# PROSYS

## The Professional Systems People And

## MICRO WORX

## Present Products From

# commodore

## And

## The Software That Makes Them Work!

### SOFTWARE

**SBSYS**
C-64, 8032, 8096 & B-Series
THE SMALL BUSINESS SYSTEM
Available for 1541, 8050 and hard disk drives. GL, AP, AR, INV. and payroll as low as $99.00 each! Call for specific pricing.

**PERSYS**
VIC 20, C-64, 8032, 8096 & B-Series
THE PERSONAL FINANCIAL SYSTEM
A complete financial package for home and small business, beginning at $69.00 on tape.

**VERTICAL PACKAGES INCLUDE:**

**LEGISYS**
8032, 8096 & B-Series.
The total legal office information, accounting and tickler system.

**LOADSYS**
8032, 8096 & B-Series.
The total truck brokerage accounting system. Call for free intro consulting.

Dealer inquiries invited.

VISA & MasterCard. Add 3% Surcharge.
Shipping paid on prepaid orders.
Prices subject to change without notice.

These are sample unit prices.
We carry support items, cables, games...
**WE HAVE IT!**

### CBM PRODUCTS

| | |
|---|---|
| 8032 Computer | $ 619.00 |
| 8050 Disk Drive | 979.00 |
| 8250 Disk Drive | 1279.00 |
| 9060 Hard Disk | 1979.00 |
| 8023 Printer | 529.00 |
| 6400 Printer | 1399.00 |

### C-64 STUFF

| | |
|---|---|
| C-64 Computer | $219.00 |
| 1541 Disk Drive | 249.00 |
| 1701 Monitor | 249.00 |
| 1526 Printer | 339.00 |
| 1600 Modem | 69.00 |

Call Toll-Free by dialing:
Outside Texas:
**1-800-221-WORX**
Inside Texas:
**1-800-692-4265,**
wait for beep, then dial 008-3378,
wait for tone and dial 993.
or Lubbock 797-2623,
Ft. Worth: 817/589-2622
807 Melborne Hurst, Tx. 76053

**MICRO WORX**
**4210 D 50th 797-2623 Lubbock, TX 79413**

# Dynamic SAVE For VIC And 64

Stephen S. Leven

**These short programs, for the VIC or 64, can take the tedium out of frequently SAVEing and VERIFYing your BASIC programs. For disk and tape users.**

When you are typing in a long program, it's a good idea to SAVE portions of it frequently, and to make backup copies. But it is tedious to continually type SAVE "PROGRAM NAME", wait until the program is saved, retype SAVE "PROGRAM NAME", then wait again. "Dynamic SAVE" will do this work for you, whether you use tape or disk, using a technique known as the dynamic keyboard.

## Why You Should SAVE Periodically

A sudden loss of power—during a thunderstorm or when you accidentally knock the power cord from the wall socket—can instantly wipe out all your hours of hard work. Even something as simple as turning on the dishwasher or garbage disposal can cause a voltage drop that garbles the program in memory.

These four bits of advice will minimize the consequences of a sudden power failure or electrical glitch:

1. SAVE your program every 15 minutes or so, or whenever the changes you have made will cause you a serious setback if they are lost.

2. If you use tape, SAVE two copies of the program, one after the other, to reduce the chance of losing the program due to accidental damage to one section of the tape.

3. Use two tapes (or disks). For tape, first SAVE on one tape, then SAVE on a second. The third time you SAVE, use the first tape again. Continue to alternate tapes, so that if something happens during the SAVE, or your tape is damaged, you still have your previous version on the other tape. (Follow this same procedure for disk backup.)

4. Finally, when you have finished debugging your program and it runs perfectly, make one or two backup copies. It is a good idea to keep an archive tape (or disk) for emergencies. If your working copy of the program fails, you can retrieve your program from the archive tape (or disk).

## The Dynamic Keyboard Technique

The term *dynamic keyboard* basically means that you have your program display certain screen instructions which are executed after the program ends. You do this by inserting program lines which CLEAR the screen and PRINT the instructions on the screen just before the END line, and then load the keyboard buffer with the cursor controls and RETURNs necessary to execute those instructions.

When the program comes to the END instruction, it goes into immediate mode. The first thing it checks is the keyboard buffer, which contains your RETURNs and cursor controls. It then executes them just as if you were typing them in. As the cursor moves across the commands printed on the screen, they are automatically executed.

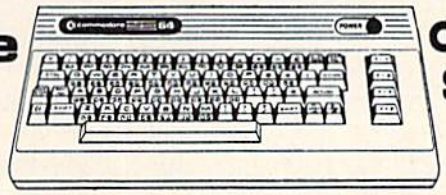
## Tape And Disk Versions

Program 1 is for tape users, and Program 2 for

# COMPUTER MAIL ORDER

commodore

## VIC-20 CALL



## COMMODORE 64. $199

| | |
|---|---|
| 1520 Color Printer/Plotter | $169.00 |
| M-801 Dot Matrix/Parallel | $219.00 |
| 1530 Datasette | $69.00 |
| 1541 Single Disk Drive | $249.00 |
| 1600 VIC Modem | $59.00 |
| 1610 VIC Term 40 | $49.00 |
| 1650 AD/AA Modem | $89.00 |
| 1702 14" Color Monitor | $249.00 |
| 1311 Joystick (each) | $4.99 |
| 1312 Paddles | $11.99 |
| 1110 VIC 8K | $42.00 |
| 1111 VIC 16K | $69.00 |
| 1011 RS-232 Interface | $42.00 |
| 1211 Super Expander | $53.00 |

### PRINTERS

| | |
|---|---|
| Epson (MX100, RX80, FX80, FX100) | CALL |
| Okidata (82, 83, 84, 92, 93) | CALL |
| Star Gemini 10X | $299.00 |
| Star Gemini Delta 10 | $559.00 |
| Smith Corona TP-2 | $439.00 |
| C.Itoh Prowriter 8510P | $379.00 |
| C.Itoh Gorilla | $209.00 |

### MSD

| | |
|---|---|
| SD-1 Disk Drive | $349.00 |

### CARDCO.

| | |
|---|---|
| Light Pen | $32.00 |
| 3 Slot VIC Expansion Interface | $32.00 |
| 6 Slot Expansion Interface | $79.00 |
| Casssette Interface | $29.00 |
| Parallel Printer Interface | $49.00 |
| Parallel Printer Interface w/Graphics | $69.00 |

## SOFTWARE

### commodore

| | |
|---|---|
| CBM 64 Reference Guide | $18.00 |

C-64 DISKS

| | |
|---|---|
| EasyCalc | $65.00 |
| EasyFinance I, II, III, IV | $19.00 |
| EasyMail | $19.00 |
| EasyScript | $39.00 |
| Word/Name Machine | $19.00 |
| EasySpell | $19.00 |
| Accounts Receivable | $39.00 |
| Accounts Payable | $39.00 |
| General Ledger | $39.00 |
| Assembler | $19.00 |
| Logo | $39.00 |
| Pilot | $39.00 |
| Pet Emulator | $19.00 |
| Screen Editor | $19.00 |
| Music Machine | $15.00 |
| Music Composer | $15.00 |

VIC 20 CARTRIDGES & DISKS

| | |
|---|---|
| Gortek & the Micro Chips(C) | $19.00 |
| Super Slot (R) | $14.00 |
| Super Alien (R) | $14.00 |
| Jupiter Lander (R) | $14.00 |
| Radar Rat Race (R) | $14.00 |
| Count Adventure (R) | $21.00 |
| Pinball Spectacular (R) | $19.00 |
| VIC Reference Guide | $15.00 |

### ARTWORX

C-64/VIC 20 CASSETTES

| | |
|---|---|
| Bridge 3.0 | $15.00 |
| Teacher's Pet | $12.00 |

### BRODERBUND

VIC 20 CASSETTES

| | |
|---|---|
| Martian Raid | $16.00 |
| Shark Trap | $16.00 |
| Multisound Synthesizer | $16.00 |

### COMMERCIAL DATA

VIC 20 CASSETTES

| | |
|---|---|
| Motor Mouse | $23.00 |
| Centipod | $23.00 |
| Frogee | $23.00 |

C-64 CASSETTES

| | |
|---|---|
| Road Toad | $24.00 |

### CREATIVE SOFTWARE

C-64 CASSETTES

| | |
|---|---|
| Home Inventory | $11.00 |
| Household Finance | $23.00 |

C-64 CARTRIDGES

| | |
|---|---|
| Trashman | $29.00 |

C-64 DISKS

| | |
|---|---|
| Home Inventory | $11.00 |
| Household Finance | $29.00 |

VIC 20 CASSETTES

| | |
|---|---|
| Home Inventory | $12.00 |
| Household Finance | $14.00 |

VIC 20 CARTRIDGES

| | |
|---|---|
| Astro Blitz | $21.00 |
| Black Hole | $32.00 |
| Trashman | $21.00 |
| Choplifter | $21.00 |

### DYNATECH

C-64 DISK

| | |
|---|---|
| Codewriter | $75.00 |

### EPYX

VIC 20 CASSETTES

| | |
|---|---|
| Ricochet | $32.00 |
| Sword of Fargoal | $24.00 |
| Rescue at Rigel | $24.00 |

### QUICK BROWN FOX

C-64/VIC 20 CARTRIDGE

| | |
|---|---|
| Word Processor | $49.00 |

### UMI

VIC 20 CARTRIDGES

| | |
|---|---|
| Amok | $30.00 |
| Meteor Run | $40.00 |
| Alien Blitz | $30.00 |

VIC 20 CASSETTES

| | |
|---|---|
| Cloud Burst | $15.00 |
| Video Verman | $19.00 |

### DUST COVERS

| | |
|---|---|
| C-64/VIC 20 Cover | $9.99 |

### ATARISOFT

C-64/VIC 20 Cartridges

| | |
|---|---|
| Centipede | $37.99 |
| PacMan | $37.99 |
| Donkey Kong | $37.99 |
| Dig Dug | $37.99 |
| Defender | $37.99 |
| Robotron | $37.99 |
| Stargate | $37.99 |

### HES

VIC 20 CARTRIDGES

| | |
|---|---|
| VIC Forth | $32.00 |
| HES Mon | $29.00 |
| HES Writer | $29.00 |
| Aggressor | $29.00 |
| Synthesound | $21.00 |
| Shamus | $29.00 |
| Protector | $29.00 |
| Turtle Graphics | $29.00 |

C-64 CARTRIDGES

| | |
|---|---|
| HES Mon | $29.00 |
| HES Writer | $35.00 |

### NUFEKOP

VIC 20 CASSETTES

| | |
|---|---|
| Alien Panic | $10.00 |
| Race Fun-Drag Race | $16.00 |
| The Catup | $10.00 |
| Exterminator | $19.00 |

