RE RUN

RUN Programs on Disk

For the C-64 and C-128

Two Bonus Programs Inside!

Introduction

ReRUN provides me with the honor of wearing two sombreros: Most of the time I'm an editor for *RUN*; in my spare time, I'm an editor for ReRUN. Few other computer publications establish and maintain such a close tie between the magazine and the disk-based product representing the magazine. In our case, the close relationship keeps me in touch with the variety of new ideas and articles that help me provide you, our ReRUN customers, with dynamic and useful programs.

Let's begin with a look at the *RUN* November programs in this edition of ReRUN. In the May-June 1987 ReRUN, I mentioned that programmer John Ryan was earning legendary status in Commodore circles with his expertise. John has accomplished some impressive programming feats, but these pale in comparison with 64 DOS Shell, his disk operating system for the C-64.

While many 128 owners are familiar with the 128 DOS Shell packaged with the 1571 disk drive, few 64 owners are likely to have a good working knowledge of how a DOS shell works. First of all, however, let me assure you that no programming experience or fancy skills are needed in order to use it. Just load and run, and you're ready to start making single- or two-drive disk-file copies, renaming and deleting files and redefining function keys, to name a few DOS Shell options.

DOS Shell differs from most other disk utilities in that it remains transparent in memory until needed. You can write or use other Basic programs and then, whenever you need 64 DOS Shell, just press the restore key, and it pops onto the screen. When you're finished with it, pressing the stop key brings you back to Basic. I think you'll agree that 64 DOS Shell represents one of the best programs we've published to date.

The next November program you'll find on the disk is Wordman. High school study hall was a popular place to play the word

game, hangman, which Wordman resembles, with the difference that, instead of "hanging" the man, you simply create him section by section at every incorrect guess of a letter. Wordman gives you the ability to create files of words with accompanying short definitions.

For C-128 owners, we've got Color Hi-Res Graphics, a pseudo hi-res 80 Column-mode program that was developed by Daniel Montes, an aeronautical engineer from Argentina. The program gives you new 80 Column-mode graphics commands that enable you to use all 16 colors to create some pretty fancy graphics.

I take special pride in our next November program, Snowflake. A C-128, 40 Column-mode program that lets you create snowflakes on screen and print them out, this program has a somewhat humorous history. The *RUN* editorial staff moved its office location this past summer and lost the manuscript of this program in the process. Repeated attempts to reach the author of Snowflake proved fruitless, so I took the liberty of writing the documentation in order to meet our deadline. If there's any moral to the story, it's that life in publishing might be frustrating, but it's rarely boring.

Easy Applications for November, Super Sort, is a C-64 sort program from our long-time Commodore computing contributor, Michael Broussard. Super Sort is a fast, relocatable sort routine that many Basic programmers will find useful.

Delete 64, November's Mega-Magic entry, is a short, easy-to-use program that provides an important Basic line deletion capability for the C-64.

Now to the December programs. Sizzle!, a fast-disk access program for the C-64/1541, lets you create fast-loading modules for all your C-64 Basic programs. These modules let you load programs up to 500 percent faster.

On a different note, there's Brickout! for the 64, a joystick-controlled game that lets you break down three moving walls of bricks at the top of the screen. It's a fairly sophisticated game, with a few twists thrown in, such as a shrinking paddle if you break too many bricks.

RUN Investor, a C-128 80 Column-mode program, helps you maintain lists of stocks and automatically performs calculations to determine whether you're losing or making money on them.

Next, there's Tenpin Ledger, another C-128 80 Column-mode program for keeping track of the scores obtained by your bowling team. Plenty of color and a clear, easy-to-read display help to make the program a striking (no pun intended) success.

Smart Shopper, otherwise known as REM Sheet, prints out a comparison sheet that shoppers will find useful in comparing product features. Let's say you want to buy a new product (car, boat, television, VCR, etc.) and you've narrowed down your selection to seven or eight possibilities. You can then use Smart Shopper to make a list for comparison-at-a-glance purposes to determine which product offers the most features per dollar and the best warranty.

December's Mega-Magic, Track and Sector Finder, enables you to identify the track and sector locations of your program and sequential files on disk.

Finally, since Christmas time is near, I want to share a little holiday cheer with our ReRUN audience. My Christmas gift this year is not the usual single bonus program; instead, I'm providing *two* bonus programs on this edition of ReRUN.

The first of these is called Graphics Cataloger, and it's used to catalog and print out collections of Print Shop graphics. Print Shop, a graphics editor for the C-64, allows you to create drawings, banners and a variety of other printed items. Print Shop is available for $44.95 from your Commodore dealer or directly from Broderbund at the following address: Broderbund Software, 17 Paul Drive, San Rafael, CA 94903; 415-479-1170.

The second bonus is Compu-Cash 128, a program that operates in 80-column mode. It converts your computer and printer into a fancy, point-of-sale cash register. If you've always wanted to open a gift shop or mom 'n' pop business, you might find it handy.

Well, that wraps things up for this issue of ReRUN. Have a Merry Christmas and a Happy New Year!

Tim Walsh
Technical Editor
RUN *magazine*

iii

Directory

* C-128 mode only

£ Bonus program

Before you run a program, 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.

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 whir 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 Disks

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

If the program is written in Basic, it is simple to make a copy of the program. Just load the program into your computer following the procedures outlined above, and then save the program back onto a separate disk that has plenty of free space for extra files.

If the program is written in ML, copying is not so simple. You cannot simply load and save an ML program. In this case, you'll need to use a disk-backup utility program, such as the one on your Commodore Test Demo disk.

v

64 DOS Shell

By John Ryan

Commodore owners who have purchased the 1571 disk drive received with it a most useful and versatile disk utility—the 1571 DOS (Disk Operating System) Shell. Unfortunately, C-64 users were out of luck, because the 1571 Shell works only in 128 mode. My answer to the problem is a C-64 DOS alternative, which I call the C-64 DOS Shell.

This program offers most of the features found in the 128 version, plus a few extras. Like its C-128 counterpart, you operate the 64 DOS Shell primarily with the space bar and cursor and function keys. The screen format and key functions perform almost identically to those of the 128 version, so if you're a 1571 DOS Shell user, you should feel right at home.

You can load 64 DOS Shell by using Menu 64 or with LOAD "DOS SHELL",8,1. Type NEW and enter SYS 34500. The pro-

gram will then be installed into the operating system.

GETTING STARTED

The 64 DOS Shell program is virtually transparent to the 64 operating system, and accessing it is as simple as tapping the restore key. Keep in mind that doing this while another program is running stops that program's execution and displays the 64 DOS Shell main menu, which appears as follows:

Disk Setup   Copy Files
Run a Program   Rename Files
Format a Disk   Delete Files
Clean up a Disk Function Keys
Mode Selection   Directory

Use the cursor keys to move through the menu, highlighting the options. You select and execute your choice by pressing the space bar when the option you want is highlighted. Then follow the prompts as they appear. To exit the DOS Shell, press the stop key while in the main menu.

Disk Setup—The 64 DOS Shell works with either the 1571

1

or 1541 disk drive. Although two drives may be used at one time, dual drives (drive numbers 0 and 1) are not supported. The 64 DOS Shell uses drives "A" and "B". These are logical drives, meaning that they are controlled entirely from within the DOS Shell.

