEQ files only
A*	Files whose names start with A

**Table 2. Examples of acceptable disk commands.**

<b>Command</b>	<b>Function</b>
>i0	Initializes a disk (loads the BAM)
>v0	Validates (cleans up) a disk
>n0:target disk,RC	Formats a disk as the target, with an ID of RC
>n0:erase disk	Erases a formatted disk
>u0>m0	Puts a 1571 drive in 1541 mode
>u0>m1	Puts a 1571 drive in 1571 mode
>s0:filename	Scratches a file

are displayed and the files copied in the order that you chose them.

The \* field shows the file number currently being copied, the To Copy field shows the total number of files yet to be done, and the Copied field shows the number of files already completed. A "copy complete" message appears when a copy is finished.

You can abort a copy operation at any time by holding down the run-stop key. It may take several seconds for the abort to take, so keep pressing the key until you see the "copy aborted" message.

### **THE DISK COMMAND OPTION**

This option lets you send any standard C-64 disk command, so you can scratch files, format new disks, validate disks, and so forth, from RUN Copy. Examples of acceptable disk commands appear in Table 2. For further help, see the section on disk commands in your drive manual.

Press the @ key to activate the Disk Command option. Then use the cursor and return keys to specify the number of the drive that should receive the command.

Here again, acceptable drive numbers range from 6 to 30, inclusive. RUN Copy will verify the presence of the drive and then prompt you for the command to be sent. Type the command and press return. After execution, the program will display the disk status as returned by the drive.

### **THE DIRECTORY COMMANDS OPTION**

The third RUN Copy option lets you display disk directories. This is similar to using the Disk Command wedge. Specify a drive device number with the cursor and return keys. Then, at the \$0> prompt, enter any valid directory pattern (see Table 1) or press return to accept the \* default.

In addition to the normal directory information, RUN Copy shows how many files are in the list. At the "pause" message, press any key to return to the main screen.

RUN Copy might not be the fastest file copier around, but it loads quickly, offers maximum drive compatibility and is easy to use. It will make a valuable addition to your utility library.



# Build a Better Basic

By Scott Weisgarber

THE C-64 IS A great machine, but the built-in Basic 2.0 lacks some useful commands that exist in more recent versions of the language. Enhanced Basic (hereafter called EB) updates Basic 2.0 by adding If/Then/Else and Case commands, labeled Gotos and Gosubs, and While and Repeat loops.

## IF/THEN/ELSE

A Basic 2.0 If statement can be followed by one and only one Then condition; multiple statements and Else conditions aren't possible. EB remedies this lack with an If/Then/Else sequence. The format for the sequence is as follows:

```
10 IF A=1 THEN
20 PRINT:PRINT "YOU SELECTED 1"
30 PRINT "GOOD CHOICE!"
40 ELSE
50 PRINT "YOU DIDN'T SELECT 1"
60 PRINT "PLEASE SELECT ANOTHER"
70 ENDIF
```

If you want to do this with Basic 2.0, you must say:

```
10 IF A=1 THEN PRINT:GOTO 50
20 PRINT "YOU DIDN'T SELECT 1"
30 PRINT "PLEASE SELECT ANOTHER"
40 GOTO 70
50 PRINT "YOU SELECTED 1"
60 PRINT "GOOD CHOICE!"
70 (Continue with the program)
```

Clearly, the EB code is easier to understand. You can have as many statements as you want following the Then and Else statements. You can also have a Then without an Else. When you're finished with the sequence, close it with an Endif.



There may be times when you prefer an old-fashioned If—for example, when you want to execute only one statement and you don't want to worry about an Endif. That's no problem; just put the statement to be executed right after the Then statement in the line, as shown below:

```
IF A=1 THEN PRINT "YOU SELECTED 1"
```

(Continue with the program)

The regular Basic 2.0 If appears to EB as a completely different command than the special EB If. The rule is: When there's something after the Then statement in the same line, it's a regular If; when there's nothing after it, it's treated as an EB If.

### CASES OF . . .

The next set of commands is the Cases Of/Case/EndCase/Otherwise/EndCases group. It provides a shorter way to use the If/Then structure when there are many possible values of the checking variable. For example, let's say you were checking user input at a menu. In Basic 2.0, the code might look like this:

```
10 INPUT A
20 IF A=1 OR A=2 THEN GOSUB 100:GOTO 60
30 IF A=3 OR A=4 THEN GOSUB 200:GOTO 60
40 IF A=5 THEN GOSUB 300:GOTO 60
50 PRINT "I DON'T UNDERSTAND YOUR CHOICE"
60 B=B+1
70 GOTO 10
```

With EB, you can do the same thing using the Case statement:

```
10 INPUT A
20 CASES OF A
30 CASE 1,2:GOSUB 100:ENDCASE
40 CASE 3,4:GOSUB 200:ENDCASE
50 CASE 5:GOSUB 300:ENDCASE
60 OTHERWISE
70 PRINT "I DON'T UNDERSTAND YOUR CHOICE"
80 ENDCASES:GOTO 10
```

Once again, the EB version is easier to understand. You start with a simple Cases Of, followed by the variable you wish to check. Then, after finishing with each Case, you close it with an EndCase. If none of the cases holds true, the Otherwise, if present, is executed. When

you're completely finished checking the variable, close the sequence with an EndCases.