C-64 CASSETTE

| | |
|---|---|
| 3-D Man | $16.00 |

### ROMOX

| | |
|---|---|
| Typo (20/64) | $29.00 |

### RAINBOW

C-64 DISKS

| | |
|---|---|
| Personal Finance | $48.00 |
| Writers Assistant | $95.00 |
| Spreadsheet Assistant | $95.00 |

### QUANTUM

| | |
|---|---|
| 40/80 Column Video Board | $95.00 |
| 40/80 Col. Video Board(16K) | $179.00 |

### SIRIUS

VIC 20 CARTRIDGES

| | |
|---|---|
| Deadly Duek | $21.00 |
| Spider City | $29.00 |

### MICROSPEC

VIC 20 CASSETTES

| | |
|---|---|
| Spelling Bee- Grades 2, 3, 4, 5, or 6 | $8.00 |
| Math Drill | $8.00 |
| Portfolio Manager | $16.00 |
| Data Manager | $16.00 |

VIC 20 DISKS

| | |
|---|---|
| General Ledger | $69.00 |
| Mailing List Manager | $35.00 |
| Inventory Package | $69.00 |
| Payroll | $69.00 |
| Data Base | $49.00 |

C-64 CASSETTES

| | |
|---|---|
| Black Box | $12.00 |
| Color Sketch | $20.00 |
| Match Maker | $16.00 |

C-64 DISKS

| | |
|---|---|
| Mailing List Manager | $45.00 |
| Inventory Package | $79.00 |
| General Ledger | $79.00 |
| Payroll | $79.00 |
| Data Base | $69.00 |
| CalcResult | $139.00 |
| Black Box | $16.00 |
| Color Sketch | $22.00 |
| Match Maker | $20.00 |

### TRONIX

VIC 20 CASSETTES

| | |
|---|---|
| Galactic Blitz | $19.95 |
| Swarm | $22.95 |
| Sidewinder | $22.95 |

### VICTORY

C-64/VIC 20 CASSETTES

| | |
|---|---|
| Adventure Pak I(3 games) | $12.00 |
| Adventure Pak II(3 games) | $12.00 |
| Annihilation | $16.00 |
| Grave Robber | $11.00 |
| Kongo Kong | $16.00 |
| Trek | $11.00 |

### PROFESSIONAL SOFTWARE

| | |
|---|---|
| Word Pro 64 | $59.00 |

**We carry a selection from the above manufacturers plus...**

**Synapse, Thorn, InfoCom, Onslow, Practicalc, Spinnaker & Timeworks.**

## =WEST= 1·800·648·3311

In NV call (702)588-5654, Dept. 0313
P.O. Box 6689, Stateline, NV 89449
Order Status #: 588-5654

## =CANADA= 1·800·268·4559

In Toronto call (416)828-0866, Dept. 0313
2505 Dunwin Ct.,Unit 1B,
Mississauga, Ontario, Canada LSLIT1
Order Status #: 828-0866

## =EAST= 1·800·233·8950

In PA call (717)327-9575, Dept. 0313
477 E. Third St. Williamsport, PA 17701
Order Status #: 327-9576
Customer Service Number: 327-1450

VISA MasterCard

No risk, no deposit on C.O.D. orders. Pre-paid orders receive free shipping within the UPS Continental United States with no waiting period for certified checks or money orders. Add 3% (minimum $5.00) shipping and handling on all C.O.D. and credit card orders. Larger shipments may require additional charges. NV and PA residents add sales tax. All items subject to availability and price change. We stock manufacturer's and third party software for most all computers on the market. Call today for our new catalog.

CANADIAN ORDERS: All prices are subject to shipping, tax and currency exchange fluctuations. Call for ex[illegible] orders in Canada.

disk users. It is a good idea to LOAD Dynamic SAVE before you start working on your program. You can change the line numbers if you wish, but, by using the line numbers I've used, you can easily remember that RUN 60000 will perform your SAVE.

Program 1 SAVEs two consecutive copies of your program, including Dynamic SAVE, to tape. When the SAVEs are completed, the screen will be set up to perform a VERIFY of each copy. Just rewind the tape and press RETURN.

Line 60010 defines the character for the quote mark, since using the CHR$ equivalent is the best way to PRINT it on the screen. This line also defines the name of the program to be SAVEd, which is stored in the variable N$. Change the contents of N$ to the name of the program you wish to save.

Line 60019 is a REM line, reminding you to use *either line 60020 or line 60021*, depending on whether you have a VIC or a 64. These two lines set the screen and border to their default colors, and define the character colors for use in line 60030. These colors are selected so that program operation on the screen is invisible. It's a good way to avoid screen clutter. If you want to see what the computer is doing, simply change the value of C1$ to that of C2$ in line 60020 or 60021.

Line 60030 changes the character color to that of the screen color by printing C1$. Then it clears the screen and displays the following message:

```
FORQ=1TO2:SAVE"DYNAMIC SAVE":NEXT
```

After printing, the character color is restored to normal by printing C2$.

Line 60040 loads the keyboard buffer with a HOME (ASCII 19) and RETURN (ASCII 13), just as though they had been typed in.

Line 60050 POKEs the following characters to the keyboard buffer: V, SHIFTed E, a colon, another V, and another SHIFTed E. (V-SHIFT-E is the Commodore abbreviation for the BASIC command VERIFY.)

The keyboard buffer (memory locations 631–640) can be loaded with up to ten characters. The first character to be executed should be POKEd into location 631, the second into location 632, and so on. (The character codes may be found in the appendices of *VIC-20 User's Manual*, *VIC-20 Programmer's Reference Guide*, *Commodore 64 User's Manual*, or *Commodore 64 Programmer's Reference Guide*.) Location 198 must be POKEd with the *number* of characters in the keyboard buffer, in this case seven. The END statement assures that the program will end at this point and pass control to the keyboard buffer.

Program 2 is similar to Program 1. The main differences are in line 60030 and in the characters POKEd into the keyboard buffer. In the disk version, line 60030 PRINTs, at the top of the screen, the disk command to SAVE and Replace the program, followed by a colon and the command to VERIFY the program on disk. Since the user does not need to take any action (such as rewinding a tape) in the disk version, verification can begin immediately after the SAVE. For that reason, the additional keyboard buffer POKEs in line 60050 of Program 1 are not needed in the disk version, so the program ENDs after POKEing a 2 in location 198 to indicate two characters in the keyboard buffer.

## How To Use The Program

To use this program for saving to tape:

1. Type or LOAD Program 1 into your VIC-20 or Commodore 64 before you start writing your program. Substitute your program name in place of DYNAMIC SAVE in line 60010.

2. When you're ready to SAVE your program, place your tape into the recorder and type RUN 60000. The screen will clear, then the message PRESS RECORD AND PLAY ON TAPE will appear.

3. Press RECORD and PLAY. The computer will supply its typical response, OK, followed by SAVING and whatever program name you supplied.

4. After the first copy of the program is saved, the response SAVING and the program name will be repeated to indicate that the second copy is being SAVEd. When the second SAVE is finished, the familiar READY message will be displayed, followed by the flashing cursor, positioned on the line with the double VERIFY command (V-:V-).

5. *Be sure to rewind your tape to the beginning of the program.* Then press RETURN. The computer will respond with PRESS PLAY ON TAPE. After you press PLAY, the normal VERIFY routine will take place: OK, SEARCHING, FOUND and your program name, VERIFYING, OK. The process then repeats for the second copy. If you choose not to VERIFY, use the cursor controls to move the cursor off the V-:V- line before pressing RETURN.

If you use disk, type or LOAD Program 2, making sure to substitute your program name in place of DYNAMIC SAVE in line 60010. When you're ready to SAVE, simply type RUN 60000. The computer will then display SAVING and VERIFYING messages at the appropriate time.

Once Dynamic SAVE is in place, you can face the possibility of a power failure with a little less dread.

*See program listings on page 165.*

# Let Your Computer "SPEAK"

## COMvoice IS AS EASY AS 1-2-3


1) PLUG COMvoice INTO YOUR VIC-20 OR CBM-64
2) TURN YOUR COMPUTER ON
3) TYPE SPEAK "HELLO, HOW ARE YOU"

SPEAK
SPEAK
SPEAK

## AS EASY TO USE AS A PRINT STATEMENT

**ONLY $149.95**

**DEALER INQUIRIES INVITED**

### ALSO ASK ABOUT OUR
## HOME SECURITY AND ENERGY MANAGEMENT PRODUCTS

**VIController**
Wireless remote control system for the VIC-20 and CBM-64. Use with BSR and Leviton remote receiver modules. $69.95

**COMsense**
Input device for the VIC-20 and CBM-64. Provides 4 open/close and 2 analog inputs. $49.95

**COMclock/AUTOboot**
Clock/calendar cartridge for CBM-64 with battery backup and auto-start software in ROM. $69.95

**GENESIS COMPUTER CORPORATION** P.O. Box 1143 Bethlehem, PA 18018 (215) 861-0850

---

## COLOR PROBLEMS?
### One of Our Four New Products will Solve Them!

You're not alone. Thousands of Commodore 64 owners have "fuzzy" color on their TV. Most have interference lines crowding out their great graphics. Many have bought expensive monitors or new TVs, and often even that hasn't helped. But, most of us just lived with the problem. Now the engineers at Bytes & Pieces have four simple, inexpensive solutions.

If you have an "old 64" (with the 5 pin Monitor Din Plug), you've probably had color, resolution and interference problems. We can solve them!

**1.** **The Interference Stopper**...A new kit that installs in minutes with two simple solder connections. Best results when combined with #2, 3, or 4 below. Absolutely stops 90% of the RF interference on your screen. **$15.95**

**2.** **The Color Sharpener**...Use if your "old 64" is hooked up to a TV. Just plug into the monitor plug, and the color and contrast immediately improve. Dramatically. Crisp letters. Great graphics. **$18.95**

**3.** **The NEW Color Sharpener CABLE**...Use if your "old 64" is hooked up to a monitor. A new 2 prong cable, with the Color Sharpener built in. All the benefits of #2, on your monitor. **$24.95**

**4.** **The Monitor "Improver"**...If you have a Commodore 1701 monitor, this cable (3 prong) gives you a picture you won't believe. Better than the cable Commodore built...by a lot. Try it, you won't be disappointed. (Also hooks your "Old 64" to the 1702.) **$24.95**

**5.** **The Reset Switch**...Here it is, a Reset Switch for Vic or Commodore. Get back into control of a "Hung-Up" program. Resets all pointers. Easy two solder connection installation. Every computer should have one. **$9.95**

If any of our products do not work to your satisfaction, send it back and we'll refund your purchase price in full.