You may assign devices 8, 9 or 10 to either the "A" or "B" drive. To change the settings, simply use the cursor keys and press the space bar. After you select the second drive, the program automatically returns you to the main menu. To get there without changing device numbers, press the stop key.

Keep in mind that the 64 DOS Shell looks for a second drive, even when not accessing it. If it doesn't detect a second drive on the serial bus, strange things may occur (see Error section, below).

When using two drives, the 64 DOS Shell prompts you for the logical drive number you wish to use for an operation. Make that choice by using the cursor keys and space bar. As with most 64 DOS Shell operations, pressing the stop key returns you to the main menu or verification menu, depending on the option you are using at the time.

Run a Program—When you select this option, the drive reads the disk directory and displays it in the directory window. You'll notice that this is only a partial listing. To display the rest of the directory, use the cursor keys to scroll up or down.

To select a program to run, move the red arrow to the desired filename and press the space bar. The filename then appears in reverse. To run it, press F7. To change the selection, press F5. (The latter two operations hold true for all of the 64 DOS Shell's selection processes.)

While this option runs *most* Basic and ML programs, you may experience problems with some ML programs. If you can't get the file to run, it probably means that the program is trying to occupy the same memory space as 64 DOS Shell.

Format a Disk—This option prompts you to input a filename for the format. As with most of the Shell's file operations, the program also asks you to verify your selection prior to formatting a disk. To format the disk, enter a header name (16 characters or less in length), a comma and a unique two-letter ID; then press return. For example: "filename", "ID" (standard format) or "filename" (short format).

By omitting the two-letter ID, you get a short format that erases all files and renames the disk header. Whenever you use an ID, a full disk format results.

Clean Up a Disk—Cleaning up a disk involves the Validate command. Once you've chosen the option, the validation begins immediately, so be sure you have inserted a disk into the desired drive.

Mode Selection—This option, for 1571 disk drive users only, allows you to toggle between 1541 and 1571 modes. This option can be useful within the DOS Shell for formatting, copying programs, and so on. The selected mode remains in effect until you reset the drives or select another mode. Use the cursor and space bar to choose the desired mode.

File Copy—This option allows you to easily copy files from one disk to another. The option works best with two disk drives, although it may be used with one. (If you're using a 1571 in 1571 mode and get a Drive not Ready error, set the drive to 1541 mode and try again.)

If you enter the option with two disk drives, the Shell will ask you which drive you wish to copy from. From then on, the selected drive will be the "Source," while the second logical drive will be the "Target" drive. Using the cursor, space bar and function keys, you may select one or several dozen files to copy. Pressing F7 will begin the copy sequence. Once the copy is complete, the main menu will reappear.

If only one drive is used, respond to the prompt as it asks you to insert either a "Source" or "Target" disk in the drive. The program differs from the 1571 DOS Shell at this point in that you need to format a target disk before beginning this procedure.

The 64 DOS Shell won't allow you to copy over 130 disk blocks. For files over 125 blocks, you may have to reinitialize the program. Also, the File Copy option will not reproduce relative files.

Rename Files—This option will allow you to rename a whole directory of files. Again, select from the directory listing the file you want to rename. A highlight bar appears and prompts you to enter a new filename. Do so and then press return. Repeat this process for as many files as you wish. But rename files *carefully*; duplicating a filename will result in a File Exists error.

Delete Files—This option allows you to highlight and scratch files you no longer need. Use F5 to deselect or the stop key to abort. Pressing F7 begins the scratch operation. Be very careful with this option, since you can't restore some scratched files with even the best disk-restoring utility.

Function Keys—This versatile option allows you to program any one of the function keys by

3

www.Commodore.ca
May Not Reprint Without Permission

assigning Basic keywords and short commands to the function keys in the same manner as the C-128's Key command. Once you're into the function key option, select Define Function Keys. Use the cursor to select the key you want to define. If that key has already been programmed, the assignment will appear in the input window. Pressing the space bar will erase the prior definition, allowing you to enter the new assignment. If the key is not already defined, you still need to press the space bar to begin entering the definition. Pressing the stop key aborts the procedure without changes.

Each key definition is limited to 30 characters, so use keyword abbreviations whenever possible to conserve line space. Once you've entered the line, press the return key or F7 to process it. Pressing the return key adds a carriage return to the definition, while pressing F7 adds no carriage return.

To enable the function keys, select the Enable option from within the Shell. The function keys will not work until you exit the Shell program, nor will they work while you're in Quote mode. You may disable the keys by tapping the run-stop/restore combination, or by selecting the Disable option. You may also "turn off" individual keys by en-

tering the Define option, pressing the space bar and then pressing F7. This assigns a null value to the key. Function key definitions remain in effect until the program is reinitialized (see below).

Directory—This option displays disk filenames in the directory window. No selection process is available. Press the stop key to exit.

ERRORS AND OTHER THINGS

The 64 DOS Shell was designed to be as user-friendly as possible, and it will catch almost any error you'll ever make. However, to squeeze the program into 6K, I had to make certain trade-offs. A definite program-crasher is the Device not Present error, which causes the program to freeze up and may also garble the screen. To get out of it, turn on the drive that is not present, press the return key and then the stop key *twice* to exit the Shell. Then reinitialize the program by entering SYS 34500.

For any standard disk error, 64 DOS Shell displays the error in the message window. When this happens, you need to press the space bar to get back to the main menu. Look at the light on your drive. It will remain green (red on 1541s) until you press the space bar.

The 64 DOS Shell protects the

4

area in which it resides; however, a large Basic program might sneak some variable storage into the Shell's program area (34500–40859 decimal; $86C4–$9F9B hex). Keep this in mind when running or copying large programs, for you may want to protect this area with the following statement:

```
10 POKE 51, 44: POKE 52, 135:
   POKE 55, 44: POKE 56, 135: CLR
```

If the restore key will not reactivate the DOS Shell, try reinitializing the program.

If you encounter an error that the program did not detect, the display may scroll from the screen during directory access. If this ever occurs, press the stop key to return you to the main menu. Also, remember that once you set up the Shell for two disk drives, it will always look for the second drive, whether accessing it or not. If your screen seems to freeze up, it's a good indication that a drive is not turned on. You may reset the DOS Shell for one-drive operation by reinitializing the program.

If you do as much file maintenance as I do, then the 64 DOS Shell can save you a lot of time and effort. Have fun! ∎

5

Wordman

By William Tumbleson

Wordman, a computer game that teaches vocabulary and spelling skills, will make learning fun. While it's based on the traditional hangman game, mistakes produce a picture of a character called "the wordman" rather than a gallows scene. Word skills are developed not only by playing the game, but by creating the word lists that the game employs.

Extensive error-trapping and on-screen directions make Wordman easy to use, yet it is flexible enough to meet a variety of instructional needs. The program employs the Kernal Plot routine extensively to move around the screen, particularly in drawing the Wordman character. The disk directory is displayed by a machine language routine.

The Wordman menu screen offers six options: 1) choose a word, 2) end session/game, 3) create/edit word file, 4) load word file, 5) delete word file and 6) print word file. The options are accessed by typing their numbers and pressing the return key.

CREATING A WORD FILE

Before play can begin, at least one word file must be created and saved to disk, using option 3. Press 3 and the return key at the menu, then indicate you want to create a new word file by pressing C. Next, type the number of words you intend to put into the file, press return and state whether definitions will be included.