The variable may be of any type. If it's named incorrectly, you won't get an error message until it's activated by the first Case statement. For example, in

```
10 CASES OF 1A
20 CASE 1:PRINT:ENDCASE
```

variable 1A is invalid, but you won't get an error message until line 20. Any time you try to compare two different types of data, such as a string and a numeric value, you'll also get the error message at the Case statement.

## LOOPING

Until now you had to use a series of If/Thens and Gotos to build a loop, as shown below:

```
10 PRINT "ALMOST DONE"
20 B=B+1
30 If B<>5 THEN 10
```

EB makes looping easier with the While/Wend and Repeat/Until commands. Here's how they look:

```
WHILE B<>5
  PRINT "ALMOST DONE"
  B=B+1
WEND
```

or

```
REPEAT
  PRINT "ALMOST DONE"
  B=B+1
UNTIL B=5
```

While/Wend and Repeat/Until differ in where the condition is tested. In While/Wend, it's tested at the beginning of the loop; if the condition is false, execution skips to Wend. If the condition is false the first time through the loop, the statements inside the loop will never be executed. Repeat/Until is the opposite. The condition is tested at the end of the loop, so the loop is always executed at least once. Notice that the operators for the two sequences are opposite: While A<>5, Until A=5. Keep this in mind.



## GOTO AND GOSUB

The most obscure commands in Basic have got to be Goto and Gosub, because all they contain are meaningless numbers. For instance, in

```
GOTO 100
GOSUB 3000
```

what's in line 100? What does line 3000 do? What happens if you change the line numbers, making line 100 become line 130? This lack of clarity can lead to problems—problems that EB solves by letting you use meaningful labels in plain English, enclosed in quotes as targets:

```
"MAIN LOOP"
PRINT "*****"
INPUT A
IF A=1 THEN GOTO "MAIN LOOP"
```

or

```
GOSUB "DO TWICE"
:
:
:
"DO TWICE":F=F+2:RETURN
```

Just remember to put the targets in quotes. Also, notice how the If in the Goto example uses the regular If format.

Of course, what good are labels if you can't use them with the On command? Well, you can, as this example illustrates:

```
INPUT A
ON A GOSUB "ONE","TWO","THREE","FOUR"
```

You can do the same with Goto. What if you want to use regular line numbers as targets? Once again, I anticipated your needs, giving you the option of using either. Both of the formats below are acceptable in EB:

```
GOTO 100
GOSUB "HELLO"
ON A GOSUB 300,500
ON A GOTO "UNO","DOS"
```

## INDENTATION

In addition to fancy commands, most Basic interpreters use indentation to make programming easier. Following that lead, I've provided an automatic indenter. It not only works by itself, but it helps you find errors.

Let me explain what I mean by indenting. In the above examples, you may have noticed that statements following special key words are indented. These key words, which I call "indent" words, include If, Cases Of, Case, While and Repeat. Whenever EB hits one of them, it indents all words that follow until an "end" word appears. With the end words—EndCase, EndCases, EndIf, Until and Wend—the indentation ceases. For instance,

```
WHILE A<1
  (Indented lines)
WEND
```

## ERROR-AVOIDING RULES

It's important to adhere to the following rules, or your program won't work properly:

While, Wend, Repeat, Until and Cases Of must be on lines by themselves.

Never put a comma inside the testing part of a Case statement, or you'll get an error.

Make sure that the variable name in a Cases Of statement is no more than nine characters long. If it's longer, it will cause a syntax error.

You can have only one Cases Of statement active at once. Do not try to nest them.

When using labels with the On command, put nothing after the last label. For example, don't write

```
ON A GOSUB "ONE","TWO":PRINT "THIS WILL CAUSE AN ERROR"
```

When using an On command, don't switch between EB and Basic 2.0 forms. For instance,

```
ON GOTO 100, "INIT","START",200
```

would cause an error.

Every indent word must have a matching end word. This is where listing the program comes in handy. If the lister comes across an end word without a previous indent word, it highlights the end



word and stops the listing. For example, listing

```
WHILE A<1  
  A=A+1  
WEND  
ENDIF  
PRINT "CONTINUE"
```

would produce the following:

```
WHILE A<1  
  A=A+1  
WEND  
ENDIF
```

An easy way to see if you have too many indent words is to list the last line of the program. If it's indented, you have an unpaired indent word.