## DUST PROBLEMS?
### Solve Them with Matching Dust Covers for Computer, Tape and Disk.
### $7.95—$9.95

These are the deluxe covers for either the Commodore 64 or the Vic 20 made of brown leather grain Naugahyde, specially lined with a soft non-scratch liner, for a cover you just can't beat.

Don't waste your money on those cheap looking, clear plastic, static filled covers. Get the quality ones, custom fitted to your Commodore computers.

Available singly or as a matched set in beautiful brown simulated leather.

Commodore 64 and Vic 20 are registered trademarks of Commodore Computer Company.

**ORDER TODAY!**
Please send me the following:

| Qty. | Item | Amount |
|---|---|---|
| ____ | Interference Stopper @$15.95 | $ ____ |
| ____ | Color Sharpener @$18.95 | $ ____ |
| ____ | NEW Color Sharpener Cable @ $24.95 | $ ____ |
| ____ | The Monitor Improver @ $24.95 | $ ____ |
| ____ | The Reset Switch @$9.95 | $ ____ |
| ____ | Computer Dust Cover @ $9.95 | $ ____ |
| ____ | 1541 Disk Dust Cover @ $8.95 | $ ____ |
| ____ | Dataset Dust Cover @ $7.95 | $ ____ |
| | Shipping & Handling | $ 2.00 |
| | 5% State Tax (Wisconsin Residents only) | $ ____ |
| | TOTAL | $ ____ |

☐ Check or Money Order enclosed
☐ Charge to my VISA or MasterCard
VISA # _____
MasterCard # _____
Inner Bank # _____
Expiration Date _____

Signature

SHIP TO:
Name _____
Address _____
City _____
State/Zip _____

**Bytes & Pieces, Inc.** Dealer Inquiries Invited. 550 N. 68th Street, Wauwatosa, WI 53213 414/257-3562

Dan Carmichael, Submissions Editor

# The Indexer

**This month's offering presents a small data base program, "The Indexer." Designed as an index for COMPUTE!'s GAZETTE articles, it can be used for a variety of purposes. It runs on any size VIC-20 and the Commodore 64.**

If you're like me, you probably keep your back issues of COMPUTE!'s GAZETTE. There's a wealth of reference material in each issue. The only problem is remembering just what issue contains that article you so desperately need.

This month, we'll look at a small data base program that allows you to keep an index of articles or books that are of interest to you. "The Indexer" stores such information as magazine (or book) name, subject matter, article title, month and year of issue, page number, and the type of computer the article applies to. It can also search for that article by subject, article name, magazine name, and type of computer.

## Storing Data In The Program

The Indexer is *machine independent*—it does not store data on a peripheral device such as a tape cassette or disk drive. Information is read into the program from DATA statements and is stored within the program in an array. If you study it carefully, you'll see some useful array and table look-up techniques.

Each DATA statement you enter must include the following six elements in order, and each entry should be separated by a comma.

DATA *magazine name, article title, subject, month.year, page number, type of computer*

Be careful when entering the DATA statements—a misplaced or forgotten comma will cause errors when the program is run. Be sure not to use commas or colons when typing in the article titles.

## How To Use The Indexer

Type in the program, carefully watching all cursor control characters, and save it to tape or disk before running. The five DATA statements at the end of the program are optional, included only as examples of the DATA statement format. If you wish to begin your own data base, you can replace the DATA statements from line 901 on.

Each time you add or delete DATA statements from the program, change the value of the variable N in line 900. This variable represents the exact number of DATA statements included. If you number consecutively, beginning at line 901, it will be easy to figure out how many DATA statements there are. And anytime you update your program, you should SAVE a copy to tape or disk.

Once the program is running, you'll be prompted to select the target of your search. You can search for article subject, article name, name of magazine, or type of computer. To start the search, press the indicated function key. You'll then be asked for the target of your search. Just enter the search keyword, press RETURN, and the program will perform the table search.

If you're using the program with an unexpanded VIC, memory will become a problem as you add DATA statements. String arrays—the kind used in this program to store data—use a lot of memory. In addition, the DATA statements take up six bytes plus one byte per character. If you accumulate a lot of data, an expander cartridge will come in handy. The Indexer is written to run on a VIC with any memory configuration, as well as on the Commodore 64.

## Tips For Data Entry

*Subject:* Your searches will usually be done by article subject, so keep this category as broad as possible. For example, let's say you want to index various articles about game paddles. Enter all of them with the subject "paddles," even if some are about drawing with paddles and others about using them in games. That way, when you enter "paddles" as the target of your search, the index of *all* articles on this subject will be displayed.

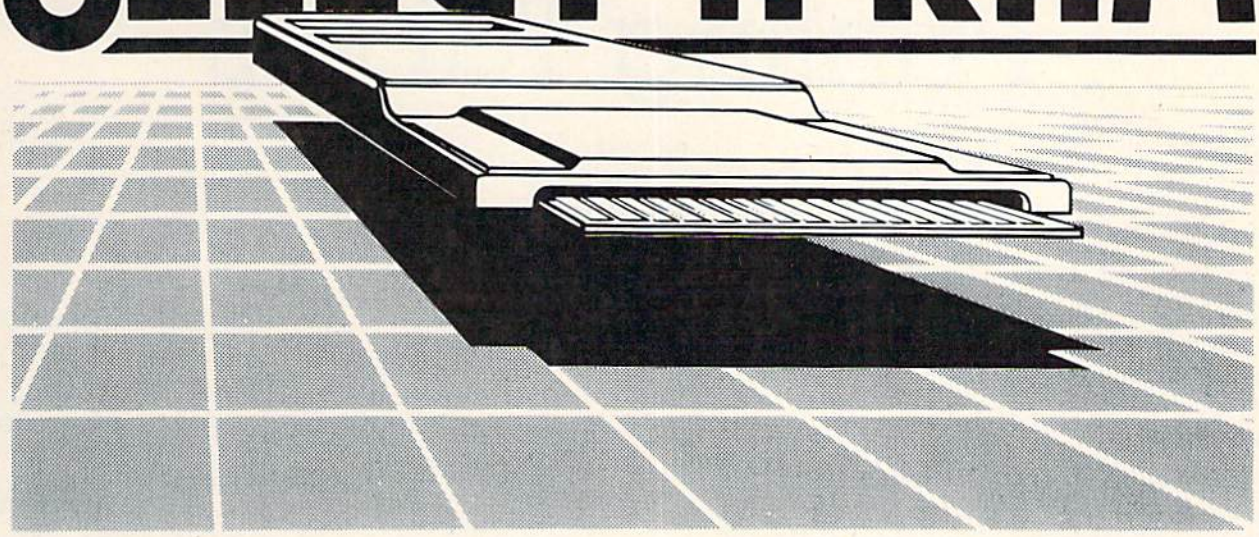
*Spelling:* Watch your spelling, and be consistent with your subject category names. For example, don't enter one subject as "paddle" and another as "paddles." The computer will see these as two completely different categories.

*Memory:* As stated before, The Indexer can use a lot of memory, so you might want to abbreviate article titles. For example, the GAZETTE column "Machine Language For Beginners" could be entered as "ML For Beginners" or even "Beg ML."

Although this program was written as an article index, it can be adapted for other uses. The data base has six elements and can search by any of four variables. It could be easily adapted for other uses such as a birthday reminder or an electronic phone book. The applications are up to you.

*See program listing on page 151.*

# 64K for VIC 20
# SELECT·A·RAM



SELECT·A·RAM BRINGS TO YOUR VIC 20 THE POWER THAT ONLY MEMORY CAN PROVIDE.

The power of any computer is measured by its memory capacity. The more memory you have, the more powerful your programs can become.

SELECT·A·RAM gives your VIC 20 the power of memory. 65,536 bytes of power to be exact. Enough programming power to rival any Home computer.

The power hungry programmer can also add more memory. Each of SELECT·A·RAM's two expansion slots will accept any amount of memory from 3K to 128K.

SELECT·A·RAM's powerful expansion capabilities are made possible by a technique we call Soft Select. Soft Select allows your VIC 20 to perform many sophisticated functions not possible with other memory expansion devices, ie., disk drive emulation, printer spooling, simultaneous and interactive program execution (to name just a few of our soon to be released packages).

Bring the power of memory to your VIC 20 . . . . . with SELECT·A·RAM.

Call or write for additional information and the dealer nearest you. Direct orders accepted.

DEALERS INQUIRIES INVITED

Vic 20 is a Trade Mark of Commodore Electronics Limited.

## SELECT·A·RAM

- 64K Memory
- Two Expansion Slots
- Write Protection
- Reset Swjtch
- Expandable with 64K and 128K Modules
- Soft Select Control
- Compatible with All Program Cartridges and Hardware Devices
- Provision for Optional External Power
- One Year Limited Warranty

Distributed in Canada by:

**PAX SOFTWARE**

60 Hanson Road, Unit 133 Mississauga, Ontario L5B 2P6
(416) 270-2639

**ADVANCED·PROCESSOR·SYSTEMS**

P.O. Box 43006, Austin, TX., 78745-0001, (512) 282-8222

# Printing Tables

Pat Slater

*If you've discovered a clever, time-saving technique, or a brief but effective programming shortcut, send it to "Hints & Tips," c/o* COMPUTE!'s GAZETTE. *If we use it, we'll pay you $35.*

One of the limitations of Commodore BASIC is the total lack of formatted PRINT statements. If you're used to PRINT with format or PRINT USING, it may seem impossible to print neatly aligned tables with Commodore BASIC.

The language does provide TAB and SPC functions for screen displays, but both simply space over when used to print to a printer or a file with PRINT#. The programs I've included will format for screen or printer, depending on your input. They're especially helpful for use with formatting printed output. Let's look at an example:

```
PRINT A$;TAB(20);B$
```

prints B$ starting in column 20 regardless of the size of A$, but

```
PRINT#4,A$;TAB(20);B$
```

prints A$, skips 20 spaces from the end of A$, then prints B$. There's no telling where B$ will end up unless you know the size of A$. This being the case, you must find a way to count spaces when doing formatted printout.