A screen for typing in your words will follow. Type each one, and press return. Do not use hyphenated words. After you have entered all your words, the program will ask if they are okay. If not, they will be redisplayed for acceptance or correction.

When you finish with your list, and if you chose to include definitions, the program displays the first word in your list and asks you to enter a definition up to 36 characters long. The definition appears at the bottom of the screen as you type; then, after

you press return, it moves into the box just below the word. If you're satisfied with the definition, press return again; if not, press shift/return for a chance to change it.

You can bypass entering a definition for a particular word by pressing the return key twice. Then a period, instead of a definition, will appear in the area below the word.

After you've finished entering all your words and definitions, you must place a formatted work disk in the drive, enter a filename for saving the list, and then press return. The filename can be no more than 13 characters long, because Wordman will automatically add the prefix WM, followed by a space, to it. When the file is safely on disk, the menu reappears.

OTHER FILE OPERATIONS

When you want to edit an existing word list, choose option 3 at the menu, then press E, for edit, type the name of the file you want to work on and press return. The file will load, and the program will take you through the list, one word at a time. At each word, make any changes you wish and press return. When you're done with the words, you'll have a chance to change the definitions or add definitions if there were none before. Then

save the file under its previous name or a new one. If the latter, limit the filename to 13 characters to avoid an error.

Option 4 is used to load a word file. All you have to do is type the name of the file you want and press return. If you don't know which file you want, or you're unsure of the spelling, you can type $ and press return to view the disk directory. The filenames in the directory include the WM-space prefix, but you don't need to type that when entering the name of the file to load. Press return to leave the directory. After the file has loaded, the program goes back to the menu.

Option 5 is for deleting a word file. This time you do have to include the prefix when entering the filename. You can use wild cards in the filenames, but be careful you don't accidentally delete any files you wanted to keep. After typing the filename and pressing return, you must verify that the program should go ahead and delete the file you've named. If not, you can reenter the filename. Should you want to abort the deletion process altogether and go back to the menu, press return without entering a filename.

Option 6 prints a copy of the word file currently in memory. After you select this option, the

program will ask you to confirm that the file in memory is indeed the one you want printed. You can exclude definitions from the printout if you like.

PLAYING WORDMAN

To play Wordman, choose option 1 after loading a word file from disk. The program will randomly select a word from the file and display a line of as many equals signs as there are letters in the word. The display will also indicate how many words you've already used out of the file and whether there's a definition available for the current word. If there is a definition, you can view it by pressing F1.

Type your first guess for a letter in the word. If you're right, the program will place the letter in its respective position or positions in the line of equals signs. If you're wrong, the letter will be displayed near the top of the

screen, as a reminder not to try it again.

Each time you make an incorrect guess, part of the Wordman picture is drawn. After seven incorrect guesses, a tone also sounds, and after ten, the word is shown along with the completed picture of the wordman.

Every time you finish with a word, the program goes back to the menu, and you must choose option 1 again to guess another word. Once you've used all the words in the file, totals for the file are displayed, and the program asks if you want to play again. If so, it takes you back to the menu; if not, it exits to Basic.

You can also exit Wordman by selecting option 2 in the menu. Your score will be displayed, and you'll be asked to verify that you want to leave the program. If you change your mind and say no, you'll return to the menu for another round of . . . Wordman. ∎

Color Hi-Res Graphics

By Daniel Montes

C-128

The excellent Ultra Hi-Res program by Louis Wallace and David Darus in the February and May 1986 issues of *RUN* gave the C-128 a set of commands for high-resolution (640 × 200 pixels) drawing in 80-Column mode. However, this mode won't let you use all 16 C-128 colors simultaneously. To make this possible, I've written a graphics utility called Color Hi-Res Graphics, which does what you might call pseudo-hi-res drawing in 80-Column mode.

Color Hi-Res Graphics lets you define a new character and update the corresponding attribute address for that character's position. The result: a 256 × 112-pixel frame with colorful high-resolution graphics. The program's only limitation is that it allows you to use no more than one color inside each 8 × 8-pixel character.

When you run Color Hi-Res Graphics, a machine language program called CHR.GRAPH is activated. Your own applications program can load CHR.GRAPH with the BLoad command. GRAPHICS DEMO, a demonstration program, presents some applications ideas. If, while using it, you get a random pattern of characters, run 128 HIRES GRAPH *first*.

If you use other machine language utilities with Color Hi-Res Graphics, make sure each Color Hi-Res command is used in bank-15 configuration. Also, the pi symbol (π) and all control characters with a screen code higher than 63 won't appear in your listings until the original shapes and characters are restored.

NEW COMMANDS
Color Hi-Res Graphics provides seven new drawing com-

9

mands. They're activated with a SYS command, followed by an address and any necessary parameters. The address of each command can be given in either numeric or variable form.

In this pseudo-hi-res mode, the text screen is displayed. You can use the Print command if printing occurs outside the borders of the window, so a "black hole" won't appear in your hard copy. After each command is issued, the cursor position for printing text or numbers is preserved.

All commands requiring parameters check the range of the parameters and produce an Illegal Quantity error if your input is out of range.

A list of the new commands follows:

1. SET UP HI-RES AREA: SYS HR, XS,YS. HR equals 4864; XS is the column number of the upper-left border of the hi-res window (0–48); and YS is the row number of the upper-left border of the hi-res window (0–11).

This hi-res window can be placed anywhere on the screen, as reflected in the ranges allowed for the coordinates. You can't see the window, since it's the same color as the background. After this routine is called, all the standard character shapes are cleared from the window.

2. PLOT A POINT: SYS PL,XP, YP,CP. PL equals 4867; XP is the X-coordinate of the point (0–255); YP is the Y-coordinate of the point (0–111); and CP is the color of the point (0–15).

A point is plotted at the given coordinate. If another point is located nearby, and if both points are involved in the same character, they share the last color defined.

3. UNPLOT (ERASE) A POINT: SYS UP,XP,YP. UP equals 4870; XP is the X-coordinate of the point (0–255); and YP is the Y-coordinate of the point (0–111).

This command erases a point at the given coordinate, giving it the same color as the background.

4. DRAW A LINE: SYS DR,XD, YD,CD. DR equals 4873; XD is the destination X-coordinate of the line (0–255); YD is the destination Y-coordinate of the line (0–111); and CD is the line color (0–15).

Every line is drawn using this method. First, the origin coordinates must be set, usually by plotting a point at that location. Then you must issue a Draw command. The same rules for color zones that apply to the Plot command are applicable here. If this command is the first encountered after the hi-res area has been defined, a default origin of 0,0 is used.

5. UNDRAW (ERASE) A LINE: SYS UD,XD,YD. UD equals 4876; XD is the destination X-coordinate of

the line (0–255); and YD is the destination Y-coordinate of the line (0–111).

The line between the last co-ordinates and the ones specified is erased.

6. CLEAR HI-RES AREA: SYS CS. CS equals 4879.

All the shapes and characters are cleared, but the hi-res area is still intact.

7. RESTORE CHARACTER SHAPES: SYS RS. RS equals 4882.

The character shapes are re-stored, and the entire screen is cleared.

80-COLUMN MEMORY CONFIGURATION

Color Hi-Res Graphics creates designs by redefining the shapes of characters instead of using Hi-Res mode (setting bit 7 of register 25). I wrote it this way be-cause the shapes are stored in the 80-column chip's 16K RAM. Here's a brief outline of the memory configuration of this chip:

1. $0000–$07CF. Video RAM—2000 bytes used to display screen characters.