You can have no more than ten While/Wend and ten Repeat/Until loops going at the same time. That gives you the possibility of 20 nested loops—more than you should ever need. But, just in case, I have provided a few self-explanatory error messages.

#### PROGRAM NOTE

Enhanced Basic occupies memory from 49152 to 51148 and uses 51149–51221 for miscellaneous data tables. Keep other programs away from both areas.

---

RUN it right: C-128 (in 40-Column mode)

## Wall Street 128

*By Michael Ballard*

HAVE YOU EVER WISHED you could invest in an active stock market without worrying about losing money? Well, now's your chance. In the Wall Street game, you receive \$5000 to invest in any of six stock categories, and the stocks make or lose money for you, depending on the wisdom of your investments.



Wall Street is set in a future time, when many of the stock market regulations have been removed. Because of this, the market is extremely volatile; a stock worth \$100 one day may be worth only \$30 the next, or, in reverse, a stock may suddenly jump from next to nothing to \$30. All this means that you must keep a close eye on the screen activity.

After running Wall Street, you'll see the trading board itself, with your initial \$5000 shown in the upper right. The upper left shows how many shares of each type of stock you hold, along with their current values. The net worth figure, back in the upper right, reflects your cash on hand plus the value of the stocks you own.

The large area at the bottom of the screen shows the current market value of each type of stock, both graphically and numerically. As soon as the screen is drawn, these figures will start to change randomly. The most the value of any stock will change per unit time is \$10, with two exceptions: Any stock under \$10 in value may suddenly jump to \$30, and any stock over \$100 may suddenly drop to one-third its value.

So why not buy all cheap stocks and sell the expensive ones? Because any stock under \$10 may also drop to zero, at which point any shares of it you own are removed as worthless. Also, any stock over \$100 may start paying dividends, figured on the number of shares owned.

One last obstacle to profit-making is, of course, taxes. Once a year Uncle Sam subtracts your previous net worth from your current net worth and levies a 25 percent tax on the difference, if positive. He doesn't tax actual profit, because, with the market deregulated, people would have too many chances to cheat.

Note that losses aren't tax deductible, and if you lose for more than three years in a row, the government will close down your investment activities. Also, if you lose in the market, yet earn dividends, you must pay 25 percent on the dividends. This may leave you with negative cash on hand, but the only consequence is that you can't buy more stock without liquidating some of what you have.

To start buying and selling and making your first million, press the space bar to go to the Activity menu. The options are Buy Stock, Sell Stock, Instructions, Quit and Return to Market. If you opt to buy or sell, you can choose from Industrials, Gold, Silver, Cattle, Grain and Computers.

All price changes are on hold while you're in Activity mode, and you may enter the mode whenever you wish. If it takes a second

for the space bar to respond, it's just that the computer is filing the current price changes first.

The game ends when your net worth falls below \$100. Have fun, and remember: Buy low and sell high!

---

RUN it right: C-64

## Deep C-64 World

By Ken Huebner

HOW MANY TIMES have you found your favorite fish floating in your aquarium—stiff as a board? Well, have I got a treat for you! This program will turn your monitor into a colorful tropical aquarium, filled with animated seahorses, angelfish, turtles and myriad other aquatic delights. You'll never have to look at a lifeless fish again; electronic creatures cannot be overfed, underfed, over- or underheated.

I created the Aquarium's multitude of creatures from a few sprite images, varying their color, width, height and sprite/screen priority. The animated plants are built from Commodore graphics characters.

Warning: Before running Aquarium, make sure your cat is in the other room, and the stopper is in your monitor!

---

RUN it right: C-64 or C-128 in 64 mode; 1571 or 1581 disk drive

## Get a Load of This!

By Michael Miller

THE 1571 AND 1581 DISK DRIVES brought increased speed and storage capacity to Commodore computing—but only when using a C-128 in 128 mode. With a C-64 (or C-128 in 64 mode), the drives

default to their usual long loading times. Directory Speed Loader (DSL), however, loads program files up to ten times faster than normal. With DSL, a 130-block file that normally takes 90 seconds to load will be in memory in less than ten seconds.

DSL itself loads in only a few seconds, then automatically displays a directory of the program files on the same disk. To load one of the files, just place the cursor over its filename and press the return key. DSL checks to see if your disk drive is a 1571 or 1581 and, if so, starts loading at high speed. If you're not using a 1571 or 1581, it will load at the normal speed.

When DSL is used with a 1571, a `PRINT#15,"U0>M1"` command in the program automatically switches the drive to 1571 mode. When used with either a 1571 or 1581, DSL can load files of up to 250 blocks (considerably more than the usual 202-block maximum), it allows load addresses of \$0400 and up, it maintains the screen display during a load, and it even checks for disk errors!