One way to count spaces is to use the LEN function. For example, the statement below will place A$ in column 1, B$ in column 15, and C$ in column 30:

```
PRINT#4,A$;TAB(14-LEN(A$));B$;TAB(14-LEN
  (B$));C$
```

The following program uses the LEN function to align rows of names:

```
3 PRINT"{CLR}SCREEN OR PRINTER S OR P":IN
  PUT I$                              :rem 151
4 IF I$<>"S"AND I$<>"P"THEN3          :rem 241
5 IFI$="S"THENJ=3:GOTO8               :rem 153
6 J=4                                 :rem 241
8 OPEN4,J                             :rem 20
10 FOR I=1 TO 5                       :rem 215
20 READ A$,B$:IFI$="S"THEN35          :rem 22
30 PRINT#4,A$;TAB(10-LEN(A$));B$:GOTO40
                                      :rem 214
35 PRINT#4,A$;SPC(10-LEN(A$));B$      :rem 19
40 NEXT I                             :rem 236
50 DATA WATTS, SORENSON, MATTHEWS, NG, JO
   NES                                :rem 186
60 DATA BURL, YATES, RUVALDS, KING, REDLA
   SIK                                :rem 132
```

When executed, the program prints:

```
WATTS       SORENSON
MATTHEWS    NG
JONES       BURL
YATES       RUVALDS
KING        REDLASIK
```

If you use this method to align numbers (along with STR$ to convert the numbers to strings), several problems pop up as shown in the next example:

```
3 PRINT"{CLR}PRINTER OR SCREEN P OR S":IN
  PUTI$                               :rem 151
4 IF I$<>"P"ANDI$<>"S"THEN 3          :rem 241
5 IF I$="S"THENK=3:GOTO8              :rem 154
6 K=4                                 :rem 242
8 OPEN4,K                             :rem 21
10 FOR N=1 TO 5                       :rem 220
20 READ I,J:IFI$="S"THEN35            :rem 222
30 PRINT#4,I;TAB(8-LEN(STR$(I)));J:GOTO40
                                      :rem 199
35 PRINT#4,I;SPC(8-LEN(STR$(I)));J :rem 4
40 NEXT N                             :rem 241
50 DATA 78.66, 40.00, 139.30, -77.22, 200
   0.00                               :rem 213
60 DATA -142.91, 6.56, 12.50, 521.12, 9.9
   9                                  :rem 83
```

This program prints:

```
 78.66        40
 139.3       -77.22
 2000        -142.91
 6.56         12.5
 521.12       9.99
```

Notice that the numbers are aligned on the left (at the sign position) rather than by decimal point, and that trailing zeros are lost after the decimal point. To make the individual numbers appear in the correct format (40 as 40.00, for example) you can convert them to strings as follows:

```
I$=STR$(INT(I))+"."+RIGHT$(STR$(I*100),2)
```

The first term gets the integer part of the number, next the decimal is added, and finally it is necessary to multiply by 100 and grab the last two digits to keep from losing trailing zeros.

Once you have the number in correct format, use the LEN function to count spaces and tab *before* printing each number as shown in the example below:

```
3 PRINT"{CLR}SCREEN OR PRINTER S OR P":IN
  PUT U$                              :rem 163
4 IF U$<>"S"ANDU$<>"P"THEN3           :rem 9
5 IF U$="S"THENK=3:GOTO8              :rem 166
6 K=4                                 :rem 242
8 OPEN4,K                             :rem 21
10 FOR N=1 TO 5                       :rem 220
20 READ I,J                           :rem 61
22 I$=STR$(INT(I))+"."+RIGHT$(STR$(I*100)
  ,2)                                 :rem 140
24 J$=STR$(INT(J))+"."+RIGHT$(STR$(J*100)
  ,2)                                 :rem 145
30 PRINT#4,""TAB(8-LEN(I$));I$;TAB(17-LEN
  (J$));J$                            :rem 200
40 NEXT N                             :rem 241
50 DATA 78.66, 40.00, 139.30, -77.22, 200
  0.00                                :rem 213
60 DATA -142.91, 6.56, 12.50, 521.12, 9.9
  9                                   :rem 83
```

Now you get the printed numbers aligned by decimal position:

| | |
|---|---|
| 78.66 | 40.00 |
| 139.30 | −78.22 |
| 2000.00 | −143.91 |
| 6.56 | 12.50 |
| 521.12 | 9.99 |

Using LEN to count spaces this way allows you to print neat-looking tables of words and/or numbers. Though a lot of functions are involved, it doesn't take nearly as much time as the printout process, so it won't slow down your program.

# HORIZONS: 64

CHARLES BRANNON
PROGRAM EDITOR

Apple recently reduced the price of its Lisa to around $8000. Lisa, whose acronym supposedly stands for Local Integrated Software Architecture, was actually just the in-house code name for the machine (insiders claim it was named after a girlfriend of Steve Wozniak, Apple co-founder). The name of the machine was leaked so extensively that Apple was forced to market with the code name, hence the apocryphal acronym.

Lisa is a dedicated workstation with a mouse, ultra-high resolution graphics, and icons (pictorial diagrams). "Dedicated workstation" means that there is one user per computer, rather than many users sharing a large computer via separate terminals. The advantage of a dedicated workstation is its exclusivity, privacy, and speed of access. The entire power of the computer can be dedicated to one user, rather than spread out among many. This is the primary concept behind so-called personal computers.

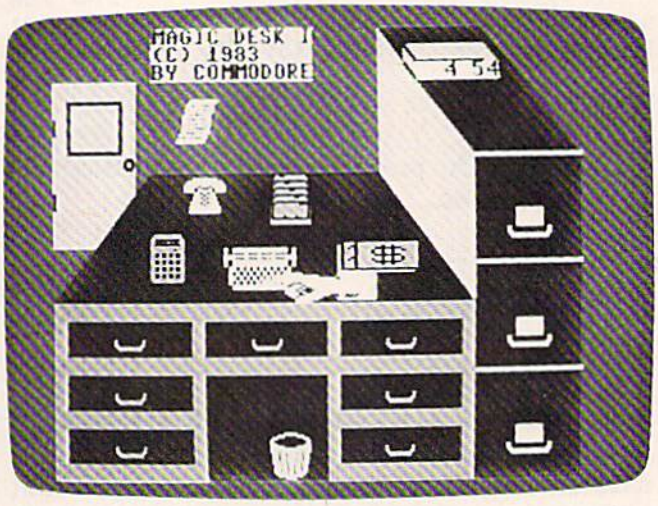
The disadvantage of dedicated systems surfaces in environments where people need to share and exchange information. That is why Local Area Networks (LANs) are hooking up these small computers, usually to a central hard disk. A LAN doesn't violate the concept of personal computers; it just broadens their communication capabilities. Some companies are going too far, though, and we are seeing expensive business microcomputers which are mediocre mimics of the large, powerful, but impersonal minicomputers and mainframes.

Apple claims you can start using Lisa within 20 minutes, without any instruction manuals. If you remember your first experience with a computer, you may be inclined to doubt this. Yet there's something new here. Apple has drawn on the experience of research at Xerox's Palo Alto Research Center (PARC). The Lisa is an intuitively understandable system.

Most of this "user-friendliness" (to use a term that has become banal in the industry) comes from Lisa's linked menus. The menus lead you from one option to another and are primarily pictorial, drawing on the Lisa's very high resolution capability (pun somewhat intended).

A highly visible characteristic of the Lisa is the mouse, used to make selections as you move the cursor about. The idea is the same as a joystick or trackball (some mouses are merely upside-down trackballs). Using the mouse is supposed to be more natural than pressing cursor control keys. Many people complain that the mouse is a gimmick, taking up extra desk space and requiring you to take your hands away from the keyboard to move the cursor.

Both hardware and software companies were impressed by and envious of the Lisa. Lisa was really a new product, an innovation in an industry of "me, too" computers and software. Well, the copycats didn't waste any time. Companies began developing their own integrated software, complete with icons, windows, and even mouses. The effort is Promethean, as companies try to bring the $8000 capability of the Lisa to your $2000 micro (of course, they charge you only $800 to $1000 for the software).




*Preparing to type with Magic Desk I.*

## Commodore's Reaction: Magic Desk I

Commodore also caught icon-fever. At the January 1983 Consumer Electronics Show (CES), they introduced Magic Desk I, a software package for the 64 which was not available until around November. The price is down-to-earth, with a suggested retail of $60–$70. What Commodore and others (Microsoft [Windows], VisiCorp [Visi On], Quarterdesk Software [DesQ]) seem to forget is that Apple spent the equivalent of a hundred man-years developing Lisa. Some integrated pro-

THE MASTER KEY

unlocks the door to the Commodore 64




Word Processing
File Management
Home Finance
Electronic Paintbrush
Terrestrial Game

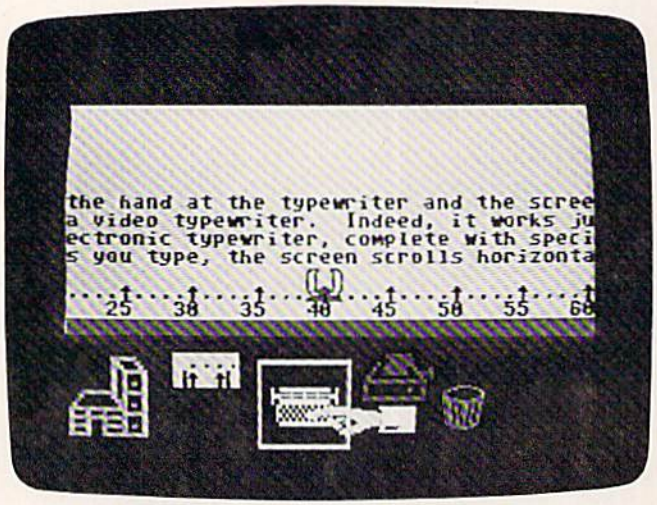
The Gateway to Five Worlds 129.95

TRI MICRO

International Tri Micro
1010 N. Batavia Ste. G
Orange, CA 92667
714-771-4038

grams may be capable, others might just copy Lisa's gimmicks.

Magic Desk may be right for some people, but it's not for everyone. When you plug in the Magic Desk cartridge and turn it on, a picture of a desk appears. On top of the desk are several icons: pictures of a typewriter, telephone, calculator, card file, and ledger. Under the desk is a wastebasket, and to the right is a filing cabinet. A digital clock on top of the filing cabinet keeps time. A door to the left "opens" the way to new applications. Above this scene hovers a ghostly hand, your cursor.




*Creating text in the typewriter mode.*

With the joystick, you move the hand to the appropriate picture and press the fire button to select it. Magic Desk I supports only a few of the icons: the typewriter, wastebasket, filing cabinet, and digital clock. Magic Desk II will support the ledger (spreadsheet) and the calculator. Presumably, the phone will be used with a modem, and the card-file as a simple data base manager.