2. $07D0–$07FF. 48 unused bytes.

3. $0800–$0FCF. Attribute RAM, a 2000-byte area that holds the attributes of the corresponding video RAM, including character color, underlining and flashing.

4. $0FD0–$1FFF. 48 unused bytes.

5. $2000–$3FFF. CHARRAM (character generator RAM)—an 8K area where character shapes are defined.

In 80-Column mode, you can define characters and display them anywhere on the screen by just poking them into video RAM. However, not all the shapes and characters can be modified. I preserved the first 64 definitions, including all the letters in upper-case, numbers 0–9, all the punc-tuation marks and some others, so you can include text with your graphics.

Here's how I calculated the number of characters available for redefinition. There is a total of 512 characters (256 each for upper- and lowercase). I sub-tracted 64 (the number of char-acters not modified), leaving 448. This allows for a 32 × 14-char-acter window anywhere on the screen, where you can put the available shapes and characters, one after the other.

The net result is a "pseudo" hi-res frame of 256 × 112 pixels. The frame occupies about one quarter of the screen, which is perfectly adequate for many applications. ∎

11

Snowflake

By Ian Adam; article by Tim Walsh

Ian Adam wrote the article that originally accompanied Snowflake 128. Unfortunately, the original manuscript and disk vanished, and we can't contact Mr. Adam, who, according to his office colleagues, is vacationing in "the islands." Rather than postpone publication of the program until Ian Adam can be reached, I have undertaken to write this documentation.

A program without equal, Snowflake 128 lets you design snowflakes in 40-Column mode on the C-128. Load and run the program; then, after the introductory snowflake appears, try your hand at custom designing snowflakes on the screen. A menu at the bottom of the screen lets you print them out on any Star Micronics or Epson-compatible printer.

Snowflake 128 also has the advantage of letting you enjoy the beauty of snowflakes any time of the year, regardless of where you live. If you live in a snow-belt region, you can hold contests to see who can design the most elegant snowflake. The prize could be a day off from shoveling.

Of course, if you live in Hawaii or Bermuda, such contests aren't very practical. In that case, on a warm January afternoon, when the tropical breezes are gently swaying the palm trees, go inside and boot up Snowflake 128. And, as you're creating those lovely snowflakes, picture the *RUN* staff up in New Hampshire, buried under tons of white stuff, and try not to laugh. ■

Super Sort

By Michael Broussard

If you write Basic programs, sooner or later you'll be faced with the chore of sorting arrays—lists of names, for example. My all-purpose sorter, called Super Sort, can be used as a subroutine in your own programs to sort any type of array.

Super Sort is short (only 332 bytes) and fast, especially when sorting large numbers of strings. In one test I ran, Basic took nearly 29 minutes to sort 600 strings, while Super Sort did them in 41 seconds—42 times faster than Basic! Table 1 shows the relative speed of Super Sort compared to Basic when sorting various lists of 600 elements.

For a demonstration, run Super Sort. Line 100 is the important part of the program; it pokes the machine language subroutine into the proper memory locations. The Data statements numbered 1000–1180 define the subroutine. To use the sort, these lines must be included in your own program.

Once it's been poked into memory, the sort is invoked with a SYS statement like this:

SYS SRT,X(1),100

SRT represents the starting address of the machine language subroutine (in either variable or numeric form); X(1) means the sort should start with the first element of the array; and 100 indicates the number of elements in the array.

To sort only part of the array, use a statement like the following:

SYS SRT,X(21),10

This example would sort elements 21–30.

For Super Sort to work properly, you must specify which element of the array it should start with, because X and X(1) are different variables as far as Basic is concerned. One is a single floating-point variable, and the other is an array. Attempting to sort a nonarray value may corrupt other variables in your program, and even lock up the computer.

Similarly, don't try to sort 200 elements of an array dimen-

sioned for 100 elements. Super Sort will obligingly attempt to sort past the end of the array, with potentially disastrous results.

If you're sorting an array of strings, you can specify which character of the string should be the starting point for comparison. For example,

SYS SRT,X$(1),100,3

specifies that the first 100 elements of array X$ are to be sorted, beginning with the third character of each string. Any strings less than three characters

long are sorted to the beginning of the array in arbitrary order. If you don't specify a starting character position for the sort, Super Sort defaults to the first character. Bear in mind that the concept of starting position holds true only for string arrays. The following statement will generate a syntax error, because N is not a string array:

SYS SRT,N(1),100,2

As a bonus, Super Sort is completely relocatable. You can place it anywhere in memory just by changing the value assigned to SRT on line 100. Just be sure to use a memory location that doesn't interfere with Basic.

To keep Super Sort as short as possible, I used many subroutines that are part of the C-64's Basic ROM. For this reason, the program will run only on a C-64 (or a C-128 in 64 mode). It uses the following zero-page locations for temporary storage: 7, 8, 20, 21, 25–32 and 251–254. ∎

Element type	Basic	Super Sort
String	28:42	00:41
Floating Point	25:46	02:45
Integer	26:58	03:34

Table 1. Speed comparison, in minutes and seconds, between Basic and Super Sort, sorting lists of 600 elements.

Delete 64

By Richard De A'Morelli

Few programming procedures are as monotonous as manually deleting lines from a C-64 program. Although some computers have a block delete command for erasing lines in a specified range, the C-64 does not.

My program, Delete 64, eliminates this headache. You can append it (it's only 13 lines long) to any program you're developing, then use it to delete dozens, even hundreds, of lines quickly and easily when the need arises. I always load Delete 64 into memory before I start writing a new program, so that it will be there whenever I need it.

After you've saved a copy of this utility, load it, then activate it by typing in RUN 63980 in Direct mode. You'll be prompted for the block you want deleted and the increment. If the line numbers are irregularly incremented, use an increment of 1.

While Delete 64 is working, line numbers will quickly flash by in the upper-left corner of your screen as the lines in your program are deleted. A message will advise you when the operation is finished. Then list the program to make sure all the lines in the target range were erased. The whole process occurs with remarkable speed and is fully automatic. ∎

15

Sizzle!

By Ray Roberts and Philip Bacon

Commodore's 1541 disk drive, as most C-64 users know, is slow at transferring data from the disk to the computer. You might guess that the problem lies in the serial interface between the drive and the computer. A serial bus transfers a single bit of data at a time, while the more common parallel bus transfers a full byte (8 bits) at a time. The serial configuration was apparently chosen by Commodore because it's less expensive to manufacture.

Curiously enough, the serial bus is not the cause of the slow data transfer. Indeed, the bus is capable of transferring data at up to 12 blocks per second. The reason it doesn't is that the built-in software that controls loading and saving is inefficient and doesn't use the hardware to its full potential.

The result is that during a normal load, the 1541 transfers about 1.5 blocks each second, and a 100-block program file will load in about 65 seconds—which is about eight times longer than it theoretically could.

FAST LOADERS

It didn't take long for some programmers to figure out the problem, and "fast-loader" programs soon became available. They work by redirecting the load vector in the computer's memory to the fast-loader. When you enter a Load command, a portion of the fast-loader is sent through the serial bus to the disk drive RAM, where it's used instead of the normal, inefficient load routines. Once there, it works in conjunction with the rest of the fast-loader program, still in the computer's memory, to transfer the data.