Note that, if the file you select to load loads any others in turn, the others will enter memory at normal speed. Also, DSL works great with memory snapshot-type programs.

## HOW IT WORKS

DSL uses a combination of Basic and machine language. The C-64 machine code is poked into the cassette buffer, and the 1571/1581 machine code is loaded into the drive via a user file, using the special 1571/1581 Utility Loader command.

DSL does not use Burst-mode serial communication, because Burst mode isn't possible on the standard C-64. However, the serial routines between the C-64 and 1571/1581 using DSL are as fast as Burst mode.

The actual speed increase using DSL with a 1571 depends on the sector interleaving that was used when the file was saved. The interleaving determines how close together sectors are written, and with the 1571 it's normally in the range 6-10. With an interleave of 10, DSL will load a file about five times faster than normal; with an interleave of 4 (the fastest that DSL can use with the 1571), the file will load ten to 12 times faster than normal. The 1581 drive has a full track buffer and uses a sector interleave of 1, making the speed increase with DSL about tenfold.

## HOW TO USE IT

DSL is best installed as the first program on a newly formatted



disk. Load both DSL and Make USR File (2 or 3, depending on which drive you have), and save them to a work disk. Then run Make USR File to create a user file named either 1571 or 1581, depending on your drive. The user file will become the second file on your disk. Finally, save to the disk any files you want to be able to load at high speed. But don't save more than 20, because the DSL directory routine might scroll the excess filenames off the screen.

To use DSL, type LOAD ".\*";8: and return, then RUN and return. The directory of program files will appear on the screen, and DSL will await your desire.

---

RUN it right: C-64 or C-128, in 40- or 80-Column mode;  
printer optional

# Graphmaker

*By Robin Franzel*

YOU CAN USE GRAPHMAKER 64/128 in either 40- or 80-Column mode to create attractive and informative bar graphs. If your printer outputs Commodore graphics characters, your graphs will appear in "3-D;" if you use a daisywheel printer, or have trouble getting your printer to obey commands, your graphs will be made of standard printer characters.

Data for the bar graphs can be entered manually or through complex analysis of sequential database files. You can even edit the data that defines your graph, and graphs can be saved to and loaded from disk.

## GETTING STARTED

After you run Graphmaker, the bottom of the title screen indicates whether you're using a C-64 or a C-128 and whether you're in 40- or 80-Column mode—a handy feature for C-128 users who often switch between 64 and 128 modes. After the title screen, the main menu, shown in Figure 1, appears. When you're just beginning to use Graphmaker, you can press return for the default choice, Create



Graph; 7 for Printer Setup; or 8 to End. The other selections aren't available until after you've created a graph, so let's get started by pressing the return key.

## CREATING A GRAPH

Refer to Figure 2 and Figure 3 for clarification as we proceed. Figure 2 shows the initial screen for creating a graph; it requests information from you. You may find it helpful to go through the exercise of re-creating the sample graph in Figure 3.

At the **title** prompt in the Figure 2 screen, you can input both lower- and uppercase letters, and the title will appear that way in printouts. However, the letters will all be uppercase when your graph is displayed on the screen.

The **legend** is a line at the bottom of your graph that can be used for explanation, such as indicating your units of measurement. Again, both lower- and uppercase may be used.

The **number of ranges** tells Graphmaker how many bars to display in a particular graph. There are 12 bars in Figure 3.

The next prompt asks where Graphmaker will get the maximum values for each range—from keyboard input or from a disk file. If you intend to type them in, enter K; if the program should look for

**Figure 1. The main menu.**

1. Create Graph
  2. View Graph
  3. Edit Graph
  4. Save Graph
  5. Load Graph
  6. Print Graph
  7. Printer Setup
  8. End Program
- Enter Selection (1-8) ? 1

**Figure 2. Initial screen for creating a graph.**

Create Graph

Press ← for Menu

Enter Title:

Enter Legend:

Number of Ranges (1-12):

Keyboard or Disk Input? (k/d)

Label for Range 1:

.

.

.

them in a database file on disk, enter D. For now, press K and return.

Next, you must enter a **label** for each range. These labels will run along below the x-axis of the graph, truncated by the program, if necessary, to prevent overlapping. In general, the fewer bars in a graph, the more space that's available for labels. In a 40-column graph with six bars or an 80-column graph with 12 bars (see Printing below), the labels can be up to four characters long.