Point the hand at the typewriter and the screen transforms into a video typewriter. Indeed, it works just like a modern electronic typewriter, complete with special effects. As you type, the screen scrolls horizontally. The margins are always visible, and you can change them at any time. With the function keys you can set, clear, and move to tab stops. When you press RETURN, the paper pops up a line and the typeball whirs to the beginning of the next line, complete with sound effects. This is perhaps a bit too cute.

After you type your letter, you can return to the desk and file the letter. With the hand, you can open one of the three file cabinets. Each cabinet holds ten file folders, each of which can hold ten pages of text. What you are really doing is loading and saving your files to disk, but this is supposedly transparent. You just select a folder, name it if you like, put your typing into it, or retrieve a page from it. Any of these operations will cause a picture of the disk drive to appear, and the actual drive will whir and click a surprisingly long time, considering that you are only trying to file one page. I found this quite frustrating.

This is also where Magic Desk gets confusing. None of the icons are labeled, so it's like reading those pictorial traffic signs in a foreign country. If you get stuck, you can press the Commodore logo key and a help screen will appear. The icons you are using will be named; you can select any of them with the hand to get a more complete description.

I found Magic Desk novel and rather easy to use, though the pictures are both a blessing and a curse (I would prefer an English menu of choices). What you gain in ease of use and intuitive operation, you lose in power, speed, and efficiency. Compared with the "old way" of doing things, Magic Desk is limited once you're no longer a beginner. It slows you down as you pursue a task through a chain of menus. The video typewriter has almost none of the advantages of a true word processor. It's just like an IBM Selectric. You don't have to learn anything, but you don't have a tenth of the flexibility and power of even a mild-mannered word processor. Apparently, this is not a problem in Commodore's eyes. Indeed, the press release sells this similarity, emphasizing a typewriter is not a word processor, and implies you're better off that way.

Nonetheless, Magic Desk can be an excellent nonintimidating way to learn to use a computer. Magic Desk may change the mind of anyone who "hates computers." It lets you perform basic computer tasks with a minimum of fuss, and can lead you into more powerful applications later. Despite its limitations, Magic Desk points the way to the future of home computer software.

Magic Desk I
*Commodore Business Machines*
*1200 Wilson Drive*
*West Chester, PA 19380* *$60–$70*

# Commodore 64 and VIC-20

# SuperTerm

## $149.95

## Telecommunications with a difference!

Unexcelled communications power and compatibility, especially for professionals and serious computer users. Look us over; **SuperTerm** isn't just "another" terminal program. Like our famous Terminal-40, **it's the one others will be judged by.**

- **EMULATION**—Most popular terminal protocols: cursor addressing, clear, home, etc.
- **EDITING**—Full-screen editing of Receive Buffer
- **UP/DOWNLOAD FORMATS**—CBM, Xon-Xoff, ACK-NAK, CompuServe, etc.
- **FLEXIBILITY**—Select baud, duplex, parity, stopbits, etc. Even work off-line, then upload to system!
- **DISPLAY MODES**—40 column; 80/132 with side-scrolling
- **FUNCTION KEYS**—8 standard, 52 user-defined
- **BUFFERS**—Receive, Transmit, Program, and Screen
- **PRINTING**—Continuous printing with Smart ASCII interface and parallel printer; buffered printing otherwise
- **DISK SUPPORT**—Directory, Copy, Rename, Scratch

Options are selected by menus and EXEC file. Software on disk with special cartridge module. **Compatible with CBM and HES Automodems**; select ORIG/ANS mode, manual or autodial.

**Write for the full story on SuperTerm; or, if you already want that difference, order today!**

Requires: Commodore 64 or VIC-20, disk drive or Datasette, and compatible modem. VIC version requires 16K memory expansion. Please specify VIC or 64 when ordering.

## Smart ASCII Plus . . . $59.95

**The only interface which supports streaming** —sending characters simultaneously to the screen and printer — with SuperTerm.

Also great for use with your own programs or most application programs, i.e., word processors. **Print modes:** CBM Graphics (w/many dot-addr printers), TRANSLATE, DaisyTRANSLATE, CBM/True ASCII, and PIPELINE.

Complete with printer cable and manual. On disk or cassette.

VIC 20 and Commodore 64 are trademarks of Commodore Electronics, Ltd.

**(816) 333-7200**

**Send for a free brochure.**

**MIDWEST MICRO inc.**

**MAIL ORDER:** Add $1.50 shipping and handling ($3.50 for C.O.D.); VISA/Mastercard accepted (card# and exp. date). MO residents add 5.625% sales tax. Foreign orders payable U.S.$, U.S. Bank ONLY; add $5 shp/hndlg.

311 WEST 72nd ST. • KANSAS CITY • MO • 64114

# COMMODORE 64

## American Peripherals

NEW ARRIVALS
(disk or tape)

1030 LANGUAGE TRANSLATOR
English to German. $10
584 PIANO-64 $19.95
Change your 64 to a piano.
126 TRAFFIC SIGNALS $14.95
Teach your child safety.
1240 POGO $24.95
Like Q-Bert.
596 TELLING TIME $24.95
Hi-res Round clock.
1249 TYPESETTER $49.50
Hi-res printing on 1525.
1276 ELIZA $19.95
Has conversation with you.
1016 ELECTRONIC GRADEBOOK
Teacher special $24.95
574 LOCATION OF COUNTRIES
Geography $14.95
575 CANCER $15.95
Kids and smoking.
762 ELEMENTARY ELECTRICITY
5th-9th gr. $24.95
1264 TRAVEL ABOUT AMERICA
Series of 7 programs on
Geog. and History $150.
578 SOUND EFFECTS $15.95
Demo and prog. aid.
139 STRATHCLYDE BASIC
12 lessons + test. $34.95
1014 GEOMETRY SERIES
20 progs. with hi-res
Disk $300 Tape $400
128 TEST MAKER $24.95
Makes multi-choice tests.
1018 LARGE NUMBER ADDITION
Great display $24.95
1275 VIC-20 EMULATOR $34.95
Run most VICs on 64.
1246 INTERRUPT MUSIC EDITOR
Machine code. $29.95
1017 MORTGAGE $19.95
Does all loan calculations.
1015 ANIMAL-VEGETABLE
6yr.-10yr. $14.95
1285 MAESTRO $34.95
All conceivable music and
sound functions in 1 program.
1280 WATTS and AMPS $14.95
Household elec. calcs.
580 MATCHING SHAPES
I.Q. Builder $14.95

EDUCATIONAL
(disk or tape)

644 Type Tutor $19.95
645 Assembly Language
Tutor $14.95
687 Fractional Parts $14.95
902 Estimating Fractions $14.95
695 Tutor Math $14.95
870 Square Root Trainer $14.95
699 Counting Shapes $14.95
694 Money Addition $14.95
689 Math Dice 14.95
678 Speed Read $14.95
643 Maps and Capitals $19.95
645 Sprite Editor $19.95
904 Sound Synthesizer
Tutor 19.95
696 Diagramming
Sentences $14.95
690 More/Less $14.95
688 Batting Averages $14.95
802 TicTac Math $16.95
904 Balancing Equations $14.95
905 Missing Letter $14.95
864 Gradebook $15.00
810 French 1-4 $80.00
811 Spanish 1-4 $80.00
807 English Invaders $16.95
809 Munchword $16.95
812 Puss in Boot $20.00
813 Word Factory $20.00
660 Hang-Spell $14.95
905 Division Drill 14.95
906 Multiplic. Drill $14.95
907 Addition Drill $14.95
908 Subtraction Drill $14.95
910 Simon Says 14.95
911 Adding Fractions $14.95
912 Punctuation $14.95

EDUCATIONAL

Series on disk

Computer Science (30 pr.) $350
HS Biology (70 programs) $500
HS Chemistry (40 programs) $450
HS Physics (60 programs) $475
HS SAT Drill (60 programs) $99
Elem. Social Studies (18 pr.) $225
Elem. Science (18 programs) $225
Further French (12 pr.) $95
Further Spanish (12 pr.) $95
Statistics (12 programs) $95

ORDERING BLANK

To: American Peripherals
122 Bangor Street 516-226-5849
Lindenhurst, NY 11757

Ship to: Name
Street
Town, State, ZIP

Please send your complete 64K catalog. Over 600 programs.

| ITEM | DESCRIPTION (specify disk or tape) | PRICE |
|---|---|---|
| | | |
| | | |
| | | |
| | | |
| | | |

Charge to my credit card .... VISA .... MASTER CARD
Account # .....
Exp. date .....
Signature .....

NY State Residents 7.25% tax $.....
COD add $2.00 $.....
foreign orders add $2.00 $.....
Shipping $..1.75...
Total Amount $.....

Commodore 64 is a trademark of Commodore Business Machines, Inc.

# SIMPLE ANSWERS TO COMMON QUESTIONS

TOM R. HALFHILL, EDITOR,
COMPUTE!'s PC & PCjr Magazine

# Q A

*Each month,* COMPUTE!'s GAZETTE *will tackle some questions commonly asked by new VIC-20/Commodore 64 users and by people shopping for their first home computer.*

**Q.** ***I've heard about emulators available for the Commodore 64 that allow it to run all Apple software and software for other computers, too. I've seen some magazine advertisements by mail-order companies for emulators. Do you plan to review any of these emulators?***

**A.** We haven't reviewed any emulators because, at this writing, they simply don't exist. Furthermore, we urge readers to be cautious about emulators—a lot of misinformation is circulating.

We, too, have heard all the stories about Apple emulators for the Commodore 64, and have seen the ads in magazines. Usually we telephone the company advertising the emulator and try to obtain one for testing and review purposes. Almost always the response is something like, "They'll be ready for delivery within a few weeks." Then several months go by, and still no emulator. Some companies have been promising to deliver emulators for more than a year. We have yet to see one.

Heed this advice: *Do not order or buy an emulator unless you first see an actual, working model.* Otherwise, you will probably be disappointed.

Why are we so emphatic? Because true emulation of another computer is not something that is easily accomplished. There are hundreds of problems to be overcome, especially when designing something like an Apple emulator for the Commodore 64. Yet, some companies persist in promising—and even advertising—Apple emulators.

Nearly all these emulator rumors can be indirectly traced to a statement made by Commodore back in early 1982. At that time, Commodore hinted that it intended to introduce a personal computer that could emulate other popular computers on the market, such as the Apple and Atari. A widely read article to this effect was published in the *Wall Street Journal*. Everyone was excited by the prospect, but eventually Commodore quietly shelved the idea—probably because it was too hard to accomplish at an affordable price.

In the meantime, the Commodore 64 was introduced. Unfortunately, some people jumped to the conclusion that this was the "chameleon computer" Commodore had hinted about. Within a few months, independent companies began announcing and even advertising Apple emulators for the 64. The idea of an Apple emulator was very attractive to new 64 owners because almost no software was available for the computer. But there are formidable obstacles to making an Apple emulator for the 64.