To use a fast-loader, all you have to do is load it and issue a SYS command to redirect the load vector. Thereafter, loading speeds will be two to five times faster than normal, depending on the size of the program. Fast-loaders are able to read two

blocks every time the disk spins one revolution. They don't quite reach the theoretical limit, because it takes time to send the program to the disk drive and to move the head from track to track, but there is surely a significant improvement.

The problem with fast-loaders is that they reside in the computer's memory. If you load a program that needs to occupy the same address as the fast-loader, everything will come to an abrupt halt in the middle of the load.

At other times, a program will fast-load without any difficulty, but if it saves data in the same areas as the fast-loader, it will corrupt the fast-loader, and the next time you attempt a load, you'll probably end up with a "frozen" computer. These problems are exasperating and lead many users to put up with the normal slow speed of the 1541.

The solution to this quandary is my program, called Sizzle. It'll create a variety of FastDOS loaders—one for each of your application programs—and they'll operate at two to five times the speed of the normal 1541.

PROGRAM OPERATION

When you run Sizzle, the main menu will offer four options. The first pokes the fast-loader into memory—at any starting address you designate—and activates it. This is just a quick way of enabling a fast-loader without having to boot it from the disk.

The second option generates fast-loader program files with various starting addresses that you designate. With this option, you're no longer locked into using a single fast-loader. You can create a number of them, then each time you need one, choose one that won't conflict with your application program.

If you created a Sizzle Fast DOS that resides at address 49152, you'd load and activate it with the following sequence of commands:

LOAD"SIZZLE.49152",8,1 <return>
NEW <return>
SYS 49152 <return>

A SIZZLE ON message would appear, and any subsequent Load operation, except a directory ($), would use the Sizzle FastDOS at 49152.

The third option creates fast-loader boot files for your application programs. Sizzle reads the directory of your applications disk into memory and displays it in a window. Then you move the pointer to the program you want to fast-load, press return and, at the prompt, enter a name for the boot file.

You also must specify whether you want the application pro-

17

gram to run automatically after it's loaded; whether you want Sizzle disabled after a single program file has been loaded; and where in memory you want Sizzle to reside. If you like, you can place Sizzle beneath Basic ROM or in screen memory, where you can watch the loading process.

To load a 1231-byte Sizzle boot file, enter its filename, followed by the usual ,8,1. It will load through the tape buffer and on up through screen memory, filling the display screen with unintelligible characters in the process.

The first two bytes of the boot file will be written over the CHROUT vector at address $0324, so after the file has loaded and the computer has printed READY, the Load routine will be redirected to a machine language program in the tape buffer. The FastDOS will then be either copied from the screen to the appropriate location for execution or will be executed in place, and the application program will be fast-loaded into the computer.

The fourth main menu option is to exit Sizzle.

PROGRAM NOTES

Here are some things to keep in mind when you're using Sizzle:

1. Run-stop/restore disables Sizzle by resetting the load vector to the normal address.

2. You can toggle Sizzle on and off by issuing a SYS command to the starting address. For example, if Sizzle were located at $C000, SYS 49152 would turn it off or on. As you toggle, SIZZLE OFF and SIZZLE ON appear on the screen.

3. When you load a disk directory ($) with Sizzle activated, execution reverts to the normal load routines—except when Sizzle is located under Basic ROM ($A000). In that case, the system crashes. To circumvent this problem, view the directory with DOS 5.1 or with a similar wedge program.

4. If Sizzle is located under Basic ROM, it can't be toggled on and off, but you can permanently disable it with the run-stop/restore combination.

5. If the program you want to fast-load doesn't clear the screen or print to the screen while it's loading, it will load flawlessly if you prepare the Sizzle boot to execute in screen memory. You should never pick the screen as the location for Sizzle if your application program does anything to the screen while it's loading.

6. One of your options when preparing a Sizzle boot is to disable Sizzle after loading a single program file. If you execute a Sizzle boot in screen memory without doing this, the load vector will remain pointed at screen

memory after the load has been completed. Then, if you try to load another program without first pressing run-stop/restore, you may expect a major crash.

7. The disable option can be useful for word processors that tend to use every corner of memory and thus may overwrite the fast-loader. Many word processors also save text in program files and use the load routines to get the text into memory. With the fast-loader disabled, a word processor can load text normally.

8. If you load Sizzle into the top of Basic RAM, you'll first need to lower the top of Basic by poking the starting address of Sizzle into locations 55 (lo) and

56 (hi). Otherwise, Sizzle will be overwritten by Basic variables. If you select the top of Basic RAM for a Sizzle boot, the top of Basic is automatically lowered by the boot program before Sizzle is copied there.

9. Many commercial copy-protected programs don't use the normal load vector when loading multiple program files. In such cases, Sizzle loads only the first file in the series faster than normal. If the program does use the normal load vector, all the files are fast-loaded.

10. Sizzle works with the 1571 disk drive, as well as the 1541. However, programs that have been saved on a 1571 load more

Program Size	Normal Load	Sizzle Load (Factor)	Sizzle Boot Load (Factor)
27 blocks	18 sec.	5 sec. (3.6)	9 sec. (2.0)
54 blocks	35 sec.	8 sec. (4.0)	12 sec. (2.9) .
90 blocks	59 sec.	12 sec. (4.9)	16 sec. (3.7)
113 blocks	72 sec.	14 sec. (5.1)	18 sec. (4.0)
194 blocks (5 files)	127 sec.		33 sec. (3.8)

Table 1. Loading time comparisons.

slowly than do those saved on a 1541. The reason is that the 1571 saves successive blocks of the file six blocks apart around the track (interleave = 6), whereas the 1541 separates them by ten blocks (interleave = 10). Sizzle isn't fast enough to pick up successive blocks if they're only six blocks apart, so it has to pick up the next block on the next revolution. You can change the interleave of the 1571 to 10 before saving a C-64 program by sending "U0>S" + CHR$(10) over the command channel.

11. One of the first things you should do is make a fast-boot for the Sizzle program. It will then load in about ten seconds. ∎

Brickout!

By John Fedor

Tired of shooting aliens? Bored with lengthy sets of instructions? If so, boot up Brickout, a Pong-type game in which you score as a ball knocks bricks out of a wall. The challenge comes in using an on-screen paddle to keep the ball in play as it bounces around; the longer the ball's in play, the more bricks it knocks out. You get three balls per game, and when you've let all three escape, your turn is over. Brickout offers all the components of an exciting arcade game, with multi-player competition, a choice of difficulty levels and colorful graphics combined with imaginative sound effects.

Begin by loading and running Brickout. Now, with a joystick plugged into port 2, you're ready to play.

As the game begins, a short tune plays and the title screen appears. Enter the number of players (1–4), each player's

name and the speed at which he or she will play. The speed ranges from 1 to 3, with 1 being the fastest.

The next screen shows how many balls the current player has left. It also lists all the players' names and scores, with the active player's name highlighted.

Press the fire-button to set the ball in horizontal motion below the bricks, then press the button again to start the ball bouncing. Once it's bouncing, move the joystick left and right to slide the horizontal paddle across the bottom of the screen and keep the ball from escaping. If you miss the ball, it's lost.