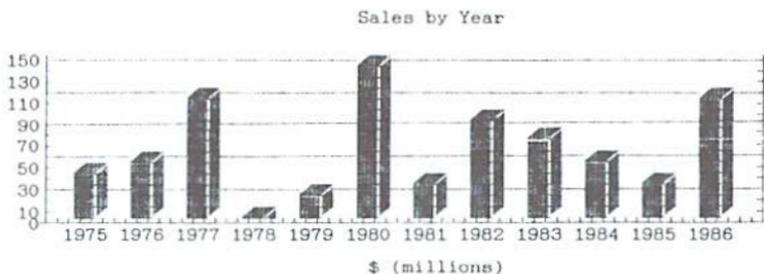
Finally, you'll be prompted for the maximum value in each range. For the graph in Figure 3, 12 numbers from 10000000 to 150000000 (no commas) must be entered. To make the graph look neater, I edited the numbers (see Edit Graph below) to run from 10 to 150, then indicated the conversion in the legend.

After you have answered all of the prompts, the main menu will reappear, with a new default choice of 2, for View Graph. Before I describe that option, however, let's see what happens if you press D at the Keyboard/Disk Input prompt.

### ANALYZING A DISK FILE

To learn how to analyze a database file, consider a database that contains statistical data concerning a fictitious company. The record

Figure 3. Sales by year.



**Figure 4. Fields in sample employee record.**

Field 1: Employee number  
Field 2: Employee age  
Field 3: Codes by type of work  
    1 Manual Labor  
    2 Secretarial  
    3 Supervisory  
    4 Analysts  
    5 Managerial  
    6 Executive  
Field 4: Salary of Employee

**Figure 5. Printout of beginning of sample data file.**

```
SAMPLE.DBF
 42
 4
EMPL NUM
EMPL AGE
TYPE WORK
SALARY
1
52
6
125000
2
45
6
100000
3
38
5
65000
```

**Figure 6. Printout of a portion of a sample database.**

NO.	AGE	TYPE	SALARY
1	52	6	125000
2	45	6	100000
3	38	5	65000
4	42	5	63500
5	34	3	32750
6	36	3	33500
7	28	3	30250
8	38	3	38200
9	42	5	55000
10	53	5	63500
11	47	2	23500
12	42	2	21500
13	22	2	18200
14	18	2	16500
15	18	2	16500

**Figure 7. The Disk Menu.**

```
Create Graph

1. User-Defined Ranges
   Tally Field (Alpha/Numeric)

2. User-Defined Ranges
   Tally Field (Numeric)

3. File-Defined Ranges
   Sum Field

4. Return to Main Menu

Enter Selection (1-4)
```

for each employee includes the fields that are shown in Figure 4.

Many Commodore database and spreadsheet programs produce sequential data files that can be read into a word processor. Before you attempt to use a data file with Graphmaker, load it into a word processor so you can look at it.

Figure 5 shows a printout of the beginning of a data file on our fictitious company created by RapidData Software's DataOrg database. That program places fields holding the filename, the number of records, the number of fields per record and the name of each data field at the beginning of the file, before the actual data. Graphmaker must be told to skip this housekeeping information. Figure 6 shows a DataOrg printout of the same database, including all the records.

Now that you know the layout of our database file, let's produce the following bar graphs:

1. Number of employees in each work category.
2. Distribution of employees by age group.
3. Salaries paid in each work category.

Graphmaker offers three types of database analysis in the Disk menu in Figure 7. With option 1, you can count occurrences based on comparisons of alpha/numeric data (we'll use this for our first graph). Option 2 lets you specify high and low values for your ranges using numeric analysis (the second graph). With option 3, you can sum a numeric field based on an alphabetic comparison of a second field (the third graph).

### ANALYSIS OPTION 1

To re-create a graph showing the number of employees in each work category, enter 6 for the number of ranges, since there are six types of work. Then press D at the Keyboard or Disk-Input prompt and 1 for Tally Field.

The number to enter next, for **fields to skip**, depends on your file organization. To analyze the entire contents of your data file, you'd enter 0. For our example, enter 7 to skip the seven housekeeping fields of the file (see Figure 5 again). Any time you've progressed this far and realize you don't know how many fields to skip, you can enter ← to return to the main menu.

To analyze the number of employees in each work category, enter 3 for **field number of data**, since Type of Work is the third field in each record.

The number of **fields per record** also depends on your file or-



ganization. Since each record in our example has four fields, enter 4 at this prompt. The number of ranges was specified earlier, but now you must indicate what each one represents by entering its Type of Work code: 1-6 (see Figure 4 again). You can use the graph editor later to change these codes to more meaningful labels.

Now Graphmaker will read your data file and create the graph; then it will display the main menu so you can view, edit, save or print the graph.

### ANALYSIS OPTION 2

Next, we'll do the analysis of employees by age. Enter 4 for the number of ranges and D at the Keyboard or Disk Input prompt, then 2, for Numeric Tally, when the Disk menu (Figure 7) appears. Again, you'll be asked for the number of fields to skip, field number of data and number of fields per record. This time, the field number of data will be 2, for the Age field.