It's true that the computers are superficially similar—both are 40-column color computers with 64K Random Access Memory (RAM), Microsoft BASIC, and compatible microprocessor chips (the 6502/6510). But the memory maps are quite different, and the Apple lacks many of the 64's special features (sprites, 16-color high-resolution graphics, polyphonic synthesized sound, etc.). This is important because much commercial software these days is written entirely in machine language, including almost all games. ML programs are specific to the memory configuration of the machine on which they were written. Even seemingly slight changes to the native computer's operating system or memory layout (such as adding expanders to a VIC-20) can render some ML programs unusable. The problems of emulating a totally foreign computer are infinitely greater.

Even if an Apple emulator were developed, there would be one more problem. Almost all Apple software is published on floppy disks. The Commodore Disk Operating System (DOS) is incompatible with the Apple DOS. And the data is stored in different formats. An Apple emulator would do you no good unless you also had an Apple disk drive emulator. This would almost certainly require that you buy a separate disk drive and compatible interface cables.

In the meantime, let the buyer beware. We heard of one person who paid more than $100 for an "Apple emulator" for his 64 and got nothing more than a cable which allowed BASIC programs to be transferred between the computers. The task of translating the programs so they would run on the 64 was left up to him, of course.


**PROGRAMS DESCRIPTION**

# ENTECH

## LEADER IN SOFTWARE FOR THE COMMODORE 64

### BUSINESS

Integrated

**MANAGEMENT SYSTEM 64** — This integrated business program gives you the computer power once reserved for large corporations. It stores the names of items, prices, item numbers and does all calculations including tax, shipping costs and discounts. It prints invoices and packing slips in addition to short-term and long-term reports. Management System 64 even prints mailing labels from customer files. Know what's selling, whose buying and forget costly, time consuming inventory close downs. $69.95

Finance

**FINANCE CALC 64** — The leader in business and home financial analysis. You can have up to 1440 itemized expenses and print 1085 different financial reports and bar graphs as it stores and compares up to 12 different budgets at once. This powerful program is a must for the rising entrepreneur. $49.95

File

**DATA BASE 64** — A perfect record system for any business or home. It can store up to 1200 records for each file. Each record has a capability of 20 fields of information. The program has multi-level sorting which facilitates quick and easy recalling of information and printing of labels and reports. It even allows you to create personal print formats. Data Base 64 also can be used with popular word processing programs. $59.95

### HOME

3 in 1

**FAMILY PAC** — Three of the finest home programs available: $59.95

- **CHECKBOOK SYSTEM 64** — This easy to use program makes the balancing of confusing checkbook finances a breeze. It handles over 1300 transactions of up to 40 expense categories. Checkbook System 64 also prints statements, reports and all types of checks. Pay your bills by computer! Also available as a separate program. $39.95

- **RECIPE KEEPER 64** — This program is a handy kitchen aid that does more that just file. It can print out ingredients and directions, as well as calculate ingredient amounts for different serving sizes. The program comes complete with a mini-word processing section that enables you to type in up to 14 lines of directions. $39.95

- **SPACE MATH 64** — This educational program makes addition, substraction, multiplication and division learning fun and entertaining for youngsters. Solve math problems, explore the universe, dance to the music and watch the show. $29.95

### ENTERTAINMENT

Music

**STUDIO 64** — Roll over Beethoven, now anyone can create music as expressive and sophisticated as the most advanced programmers!! With Studio 64, the computer does all the work while you play and enjoy. It is the only music program available that allows you to create music without any programming knowledge. Just play and the computer will instantly write the music on the screen. Enter up to 3 voices and choose from 4 wave forms on any voice. Play your compositions at different speeds and filter settings with synthesized drum rhythms! It will save and recall, add music to your own programs and print lead sheets. Music is only as far away as your C64. Disk/Cass. $39.95

Art

**GAME DESIGNER 64** — This package contains everything you need to design colorful game characters, backgrounds and entire screens. Animate up to 16 sprites on the background of your choice. Add it to your own program. The limits are your own creativity. Combine it with Studio 64 to create any game imaginable. Power game sub-routine included. Disk/Cass. $39.95

— All programs come in disk unless otherwise indicated. —

**WHEN THE DUST SETTLES THE QUALITY REMAINS**

ENTECH®

8224 Sunland Boulevard • Sun Valley, California 91352 • Phone (213) 768-6646

# A Guide To Commodore User Groups

## Part 1

Kathy Yakal, Editorial Assistant

Here is an updated list of Commodore user groups throughout the world. If you have a group that you would like listed here, or if your listing needs to be changed or deleted, please contact us so that we may keep our records current.

In most cases, contact people have chosen to list their home addresses and telephone numbers. When writing to them for information, please enclose a self-addressed, stamped envelope.

### ALABAMA

**Birmingham Commodore Computer Club**
Harry Jones
4845 Ave. B, Lot 7B
Birmingham, AL 35208
205/923-9260

**Huntsville Alabama Commodore Komputer Society (HACKS)**
Hal Carey
9002 Berclair Rd.
Huntsville, AL 35802
205/883-0223

**Shoals Commodore User Group**
Warren Pratt
809 W. 6th St.
Sheffield, AL 35660
205/381-1561

**Walker Area Computer Club (WACC)**
Daniel McGuire
89 McCrory St.
Cordova, AL 35550
205/483-7833

### ALASKA

**COMPOOH-T**
Paul Mercer
P.O. Box 118
Old Harbor, AK 99643
907/286-2253

### ARIZONA

**Arizona VIC and 64 Users**
Tom Monson
904 W. Marlboro Cir.
Chandler, AZ 85224
602/963-6149

**Catalina Commodore Computer Club**
George A. Pope
2012 Ave. Guillermo
Tucson, AZ 85710

**Commodore User Group**
Michael Stephany
4578 Monarch Dr.
Sierra Vista, AZ 85635

**Commodore User Group of Arizona**
Doug Peters
P.O. Box 21291
Phoenix, AZ 85036
602/831-1534

### AUSTRALIA

**Australian Computer Education Association**
P.O. Box 194
Corinda 4075, Old Australia

### CALIFORNIA

**B & S 64-PET User Group**
Bryan Goldschlag
46 Banbridge Pl.
Pleasant Hill, CA 94523
415/938-0764

**California Area Commodore Terminal User Society (CACTUS)**
Darrell L. Hall
P.O. Box 1277
Alta Loma, CA 91701
714/947-0742

**Central Coast Commodore User Group**
Gilbert Vela
4237 Plumeria Ct.
Santa Maria, CA 93455
805/937-4174

**C-64 Users E & R**
215 W. 1st St., Suite 10548
Tustin, CA 92680

**Commodore 64 West**
Don Campbell
2917 Colorado Ave.
Santa Monica, CA 90404
213/828-9308

**Diablo Valley Commodore User Group**
Ben Braver
762 Ruth Dr.
Pleasant Hill, CA 94523
415/671-0145

**DUG (Danville User Group)**
Kent E. Davis
185 Front St., Suite 106
Danville, CA 94526
415/820-1222

**Napa Valley Commodore Computer Club**
Mick Winter
P.O. Box 2935
Yountville, CA 94599
707/944-2797

**PALS**
Jo Johnson
886 S. K
Livermore, CA 94550

**Peninsula Commodore User Group**
Timothy Avery
549 Old County Rd.
San Carlos, CA 94070
415/593-7697

**PET-On-The-Air**
Max J. Babin
525 Crestlake Dr.
San Francisco, CA 94132

**Sacramento Commodore Computer Club**
Robyn W. Graves
8120 Sundance Dr.
Orangevale, CA 95662

**San Fernando Valley Commodore User Group (SFVCUG)**
Thomas Lynch
21208 Nashville
Chatsworth, CA 91311
213/709-4736

**San Francisco Commodore Users Group**
Roger Tierce
278-27th Ave. #103
San Francisco, CA 94121
415/387-0225

**San Luis Obispo VIC-20/64 Computer Club**
1766 9th St.
Los Osos, CA 93402
805/528-3371

**So. Cal. 20/64 Users Group**
Star Route Box 1-C
Pine Valley, CA 92062
619/473-8087

**South Bay Commodore 64 Users Group**
Q.J. Miguel Gallego Garcia
P.O. Box 3193
San Ysidro, CA 95073

**South Orange County User Group**
Steve Wimer
2314 Monte Cristo
San Clemente, CA 92672

**SPHINX**
Richard L. Heinze
1240 Mills St. Apt. 4
Menlo Park, CA 94025
415/325-0127

**Twenty/Sixty-Four**
Don Cracraft
P.O. Box 18473
San Jose, CA 95158
408/264-2064

**Valley Computer Club**
Dr. Evan M. Thompson
661 Meadowlark Dr.
Turlock, CA 95380

**VIC-20 Owners Resource Computer Enthusiasts (V.O.R.C.E.)**
Stu
1740 Bodega Ave.
Petaluma, CA 94952
707/763-8552

## CANADA

**Barrie User Group**
Gail Hook
58 Steel St.
Barrie, Ontario
Canada L4M 2E9

**Calgary Commodore User Group**
John Hazard
37 Castleridge Dr. N.E.
Calgary, Alberta,
Canada T3J 1P4

**Calgary Commodore Computer Club**
Radu Olanson
47 Coachwood Pl. N.W.
Calgary, Alberta,
Canada T3H 1E1

**Commodore Computer Club**
Niels Hansen Trip
P.O. Box 91164
West Vancouver, B.C.,
Canada V7V 3NG
604/738-3311

**Commodore 64 User Group**
Carol Scheniman
1322 Naples St.
Oshawa/Whitby,
Ontario, Canada L1K 1J6

**Nova Scotia Commodore Computer User Group**
John Robinson
66 Landrace Cres.
Dartmouth, Nova Scotia
Canada B2W 2P9
902/434-1524

**Sarnia Commodore-64 User Group**
Susan Timar
1122 Wilson Dr.
Sarnia, Ontario,
Canada N7S 3J6
519/542-2534

**Toronto PET Users Group**
1912A Avenue Rd., Suite 1
Toronto, Ontario,
Canada M5M 4A1
416/782-9252
BBS 416/223-2625 (7:30 p.m.-9:00 a.m. Eastern Time)