The bricks are arranged in three rows of different colors: red, blue and green. Points, based on color, are awarded for knocking out the bricks. You receive five points for red bricks, ten for blue and 20 for green. When you've eliminated all the bricks of one color, they're automatically replaced.

When all the players have used up their three balls, the

game is over. The final scores appear, and two options are presented. Pressing the fire-button restarts the game with the current settings; pressing return lets you choose new options before resuming play.

Note that the bricks of different colors scroll across the screen at different rates and in different directions. This makes it possible for the ball to reach the upper levels before the lower levels are completely gone. Also, the paddle shrinks with every 1000 points earned, which increases the level of difficulty for advanced players.

Now, get busy and warm up your throwing—er, joystick—arm! ∎

RUN Investor

By Stan Krawczyk

I wrote RUN Investor, a C-128, 80-Column mode program, to keep track of my stocks on a daily basis. It tells me how much each of my stocks is worth at its current price and whether I'm making or losing money on it. For instance, say I buy XYZ Co. stock for $10 a share in June, $11 a share in July and $9.50 a share in September. RUN Investor will tell me the value of each transaction if the stock is currently selling for $10.50 a share, as well as the total market value at that price.

RUN Investor supports one or two disk drives (devices 8 and 9) and the Commodore 1525/MPS-803 or 1526-compatible printer. It's menu-driven and easy to operate, and it uses the C-128's window capabilities.

I've written the program in logical sections, so you'll find it easy to locate the code for different functions. For example, the List File option, which is activated by pressing F3, is located in lines 300–400 and 1300–1400 and the Update Stock Price option, F7, is located in lines 700–800.

Note that the function keys, F1–F8, are redefined in lines 5 and 6 to their C-64 values. The number of disk drives is determined automatically in line 7. If the inquiry to device 9 results in a status of 0, the program sets itself up for two-drive operation. Otherwise, it uses just one.

As written, RUN Investor accommodates up to 100 stock names. However, you can increase this number as your needs grow by changing the DIM values in line 4:

CREATING A STOCK FILE

To use RUN Investor, key in the date and initialize a stock file by pressing the F5 key. Then enter, in turn, the name of the company whose stock you want to buy or already own, the number of shares involved—stored in variable SO(I)—and the cost per share. The number of shares can

be fractional, down to a ten-thousandth of a share—for instance, 7.1275 shares. When entering the cost per share, use multiples of .125 to represent eighths of a share. You can write your data to disk (device 8 or 9, or both) as soon as you've finished entering it, or you can wait until later. Press the F8 key to execute the save.

To see the total market value of your stock, press F3. You can print out a report from this option, too.

To create another file, perhaps for another member of your family or to follow some stocks you may want to buy later, press F5 again. Just remember to save it to another disk and to label it accordingly. The program writes the filename Stock File to disk, so make sure your label indicates which stock file. Any previous stock file on a disk will be overwritten when a second save is done.

To quit the program, press the F6 key. If you haven't saved your file, the program will remind you, so the file won't be lost.

The upper-left portion of the RUN Investor screen displays the program and error status. This is done with a POKE 235,0 and PRINT"''', which eliminates any character flash on the screen and places the next print on line 1 of the screen.

UPDATING YOUR FILE

The fun begins when your daily paper is delivered and it's time to see how your investments are doing. With the stock section of the paper in hand, load RUN Investor, then enter the date, press F1 to load your file and F7 to update the prices. Your stocks will be listed one at a time, with a prompt for the current quote. If a particular stock hasn't changed in value, enter N or hit the return key; if it has changed, enter Y and the new price, then verify the amount when you are prompted. After you've updated all your stocks, you can save the update to disk, or wait until later. Press the F3 key to see your gain or loss on each issue of stock and to print out a hard copy.

Say some time lapses and you buy more stock. All you have to do is load RUN Investor, press the F4 key and follow the prompts.

If you decide to sell some stock, press the F2 key and answer the prompts. The program will total all shares of the stock name you want to sell, ask how many shares you plan to let go (in whole amounts or fractions down to a ten-thousandth of a share) and then subtract the number you've specified. When all the shares are gone from one issue, it will go to the next issue,

as long as the stock name is the same. When an issue is completely gone, all the variables move up one and automatically delete the entry from the file in memory.

If you have a stock file on a second disk, you can load it by pressing F1. The program will remind you if you already have a file in memory. If you want to go ahead and load the new file, change disks and the loading will proceed.

Right now I use RUN Investor's Stock Purchase option (F4) to record dividends reinvested automatically by my broker. When the program asks for a stock, I enter the number of the stock that paid the dividend, then the number of shares bought (the dividend), a purchase price of $0.00 (since it's a dividend) and the price at which the dividend was issued (from my broker's report). If the program arouses enough interest, I plan to expand RUN Investor to handle dividend reinvestment and tracking, and to graph selected periods of a stock. ∎

Tenpin Ledger

By Barbara Schulak

Bowling is a ball, especially during the cold and wet winter months. If you often head for the alley and keep records of your games, you'll love Tenpin Ledger. Written for the C-128 in 80-Column mode, this spreadsheet-like program stores game scores, calculates total pins and averages for each bowling session and keeps a running total and average from session to session. By creating a file for each individual or team, a league secretary could easily use this program to keep track of individual averages and team scores.

THE MAIN MENU

When you run Tenpin Ledger, the main menu, with six options, appears first. You can load a file, save a file, examine the ledger pages to make additions or changes, view the disk directory, print a file and exit. Just follow the prompts to execute any of these choices. Pressing only the return key at any prompt sends you back to the main menu without executing a routine.

You'll need a formatted work disk to save your Tenpin Ledger files. The program creates sequential files and automatically adds the suffix .tp to their names to distinguish them from other files. You don't need to type this suffix with the filename when loading or saving a file.

DATA INPUT AND CALCULATIONS

Tenpin Ledger provides four ledger pages. Each page has room for 13 bowling sessions, so the program can handle a total of 52 sessions. Column headings for data you input include date of the bowling session, number of games rolled (1–3) in that session and scores for those games. Other headings, for which the program calculates the data, include total pins and average for the session, and cumulative number of games, pins and average. The bottom of the screen lists the various commands available in this editing mode.

26

When you access the ledger from the main menu, the first page of the ledger appears on the screen. Use the cursor keys to place the cursor on the cell where you want to enter or change data, then press the up-arrow key. When the word "enter" and a flashing cursor appear on the purple status line, type the data you want in the cell and press return. Enter the date in MO/YR format, such as 06/06 for June 6. If you press the return key without entering any data, the contents of the cell remain unchanged.

After you've entered all your data and made all your corrections, press the back-arrow key to make the program calculate the rest of the ledger sheet.

To move between the four pages of the ledger, press the control key and a number key (1–4) simultaneously. The current page number and filename are displayed on the top line.

When you're viewing a ledger page, you can find any games above 200 and any series score above 500 by pressing control/F. The scores will be highlighted in light red.

You can also erase the entire contents of a file to create a new file by pressing control/C. Since all current data is lost unless it's saved to disk, you'll be prompted to verify this choice.

Any time you want to return to the main menu from the ledger, you do so by pressing M.

Note that Tenpin Ledger's printing and calculating routines stop when they come to a zero in the number-of-games column, so make sure you don't skip any rows when entering your data.