When the program prompts for the **low/high value for the range**, enter the minimum and maximum age in each group. If you enter a high value that's smaller than the corresponding low value, Graphmaker will alert you and let you re-enter the values. The labels for the ranges will be generated automatically, based on your input.

### ANALYSIS OPTION 3

The final type of analysis we'll do sums up one field based on the contents of another. It produces a graph that shows how much money was paid in salaries to each type of worker. Enter 6 for the number of ranges, since there are six types of work for which salaries will be totaled. Then select 3 at the Disk menu (Figure 7) and enter the number of fields to skip.

With this analysis, the prompt for **field number of data** refers to the field you want to sum. Enter 4 for our example, since salary is the fourth field. **Field number for ranges** is again 3, Type of Work.

If you wanted to use an alphabetic field for selection of records to be included in summing a numeric field, you'd respond with A to the Alpha or Numeric prompt. For our example, we're using the numerically coded Type of Work field, so enter N.

Finally, as before, enter the number of fields per record and the ranges to be searched.

### VIEWING YOUR GRAPH

Once you've created your graph, the default at the main menu



becomes 2, for View Graph. Press return, and, if your computer is in 40-Column mode, the first six bars of the graph will appear. The bars to the right are still there, but there isn't any room to display them. As I mentioned before, they'll appear in the printout (see Printing, below). In 80-Column mode, your entire graph will fit on the screen.

To make changes to your graph, press any key to access the main menu, and then select 3, Edit Graph.

### **EDITING YOUR GRAPH**

In Edit mode, you can retype the title, legend, range labels and even the range values. When you're done, the main menu will reappear, offering you the option to proceed to another part of the program or to edit some more by pressing 3 again.

### **SAVING YOUR GRAPH**

Once you've created your graph and edited it as desired, you'll probably want to save it to disk. Select 4 from the main menu, then enter a filename that's no more than 16 characters long. If you enter one that's too long, Graphmaker will automatically truncate it. Any time the program encounters a problem with the disk drive, it will display an error message on the screen and return you to the main menu to try again.

### **LOADING YOUR GRAPH**

To load a previously saved graph into Graphmaker, select 5 at the main menu and enter the filename. Here again, the program will report any disk errors.

### **PRINTING YOUR GRAPH**

Printing a graph is as easy as pressing 6 at the main menu. If the printout doesn't look right, see the Printer Setup section below to correct it.

The printout of a graph that contains six or fewer bars will be one-half page wide; a graph that contains more than six bars will extend across a full page. You can expand a graph with six or fewer bars to make more room for range labels by adding enough "invisible" bars (no label and a range value of 0) to bring the total to seven.

### **PRINTER SETUP**

Graphmaker makes use of Commodore graphics to print out the



same three-dimensional graph you see on the screen. However, you can also produce a printout with normal daisywheel characters. Just select 7 at the main menu and respond "no" at the CBM Dot Matrix Printer prompt.

You'll be prompted to enter the secondary addresses that Graphmaker should use to open your text and graphics printer files. Normally, you'd select 7 for the text secondary address and 0 for graphics, but with a daisywheel printer, you must select 7 for both. If these secondary addresses don't work with your printer, consult your printer or interface manual for other possibilities.

Finally, Graphmaker will ask for the number of printer control codes to send to the printer when the printer file is opened. If you don't want to send any codes, overwrite the default with 0. The default settings will tell an Epson-compatible printer to print approximately 10 lines per inch (a  $\frac{7}{72}$ -inch linefeed), so the lines barely touch. If these control characters aren't correct for you, refer to your printer manual to determine how to adjust the linefeeds.

## END PROGRAM

Selecting 8 at the main menu terminates execution of Graphmaker. If you have a graph in memory, the program will ask for verification that you want to exit—in case you forgot to save the graph or need to print it out. Then type N to return to the main menu, or Y to leave the program.

Even if you type Y, it's not too late to save or print your graph. Just enter RUN 740, and the main menu will return, with your last graph still in memory.

---

RUN it right: C-128 (in 40-Column mode)

# Season's Greetings

*By Malcolm Gardner*

CIRCLE AROUND YOUR C-128 and enjoy the magical holiday scene created by Season's Greetings, a program for the C-128 in 40-Column mode. Much of Santa's know-how about C-128 sound



and graphics went into this program. The scene features a wreath of moving multicolored lights that's highlighted by a big red bow. The wreath hangs in a window that looks out on a night sky filled with stars and a crescent moon. Snow sifts down, piling higher and higher on the outside window sill for about an hour. The background music is not a run-of-the-mill holiday ditty, but the Ode to Joy, from Beethoven's ninth symphony, played by various "instruments" at different tempos.

If you're in 80-Column mode when you run Season's Greetings, you'll be prompted to switch your monitor to 40 columns, then to return to 80 columns when you end the display (just switch the monitor button again). From 40-Column mode, you'll be prompted to wait 20 seconds while the program loads some of Santa's graphics data.