**Utilisateurs De Commodore (Commodore User Group)**
P.O. Box 685 STN. H
Montreal, Quebec,
Canada H3G 2MG

**Vancouver Commodore User Group**
Chris Brown
Canada
503/573-8489

## CONNECTICUT

**Commodore Users**
Christopher D. Roberts
P.O. Box 1213
Stratford, CT 06497
203/378-8258

**Computer Users Group**
Liz Rafalowsky
Halls Hill Rd.
Colchester, CT 06415
203/537-2117

**New London Area Commodore Users**
Robert Kind
P.O. Box 1608
Groton, CT 06340
203/887-0238

**VIC User Group**
Carol Doyle
1070 S. Colony Rd.
Wallingford, CT 06492
203/269-7595

## ENGLAND

**PET User Group**
Barry Miles
Polytechnic of North London-
Holloway Rd.
London, England N7 8DB

## FLORIDA

**Bits and Bytes Computer Club**
Frank H. Topping
1859 Neptune Dr.
Englewood, FL 33533
813/474-6359

**Brandon User Group**
Paul Daugherty
108 Anglewood Dr.
Brandon, FL 33511
813/685-5138

**Central Florida Commodore User Club**
Stephen K. McHaney
P.O. Box 15949
Orlando, FL 32858
305/298-4709

**Central Florida Commodore User Group**
Earl Preston
6321 Ridgeberry Dr.
Orlando, FL 32811

**Commodore Computer Club**
Chuck Fechko
P.O. Box 21138
St. Petersburg, FL 33742
813/391-5219
BBS 5-10 p.m. 7 days a week

**Commodore Computer Club**
David Phillips
P.O. Box 9726
Jacksonville, FL 32208
904/764-5457

**Commodore Users Group of Tallahassee**
Dave Lang
2501 Debden Ct.
Tallahassee, FL 32308
904/893-6749

**El Shift O (VIC 20/64 Commodore Users)**
Mike Schnoke
P.O. Box 548
Cocoa, FL 32922

**Gainesville Commodore User Group**
Louis Wallace
P.O. Box 14716
Gainesville, FL 32604

**Gulf Coast Computer Club**
Jim Johnson
131 Fox Run
Port Richey, FL 33568
813/863-7954

**Miami 64 User Group**
Dr. Eydie Sloane
P.O. Box 561689
Miami, FL 33256
305/274-3501

**OTog Users Group for Commodore 64 VIC 20 Users**
Dax Tacey
321 Alma St.
Kissimmee, FL 32741

**VIC/64 Heartland User Group**
Tom Keough
1220 Bartow Rd. #23
Lakeland, FL 33801
813/666-2132

## GEORGIA

**C.C.S. User Groups**
Chuck Morris
P.O. Box 656
Newman, GA 30264

**Commodore User Group of Augusta**
David A. Dumas
1011 River Ridge Rd. Apt. 14-A
Augusta, GA 30909
404/738-7223

**Data Swappers**
Billy Peeples
1773B Alabama Ave.
Albany, GA 31705
912/431-0031

**Liberty Commodore Users Group**
Jeannette Burger
P.O. Box 973
Hinesville, GA 31313

**Middle Georgia Commodore User Group**
Anthony "Jim" Hornick
510 Forest Lake Drive
Warner Robins, GA 31093
912/922-7876

**VIC Educators User Group**
Dr. Al Evans
Cherokee County Schools
110 Academy St.
Canton, GA 30114

## HAWAII

**20/64 Hawaii**
T.A. Clay
98-487 Koauka Loop Apt. 804
Aiea, HI 96701

**20/64 Hawaii**
Wes Goodpastor
P.O. Box 966
Kailua, HI 96734

**Commodore User Group of Honolulu**
Jay Calvin
1626 Wilder #701
Honolulu, HI 96822
808/944-9380

## IDAHO

**Commodore User Group**
Leroy Jones
548 E. Center
Pocatello, ID 83201
208/233-4294

**S.R.H.S. Computer Club**
Barney Foster
Salmon River High School
Riggins, ID 83549

## ILLINOIS

**ASSM/TED User Group**
Brent Anderson
200 S. Century
Rantoul, IL 61866
217/893-4577

**Bloomington-Normal Commodore User Group (BNCUG)**
Debra A. Landre
P.O. Box 1058
Bloomington, IL 61702
309/454-1061

**Commodore 64 User Group**
Gus Pagnotta
P.O. Box 572
Glen Ellyn, IL 60137
312/790-4320

**Fox Valley PET User Group**
Art Dekneef
833 Willow
Lake in the Hills, IL 60102
312/658-7321

**PAPUG Peoria Area PET Users' Group**
Max Taylor
800 SW Jefferson St.
Peoria, IL 61605
309/673-6635

**PET VIC Club (PVC)**
Paul Schmidt
40 S. Lincoln
Mundelein, IL 60060
312/566-8685

**Regional Association of Programmers (RAP 64/20)**
Gene A. Meyers
7358 W. 108th Pl.
Worth, IL 60482
312/448-0485

**Southern Illinois Commodore User Club**
David E. Lawless
1707 E. Main St.
Olney, IL 62450

**The Kankakee Hackers**
Rich Westerman
RR #1, Box 279
St. Anne, IL 60964
815/933-4407

**VIC-20 Commodore-64 User Support Group**
David R. Tarvin, Sr.
114 S. Clark St.
Pana, IL 62557
217/562-4568

**Western Illinois PET User Group (WIPUG)**
Edward L. Mills
Rt. 5 Box 75
Quincy, IL 62301
217/656-3671

## INDIANA

**Commodore Computer Club**
John Patrick
3814 Terra Trace
Evansville, IN 47711
812/477-0739

**Commodore Hardware User Group (C*H*U*G)**
Tim Renshaw
9651 E. 21st St.
Indianapolis, IN 46229
317/899-2003

**Commodore User Group/ The National Science Clubs of America-Commodore Users Division**
Brian Lepley
7704 Taft St.
Merrillville, IN 46410

**Louisville Users of Commodore of KY (LUCKY)**
Melanie A. Roesser
127 Locust St.
Jeffersonville, IN 47130

**Manchester User Group (MUG)**
Richard M. Bellows
606 E. Ninth St.
N. Manchester, IN 46962

**Northern Indiana Commodore Enthusiasts (NICE)**
Eric Bean
927 S. 26th St.
South Bend, IN 46615
219/288-2101

**The CBM 64 Club**
Jim Colyer
4755 Kinser Pike
Bloomington, IN 47401
812/332-6645

**VIC Indy Club**
Fred Imhausen
P.O. Box 11543
Indianapolis, IN 46201
317/357-6906

**Western Indiana Commodore Users (W.I.C.U.)**
Dennis C. Graham
912 South Brown Ave.
Terre Haute, IN 47803
812/234-5099

## IOWA

**Commodore Computer User Group of Iowa**
Curtis L. Shiffer
P.O. Box 3140
Des Moines, IA 50316
515/282-1388

**COUGAR (COmmodore Users Group Ames Region)**
Perry Hansen
662 Pammel Ct.
Ames, IA 50010
515/296-2181

**Quad City Commodore Computer Club**
John N. Yigas
1721 Grant St.
Bettendorf, IA 52722
319/355-2641

**Siouxland Commodore Club**
Gary Johnson
2700 Sheridan St.
Sioux City, IA 51103
712/258-7903

## KANSAS

**Commodore User Group of Wichita Inc.**
Walter Lounsberry
Rt. 1, Box 115
Viola, KS 67149
316/545-7460

**Strictly VIC Users Group**
Lloyd D. Pitchford
P.O. Box 66
Sedgwick, KS 67135

## KENTUCKY

**Bowling Green Commodore Users Group**
Alex Fitzpatrick
Rt. 11, Creekside Apt. #6
Bowling Green, KY 42101
502/781-9098

**The Commodore Connection**
Jim Kemp
1010 S. Elm
Henderson, KY 42420
502/827-8153

## LOUISIANA

**Ark-La-Tex Commodore 64 Club**
Pete Whaley
198 India Dr.
Shreveport, LA 71115
318/797-9702

**C-64 Club of Baton Rouge**
Tommy Parsons
5551 Corporate Blvd. Suite 3L
Baton Rouge, LA 70808
504/766-7408

**Commodore PET User Group**
Wayne Lowery
616 N. Niagara Circle
Gretna, LA 70053
504/821-8436

**Commodore User Group of Ovachita**
Beckie Walker
P.O. Box 175
Swartz, LA 71281
318/343-8044

**Sixty Four 'EM**
Dennis Dillenkoffer
4559 Cerise
New Orleans, LA 70127
504/244-0237

**The VIC/64 Connection**
Ronnie Romero
P.O. Box 1322
Abbeville, LA 70511
318/898-0635

## MAINE

**So. Me.-64**
Ed Moore
10 Walker St.
Portland, ME 04102

**VIC and Commodore Users of Maine (V.A.C.U.M.)**
Pat Young
RFD 1, Box 103
Ellsworth, ME 04605

## MARYLAND

**Capitol Area PET Enthusiasts (VIC-20)**
William Spillane
P.O. Box 1602
Rockville, MD 20850
301/340-7417

**Compucats' Commodore Computer Club**
Betty Schueler
680 W. Bel Air Ave.
Aberdeen, MD 21001
301/272-4195
or 272-0472

**Hagerstown User Group (HUG XX)**
Joseph Rutkowski
23 Coventry Ln.
Hagerstown, MD 21740
301/797-9728

**Long Lines Commodore Club**
Tom Davis
323 N. Charles St. Room 201
Baltimore, MD 21201
301/547-2566

**Parklawn VIC-20 C-64 User Club**
Dr. S. R. Joshi
6001 Poindexter Ln.
Rockville, MD 20852
301/443-4300

**Rockville VIC/64 Users Group**
Thomas L. Pounds
P.O. Box 8805
Rockville, MD 20856
301/231-7823

**VIC and 64 User Group**
Tom Deriggi
21000 Clarksburg Rd.
Boyds, MD 20841
301/428-3174

**Westinghouse Friendship Site Commodore Users Group**
Lee Barron
P.O. Box 1693
Baltimore, MD 21203
301/765-7631
(c/o Westinghouse Electric Corp.)