Happy kegling! ∎

Smart Shopper

By Jerome Reuter

On a scale of one to ten, how do you rate yourself as an intelligent shopper? How often do you find a better product at a lower price right after you've made a purchase? You can prevent such unhappy occurrences by comparing before you buy; that's where REM Sheet helps.

REM Sheet is a Basic program that will print out a comparison-shopping worksheet that you can take with you to the store. The program doesn't care if you're shopping for a new printer, a database, a washing machine or an automobile; and it's versatile, letting you include up to 12 products and 12 comparison categories. Your printer must be a dot matrix machine with graphics keys, such as the Commodore MPS-801, -803, -1000 or -1200, or else a compatible non-Commodore printer.

When you run the program, it will ask you to enter the number of products you are interested in and the comparison categories you want, plus an optional title. Then it will produce a printout with the title line, a grading scale and a grid where you check off your findings.

The top of the grid is labeled with the features you entered, and down the left side are lines for you to write in the names of the products. On the right, there's one blank column for penciling in some feature that didn't occur to you until you got to the store, as well as a Totals column for summarizing your findings.

REM Sheet has many other uses besides comparison shopping. I find it handy for listing the months of the year across the top and my monthly bills down the left side. Then I can tell at a glance which bills I've paid and which will soon be due. I'm sure you'll find other uses for REM Sheet, too. ∎

Track and Sector Finder

By Joseph R. Charnetski

My utility program is designed to give you the disk track and sector of any sequential file or Basic or machine language program.

When you run the program, enter the complete filename (no wild cards); then enter P or S for the type of file you want to lo-cate. The program will open the directory and read the two bytes immediately preceding the file-name you've entered. These two bytes hold the track and sector information for the first block of your file. They'll be displayed on the screen, and if there are more blocks, the program asks if you want to see the track and sector of each remaining block. If you answer Y, the other blocks will also be displayed. ∎

Graphics Cataloger

By Frank Colaricci

C-64

I wrote Graphic Cataloger for owners of Broderbund's The Print Shop, The Print Shop Companion and Graphic Library Disks who use side A of the disk with a Star Gemini-10X, Star SG-10 or almost any other Commodore-compatible printer. With this program, you can select and preview Print Shop graphics on your screen and then catalog them on your printer.

Graphic Cataloger is written entirely in machine language. I've tested it successfully with a C-64, a Cardco G-Wiz interface with all switches off and a Star SG-10 printer with all switches on.

CHOOSING FILES

To use the program, load and run PSGC.A.RUN. The program will greet you, remind you to align the paper in your printer and prompt you to insert into the drive a write-protected disk containing the Print Shop graphics you want to catalog.

After inserting the graphics disk, press the return key to load the disk directory into memory. Then a menu of function-key commands and a list of up to 16 three-block program files will appear on the screen. Select the filenames you want by pressing the letter, A–P, preceding each one. You can deselect a filename by pressing the letter again. When you select a file, its name changes to reverse print.

F1 displays the next group of filenames. This function has a wrap-around feature, so the first page follows the last page.

When you've finished selecting filenames, F2 loads memory with up to 32 Print Shop graphics and displays the Cataloger menu on the screen.

F3 toggles all of the items on a page.

F4 quits the program, giving you a chance to reconsider in the process.

CATALOGING

You choose an option in the

30

Cataloger menu by pressing the key for the first letter of the option.

C (Catalog) dumps the disk name, disk ID, filename and graphic to your printer.

After the dump has been completed, G (Get) displays the next file in memory as a graphic, so you can confirm that it is indeed a graphic. This is important, because, while all Side A Print Shop graphics are three-block program files, not all three-block program files are Print Shop graphics. When you've processed the last graphic in memory, the program loads up to 32 more graphics for you to select and catalog.

N (New) positions the next sheet of paper in the printer. The program also does this automatically after the predetermined number of graphics have been printed on a page.

Q quits the program. Here again, you have a chance to reconsider. This command also restores the printer linefeed spacing to normal for text.

When the last three-block pro-gram file you selected has been processed, Graphic Cataloger displays a message to that effect and restores the printer linefeed spacing for normal text. If you want to catalog another disk, align the printer paper at the top of the page and type RUN again.

DEVICE NUMBERS

By editing the Data statement in line 1160 of the program generator, you can change the printer device number (second data item), printer secondary address (third data item), device number of the disk drive you'll use for loading graphics (fourth data item), number of graphics per page (fifth data item) and number of linefeeds per page (sixth data item). All the numbers in the Data statements are in hexadecimal. If you change any of the above parameters, please change the checksum on line 1110 accordingly.

To generate the machine language code, place a work disk in your drive and type RUN. The code, with changes, will be complete in about $1\frac{1}{2}$ minutes. ■

CompuCash 128

By Wayne Arnett

Whether you own a small business or sell disks at your user group meetings, keeping track of cash without a computerized cash register presents a real problem. CompuCash 128 provides an easy solution to this problem by converting your C-128 into a full-featured, four-department cash register and inventory tracking system.

With the exception of accessing optional features, the program lets you complete the entire process of ringing up a sale by using only one hand on the C-128's numeric keypad. With the keypad and the function keys, you can enter up to eight item sales, discounts and refunds per customer, including multiple-item purchases, such as five of one item at $1.98 each. Sales can be taxed at either of two rates, including tax-free.

The computer screen displays activity by product category and also gives a running total of the dollar volume. CompuCash 128 even includes a keyboard lockout, so you can safely leave the cash register unattended.

It also prints out a detailed receipt, with your own promotional message, for every customer, as well as a daily activity summary for you. This summary lists each sale by product category, price, discount amount and the clerk who handled the transaction. For inventory purposes, CompuCash 128 provides an item count, with net sales for each product category listed by amount and percent of total.

USING COMPUCASH 128

First of all, you may find that program startup and the first sale seem to plod along. Don't worry; the speed will pick up later.

The program begins by asking you for the date in MMDDYY format (for example, 070587 for July 5, 1987). The date is important, because it appears on customer receipts and daily summaries. However, you can by-

32

pass date entry by pressing the return key, which enters a default of January 1, 1988.

Next, you must input a two-character code that identifies the shift and who's running the cash register. For example, if, as the owner of the business, you're user #1, and you're working the B shift, the code might be B1. The ID should be meaningful, as it appears on all the printouts indicating who handled any given sale. This code is also used to regain access to the system after a security lockout. If you're the only operator, you can press the return key to enter the default, A1.

After logging on, you proceed to the main menu, which offers six options and, in the upper-right-hand corner, displays the number of transactions you can still input. Press a number key to choose any of the options, or press return to choose option 1. Either way, the keypress will take you to the transaction-entry screen, where customer sales are entered. I'll cover the sale-entry procedure in more detail later.

Option 2 brings up the recap screen, which provides an overview of the day's activity. This summary includes the number of customers (transactions), individual item sales, net dollar amounts by product category, and refunds or discounts. From the re-cap screen, you can proceed directly to transaction entry by pressing the return key, or you can tap the back-arrow key to view the menu again.

To print out a daily activity summary, use option 3. The printout report may be in short form—just a hard copy of the recap screen—or in long form, which includes a detailed record of every transaction and item sold. Sales tax information is also provided. By adding your change fund to the totals, you can verify ending cash balances. This summary can be printed out as often as you like during the day or only at the end of the day. Access to the print menu is denied unless at least one sale has been entered.

Use option 4 to change the date or log in a new clerk. When you're in this option, pressing return will accept previously entered information.