When you're ready for bed or to depart to an alternative program, just push the run-stop key while the light is moving around the upper portion of the wreath. With warm season's greetings to everyone!

---

RUN it right: C-128 (40- or 80-Column mode)

## The Caroling Commodore

*By Rami and Elias Tabello*

NOW CHRISTMAS BELLS can come from your C-128! That's right, not a horse's harness or a church steeple or an orchestra, but your good ol' Commodore computer. How does it get so musical? By running Christmas Carols 128, a Basic 7.0 program that plays eight popular carols, most of them in two or three voices.

Load Christmas Carols 128 from Menu 128, then run it to accompany family sing-alongs, entertain friends at a party or even amuse Santa Claus as he works. All you have to do is select the carol you want at the prompt. To exit the program, press Q at the prompt. Of course, if you set up Christmas Carols 128 for Santa to use, be sure to leave a note with instructions next to the milk and cookies!



# Wraparound

By Tony Brantner

IF YOU LIKE to get wrapped up in a game once in a while, try Wraparound, an engrossing puzzle for the C-64. In Wraparound, two multicolored patterns are generated by the computer. Both contain the same blocks, but in different arrangements. Your job, using a joystick in port 2, is to shift rows and columns within the pattern on the left until both patterns match—and to do it in the fewest moves possible!

As the game begins, you select the size of the patterns you want to work with; Wraparound can generate from three to six rows and the same number of columns. You also select the number of colors, from two to six, to appear in the patterns. (You may find, when using five or six colors, that some of them are quite difficult to distinguish. This can be corrected by tweaking the color adjustment on your monitor.) Both selections are made with the joystick. When you are prompted, push the joystick forward to decrease the flashing number or pull back to increase it; then press the firebutton to enter your choice.

As soon as it knows your preferences, the program displays the master pattern on the right and the scrambled pattern on the left, the latter with a square white cursor in its top-left corner. Use the joystick to move the cursor to a square in the row or column you want to shift and press the firebutton to enter Shift mode. The cursor will turn into a four-directional arrow. Then push the joystick in the direction you want the row or column to move, and all the squares will shift one space, the last wrapping around to the first. After the shift, press the firebutton to exit Shift mode and move the cursor freely again. As you play, the number of moves you've used is continually shown at the top of the screen.

The game ends when you either duplicate the master pattern successfully or press the F7 key. When the program asks if you'd like to play again, push the joystick forward for yes or pull it back for no, then press the firebutton to register your choice.



# The Family Tree

By Hugh McMenamin

IN GENEALOGICAL SEARCHES, handling the data on individuals and their relationships in an efficient manner is a real problem. The Family Tree addresses this by providing a logical and easy-to-use storage and retrieval system—a genealogical database. Personal data on individuals may be entered and recalled as desired. Output is to the screen or to the printer, with the succession of generations portrayed through a series of indents.

The Family Tree keeps data in two files: Branch, a sequential file that holds names and their relationships, and Leaves, a relative file that contains information on individuals. Leaves occupies 336 blocks, enough space for 400 individuals, so after you've loaded the program from the ReRUN disk (using Menu 64), you should immediately copy it to a freshly formatted work disk and use that for your genealogical work. You also need a fresh work disk for each family.

When you run the program, it asks for a family name, then spends about three minutes writing the Leaves file to disk. When that's done, the work screen appears, with the family name in place.

The program is controlled with the space bar, the function keys, the cursor-up and -down keys, and the back arrow (←) key.

## ENTERING DATA

To start entering data, press the space bar. Then, following the quotation marks in the entry area, type the name of the father of the first (most recent) generation. Next, press return, and the name moves down from the entry area to start a list lower on the screen. If the father of the first generation is unknown, just press return and "Unknown" is entered automatically. The quotes make it possible to use commas and colons in your entries.

After the father, enter the mother and the children, whom the program refers to as siblings. Each time you enter a name and press the return key, you add the name to the list below, preceded by f, m or s, standing for father, mother or sibling. The maximum number

of entries in each generation is 14. To correct an entry, type the new version over the old. To cancel the change, type a back arrow at the beginning or end of the entry line and press return.

After creating the list, you can select any name in it by moving the indicator arrow up and down with the cursor controls. If you then press the space bar, the name you selected appears in the entry area, ready for further action.

## THE FUNCTION KEYS

With the indicator arrow pointing at the father or mother, press the F1 key to enter the previous generation (paternal or maternal grandparents). Then enter the names of individuals of that generation as before. You can progress to prior generations only via the father or mother.

The Family Tree provides only for consanguineous relationships. There's no provision for step-relatives or in-laws in the genealogical chain. However, they may be noted in the remarks (see F4 below).

To return to the next younger generation, press F3.