If you're called away from the keyboard for any length of time, you should use menu option 5. It prevents any entries from being made until you return and enter your ID. In case you forget your code after engaging the lockout, it's shown at the top of the screen, disguised amidst some other information. For this reason, blank spaces are not allowed in IDs.

The final menu option exits

from CompuCash 128. As a safety feature, confirmation is required. If you change your mind about exiting, you'll be sent back to the main menu.

RINGING UP A SALE

As I mentioned, pressing 1 or the return key at the main menu brings up the transaction-entry screen. The top row of this display shows the four product categories you specify, along with their corresponding F (function) keys. The second row displays the date, user ID and a running count of transactions and sales, both entered and still available. The keyboard status (R for ready or L for security lockout) is also shown. At the bottom of the screen, you'll see instructions that change as you proceed through ringing up a sale.

At this point, you're prompted to press an F key to enter a sale or the back-arrow key to back up to the main menu. (The back-arrow key will usually help you change your mind when you find yourself in the wrong place.) Say your F1 product category is hamburgers, and you've just sold one. Press F1, and the word "hamburgers" will appear on the screen. If you strike the wrong F key, enter R to reselect. At the item-price prompt, type in the amount and press return. The right edge of the screen will dis-play the sale as it's entered, along with a continually updated running total.

Next, a window pops up and presents several options. If the sale requires no special treatment, just press return; the sale will be entered as is and taxed at the regular rate. A beep will warn you if only one more item can be entered.

The other options are Discount/Coupon, Percent Discount, Tax Rate and Correction. Press D to enter an actual discount amount or P for a percentage discount; either can be zero if you change your mind. Press T to assess the alternate tax rate on this particular item. You must press T for each specially taxed item. Finally, press C to correct (erase) the current entry.

This is your second chance to correct either the product category or the amount. R, to reselect, was the first, and you'll have a third opportunity after you've completely entered a sale. At that time, when you're prompted for the next item, press any F key to get the item-price prompt. Then enter F to fix the last item, and you'll be back at the previous sale.

After you're done with the special-treatment window, you can repeat the transaction-entry procedure if the customer is buying another item, or you can press

34

return to finalize the sale. If you press return, the price, discount and tax amounts will be listed and totaled, the program will prompt you for the amount tendered and will then display the amount of change due. If the exact amount is tendered, press return to bypass the change-calculation feature and save the trouble of typing in the amount.

Should you arrive at this point by mistake, or the customer suddenly decides to purchase another item, enter a back-arrow at the change-due prompt. This will bring up the transaction-entry screen, unless you've already reached the maximum allowable number of items.

Once the transaction is correct and complete, you must decide whether to print a receipt. The default is yes. To bypass printing, press any key except Y. You can also enter a back-arrow to correct an error in the amount tendered.

Every customer receipt shows the date, transaction number and clerk code, as well as the name of your business, a detailed record of the transaction and a promotional message. For the customer's convenience, the total amount due, amount tendered and change due also appear.

SPECIAL FEATURES
Several special features are available for transaction entry. One is the multiple-sale feature. Suppose a customer buys three items for $1.25 apiece. Rather than entering 1.25 three separate times, or mentally calculating the total, just type in 3 + 1.25 <return>. This will be interpreted as "three items at $1.25 each" and entered as 3.75. The multiple-sale feature also works if you sell products by the pound, although fractional quantities are rounded to the next highest integer on the receipt. The number of items involved in a sale is shown on both the receipt and the daily report. While the @ symbol might seem more appropriate for multiple sales, it would require moving your hand from the numeric keypad, so I used the plus key instead.

To enter a refund, first select the proper product category. At the item-price prompt, type a minus sign before the refund amount. This entry will be processed like any other; you can even select alternate tax rates to duplicate the conditions of the original sale. However, you cannot discount a refund; always enter the net amount before tax.

Follow the same procedure for an exchange. First enter a refund for the returned merchandise, then enter the price of the replacement item. If any price adjustment is required (either pos-

itive or negative), it'll show up under amount due. Refunds and exchanges are tracked on the recap screen and daily report.

If an error slips past all the correction options I mentioned earlier, the refund feature offers one last chance to salvage the situation. Say you enter a sale in product category F3 for $10, which is wrong, and then ring up two more items before noticing the mistake. You can make the next item sale a refund to reverse the error, then enter the first item correctly. You can also use a refund if a customer decides not to buy something you've already rung up or comes up short of money.

If you accidentally press an F key after a customer's last item has been recorded, you don't have to back out of the entry screen to finalize the transaction. Just hit the return key at the item-price prompt, and you'll proceed directly to the amount-tendered prompt.

CUSTOMIZING

Starting at line 140 of the program, substitute the name of your business, its address and any other information you want to appear on customer receipts. Four Data statements are optimal, but you can have up to six. If you add Data statements, squeeze them in between the ex-

isting ones, so subsequent program line numbers will remain unchanged. Also, be sure the final Data statement consists only of the word END.

Now compose a promotional message to be printed on each receipt. The message might be a thank-you, a notice of new store hours or an announcement of a coming event. Change it as often as you like, but again, limit it to six lines and always finish with END.

Next, insert the sales tax rates for your area into the program. In line 250, the variable TR is the primary tax rate automatically charged to every sale. TA is the tax alternate offered in the special-treatment window. Either or both can be zero for untaxed sales. If you like, change CK$, the default clerk ID, to any other two-character combination.

Now, specify in lines 260–290 the names you want for product categories. If your business is a snack shop, the categories might be Sandwiches, Side Orders, Drinks and Miscellaneous. If it's a small motel, they might be Room Rent, Telephone, Dry Cleaning and Misc Charges. Regardless of your choices, each entry can be no longer than 16 characters.

Finally, specify values for NT, the maximum number of transactions, and NS, the maximum

number of sales per transaction, in line 250. Keep NT as small as possible, because several arrays are dimensioned to it and memory may be consumed rapidly. More importantly, execution speed begins to suffer at larger values.

Out of Memory errors would probably occur if the number of transactions exceeded 200, so I've made the program set 200 as the maximum. This number provides for one transaction every 2½ minutes during an eight-hour session, which should be more than adequate. If you make the maximum less, you'll enjoy a faster-running program. Due to screen-size limitations, no more than eight item sales per transaction (NS) are allowed.

After a customer receipt is printed, the paper automatically advances so you can tear it off. Depending on your printer, you may want to adjust the value of AP in line 2660 to change the number of linefeeds the paper moves.

Since the disk drive isn't used during program operation, you can save wear and tear on it by turning it off after the program is loaded. ■

14 *RUN* Programs Included on this Disk:

DOS Shell for 64 ► 128 Color Hi-Res Graphics
C-64 Fast Loader ► Games ► 64 Sorter
128 Bowling Score Recorder ► 128 Investment Program
128 Cash Register ► Disk Utility

From the November RUN:

- ► DOS Shell 64
- ► Wordman
- ► Color Hi-Res Graphics
- ► Snowflake
- ► Super Sort
- ► Delete 64

From the December RUN:

- ► Sizzle!
- ► Brickout!
- ► RUN Investor
- ► Tenpin Ledger
- ► Smart Shopper
- ► Track and Sector Finder

Bonus Programs:

- ► Graphics Cataloger
- ► CompuCash 128

CW COMMUNICATIONS/PETERBOROUGH