You can delete a sibling's name by pressing the F5 key. Before deleting, the program checks to make sure that's what you want to do. It won't let you delete father and mother entries with F5.

To quit the program, press the F7 key. *Do not use the run-stop/restore key combination to disable The Family Tree*, for you might then lose updated names, which are automatically stored in the sequential Branch file before the program is terminated.

The F2 key prints the lineage of entered names to your choice of the screen or the printer. You can also choose to print out only those who are parents, or to include, as well as parents, those who have had no offspring. In any case, both fathers' and mothers' names are preceded by different distinguishing symbols—a solid block and an open block, respectively.

At the top of the screen and on every tenth line of the printed page, you'll see a line of numbers that indicate generation levels. The number of the generation is in the column directly above the first letters of the names belonging to that generation. If you press F2 at other than the beginning level, the generation indicator changes appropriately. If you try to enter more generations than the screen or page width will allow (about 22), you'll get a Not Enough Room message.

F4 calls the file of the individual you've selected with the indicator arrow. If you haven't previously entered data for that person, you're

asked if you want to do so now. If so, press Y, then follow the prompts to enter the data, including birth date and place, date and place of death (if appropriate and known), occupation (limited to 15 spaces), date of marriage and premarital name of spouse (if appropriate). You may also enter two 40-character lines as remarks.

After entering a data item, you're asked if it's correct. When all the entries are finished, the program stores the data in the Leaves file on the disk. You may enter data for 12 individuals consecutively, after which the program automatically stores the updated Branch file. To continue using the program, type RUN and press the return key, which starts the program over.

If data is present in the individual file when you hit F4, it's recalled to the screen. Then you have the following five options: Change, Delete, Print, Store and Return.

**Change** displays the data in the selected file with line indicators 0-8 preceding the data lines. Unknown dates are shown as 99999999, while nonapplicable dates are shown as 00000000. The other information is quite recognizable. To correct a line, call it and follow the prompts to alter specific data.

**Delete** removes an entire entry from the Leaves file (useful when a sibling entry has been deleted).

**Print** sends file information to the printer, including generation level and relationship to the previous individual.

**Store** places the changed data in Leaves. This must be done before exiting F4 mode, or the changes will be lost.

**Return** takes you back to the entry screen.

## HINTS

If, by some chance, the program stops before the final save of the Branch file, type, in Direct mode, GOTO I40 <return>. This brings you back to the main screen without losing data. Then, using F7, exit immediately to preserve all your data.

Be sure to start entering data with the youngest father in the family chain. In that way, his children will be designated as generation 1, the father as generation 2, the grandparents as generation 3, and so forth.

If you want to start afresh, scratch the Branch and Leaves files. And remember, a new family requires a separate disk.

*If you have questions concerning this program, please write the author, Hugh McMenamin, 1014 Greenfield, Peoria, IL 61614. ■*



# **RE** **==** **RUN**

EDITOR-IN-CHIEF

**DENNIS BRISSON**

TECHNICAL MANAGER

**TIM WALSH**

MANAGING EDITOR

**SWAIN PRATT**

SENIOR EDITOR

**BETH S. JALA**

ASSOCIATE EDITOR

**HAROLD R. BJORNSEN**

COPY EDITOR

**PEG LEPAGE**

ART DIRECTOR

**HOWARD G. HAPP**

DESIGN AND LAYOUT

**ANN DILLON**

TYPESETTING

**DEBRA DAVIES**

**KEN SUTCLIFFE**

FULFILLMENT CONSULTANT

**DEBBIE BOURGALT**



# 10 Programs Included on this Disk

## *From the November RUN:*

- ▶ Friendly File Copier
- ▶ Build a Better Basic
- ▶ Wall Street 128
- ▶ Deep C-64 World

## *From the December RUN:*

- ▶ Get a Load of This!
- ▶ Graphmaker
- ▶ Season's Greetings

## *Plus: Extra Bonus Programs!*

- ▶ The Caroling Commodore
- ▶ Wraparound
- ▶ The Family Tree

If any manufacturing defect becomes apparent, the defective disk will be replaced free of charge if returned by prepaid mail within 30 days of purchase. Send it, with a letter specifying the defect, to:

**RUN Special Products • 80 Elm Street • Peterborough, NH 03458**

Replacements will not be made if the disk has been altered, repaired or misused through negligence, or if it shows signs of excessive wear or is damaged by equipment.

The programs in ReRUN are taken directly from listings prepared to accompany articles in *RUN* magazine. They will not run under all system configurations. Use the RUN It Right information included with each article as your guide.

The entire contents are copyrighted 1989 by IDG Communications/Peterborough. Unauthorized duplication is a violation of applicable laws.

© Copyright 1989 IDG Communications/Peterborough

 **IDG**  
COMMUNICATIONS/PETERBOROUGH