**Woodlawn 20/64 Computer Club**
George Towner
1712 Aberdeen Rd.
Baltimore, MD 21234
301/608-7867

## MASSACHUSETTS

**Boston Commodore Users, c/o The Boston Computer Society**
Three Center Plaza
Boston, MA 02108
617/367-8080

**Commodore 64 User Group of the Berkshires**
Ed Rucinski
184 Highland Ave.
Pittsfield, MA 01201

**Eastern Mass VIC-20 User Group**
Frank Ordway
6 Flagg Rd.
Marlboro, MA 01752
617/485-4677

**MASSPET Commodore User Group**
David Rogers
P.O. Box 307
East Taunton, MA 02718
617/823-1974

**Needham Area VIC-20 User Group (VICHAM)**
Ilene Hoffman-Sholar
366 Hunnewell St.
Needham, MA 02194

## MEXICO

**SIGMA**
Orvananos Enrique
Holbein No. 174-6 Col. Napoles 03710,
Mexico, D.F.
563-03-05

## MICHIGAN

**Commodore Computer Club**
John R. Walley
4106 Eastman Rd.
Midland, MI 48640
517/835-5130

**Commodore Computer Club of Toledo**
Gerald W. Carter
734 Donna Dr.
Temperance, MI 48182
313/847-0426

**DAB Computer Club**
Dennis Burlingham
P.O. Box 542
Watervliet, MI 49098
616/463-5457

**DUC-Durand Users Club (VIC-20/64)**
John Davis
6780 S. Byron Road
Durand, MI 48429
517/288-4566

**Jackson Commodore Computer Club**
Alfred J. Bruey
201 S. Grinnell St.
Jackson, MI 49203

**South Computer Club**
Ronald Ruppert
South Junior High School
45201 Owen
Belleville, MI 48111

**Southeast Michigan PET User Group (SEMPUG)**
Norm Eisenberg
32800 W. 12 Mile Rd.
Farmington Hills, MI 48018

**University of Michigan VIC-20 and C-64 User Group**
John J. Gannon
School of Public Health-Univ. of Mich.
Ann Arbor, MI 48109

**VIC 20 Mail Club**
David Koski
51074 Mott #11
Canton, MI 48188

**VIC for Business**
Mike Marotta
6027 Orchard Ct.
Lansing, MI 48910
517/394-2345

## MINNESOTA

**Minnesota Users of PET (MUPET)**
Jon T. Minerich
P.O. Box 179
Annandale, MN 55302
612/963-5056

## MISSOURI

**Association of Commodore User Groups (ACUG)**
Tony Ott
10378 Coburg Lands
St. Louis, MO 63137
314/867-0016
BBS 867-6950

**Joplin Commodore Computer User Group**
R. D. Connely
422 S. Florida Ave.
Joplin, MO 64801

**Kansas City PET User Group (KCPUG)**
Rick West
P.O. Box 36492
Kansas City, MO 64111
816/252-7628
BBS 257-2502

**Mid-Missouri Commodore User Club**
1804 Vandiver Dr.
Columbia, MO 65202
314/474-4511

**Northland Amateur Radio Association**
Alan Boyer
528 Skyline Dr.
Liberty, MO 64068
816/781-6987

**The Commodore User Group of St. Louis, Inc.**
Dan Weidman
P.O. Box 6653
St. Louis, MO 63125
314/968-4409

# NEWS & PRODUCTS

## New Printer From Commodore

Commodore has released a new printer, the MPS-801, which will replace the 1525. According to a source at Commodore, the MPS-801 is slightly faster than the 1525, features a cartridge ribbon, paper advance knob and button, and supports full-size paper. The printer also includes two serial ports to enable daisy chaining, and the printing mechanism has been relocated for a quieter sound. Price was not set at press time.

## *Quick Reference Guides* For VIC And 64

John Wiley & Sons has published *Quick Reference Guides* for the VIC, 64, and Timex 1000 computers, similar to the guides the company previously published for the IBM PC, Apple II, and Atari 800.

The 6 × 12-inch four-panel guides list and define such items as BASIC statements, system controls, memory statements, video and graphic controls, and other topics.

The guides are available for $2.95 each.

*John Wiley & Sons, Inc.*
*605 Third Avenue*
*New York, NY 10158*
*(212) 850-6000*

## VIC-20 Typing Tutorial

*Mastertype*, a typing tutorial from Brøderbund, is now available for the VIC-20 in a cartridge format.

The tutorial consists of 18 progressive lessons set within a game format. You defend a spaceship from a barrage of letters or words by correctly typing them as rapidly as possible.

Lessons may be saved on either disk or cartridge for future use. Included are an illustrated instruction manual, hints for winning the game, and directions for creating your own customized lessons to improve typing and spelling.

The VIC-20 cartridge version of *Mastertype* sells for $39.95.

*Bróderbund Software, Inc.*
*17 Paul Drive*
*San Rafael, CA 94903*
*(415) 479-1170*

## Twin-Disk Adventure Game For 64

*Cyberworld*, a two-disk, interactive keyboard/joystick adventure game for the Commodore 64, has been produced by Progressive Peripherals & Software.

The player's mission is multilayered, set on the planet Cyber, and includes full use of the 64's sprite capabilities. There are three subadventures, which begin with the hijacking of a Drokon warship and advance to a defense of the planet. The final mission is a hunt-and-tracking game.

Available only on diskettes, the program is available for $39.95.

*Progressive Peripherals & Software*
*2186 South Holly, Suite 2*
*Denver, CO 80222*
*(303) 759-5713*

## Music, Business, Personal Programs For 64

M'Soft has developed several new programs for the Commodore 64 on disk.

*Smart* ($79.95) features five different programs, all of which load from one main menu: word processing, money management, amortization, record keeping, and time management.

*Wallstreet Microscope* ($99.95) offers price and financial analysis of common stocks, with each stock rated against ten criteria. It is available with a ten-year history of Fortune 500 companies.

*Musicwriter-64* ($69.95) is a music-composing, editing, and playing program that also prints sheet music for compositions created using the system.

*Double E Electronics*
*12027 Pacific Street*
*Omaha, NE 68154*
*(402) 334-7870*

# Computa-Law

## Legal Agreements*

For Your

## COMMODORE 64
## VIC 20 (16K)
## IBM-PC & (Jr.)

**Just answer the questions & your computer & printer does the rest!**

- ☐ Simple Will
- ☐ Agreement of Sale - Real Estate
- ☐ Agreement of Sale - Goods
- ☐ Lease - Residential
- ☐ Lease - Commercial
- ☐ Power of Attorney
- ☐ Employment Contract
- ☐ Promissory Note
- ☐ Partnership Agreement
- ☐ Computer Software Contract
- ☐ Computer Hardware Contract
- ☐ Pre-Nuptial Agreement
- ☐ Separation Agreement
- ☐ Construction Contract
- ☐ General Release

For informational purposes only not intended as a substitute for legal advise.

**Guaranteed to work on your printer.**

☐ $19.95 Each Program (Cassette)
☐ $24.95 Each Program (Disk)
Add $1.50 postage & handling.

65 Other Business & Home Programs also available

FREE CATALOG

LEGAL BYTE SOFTWARE
Box 579, Gwynedd Valley, PA 19437
(215) 643-7666 (609) 424-5485


**The Banner Machine™**

For the **Commodore 64** (3 extra fonts available). For the **VIC-20** with 24K memory (2 extra fonts available). • Use on any Gemini or Epson MX with Graftrax or the FX and RX printers. Also Commodore 1525E and Banana with the C-64. • Menu-driven program operates like a word processor. • Makes signs up to 13" tall by any length. • Makes borders of widths up to ¾". • 8 sizes of letters from ¾" to 8" high. • Proportional spacing; Automatic centering; Right and left justifying. • $49.95 Tape or Disk (Specify computer equipment)

**For the Commodore 64:**

**Home Finance Manager** Keep detailed records of tax deductions, bank payments, monthly charges, individual item expenses, and checks. Store more than 200 transactions per month. $39.95

**CTRL-64** Permits listing of C-64 programs on non-Commodore printers. Lists control symbols in readable form. Tape or disk $24.95

**Microbroker** Exciting, realistic and educational stock market simulation. $34.95 Tape or Disk

**Preschool Educational Programs** ABC Fun; 123 Fun; and Ginger the Cat with: Addition and Subtraction, Number Hunt, and Letter Hunt. All 5 programs have bright color, music, and action. **Each** $14.95

**Formulator** A scientific calculator for tasks which require repetitive arithmetic computations. Save formulas and numeric expressions. $39.95

**Grade Organizer** Teachers—store grades for 6 classes, up to 40 students each, 680 grades per student. Print interim and final reports, class rosters, and more! Disk $39.95

**Cardinal Software™**

Virginia Micro Systems, 13646 Jeff Davis Hwy., Woodbridge, VA 22191
**Phone (703) 491-6502**
Ask for our free catalog

Commodore 64 and VIC-20 are registered trademarks of Commodore Electronics Ltd.

## For VIC - 20 / COM - 64

| MICRODIGITAL ARCADE GAMES | VIC | 64 |
|---|---|---|
| Skramble (T) (exciting) | 10.75 | 12.75 |
| Gridder (T) (grid chase) | 10.75 | 12.75 |
| Snakman (T) (pac man) | 10.75 | 12.75 |
| Pinball Wizard (T) | 10.75 | 12.75 |
| **PRACTICALC PLUS (16K)(T)** | **43.95** | **43.95** |
| Temple of Apshai (16K) (T) | 27.50 | 27.50 |
| Sword of Fargoal (16K) (T) | 21.80 | N/A |

(CALL FOR DISK PRICING)

# 16K MEMORY EXPANSION

**ONLY $49.95**

★ 14 Day Money Back Guarantee
★ Boosts VIC to 21K RAM
★ Top Quality, Fully Tested
★ 90 Day Warranty

**for IMMEDIATE SHIPMENT on Credit Card Orders**

## Call: (303) 245-9012

10 AM - 9 PM MST Every Day

**ASSEMBLY TECHNOLOGY**
2692 Hwy 50 Suite 210
Grand Junction, CO 81503

Personal checks allow 3 weeks
Shipping & handling $2.50
Colorado Residents add Sales Tax
COD add $2.00

# COMPUTER DISCOUNT

**TOLL FREE 1-800-621-6131 FOR ORDERS**
**4251 W. Sahara Ave., Suite E Las Vegas, Nevada 89126**
**MONDAY THROUGH SATURDAY ● 9 AM TO 6 PM**

| | |
|---|---|
| Comm 64 | $229 |
| 1541 Disk Drive | 249 |
| 1525 Printer | 229 |
| 1702 Color Mont. | 259 |
| Hes Mon | 29 |
| Paper Clip w/p | 115 |
| Calc Result | 140 |
| Sysres-Utility | 90 |
| Renaissance | 30 |
| Vic-20 | 90 |
| Datasette | 64 |
| 1600 Modem | 85 |
| Word Processor | 95 |
| 1311 Joystick | 8 |
| 1312 Paddles | 16 |
| 1210 3k Expander | 34 |

**SPECIAL**

Comm 64 ..... $785
1541 Disk Drive
152EE Printer
1600 Phone Modem

## HARDWARE

| | |
|---|---|
| C. Itoh Prowriter | $379 |
| Nec. 8023A | 429 |
| Banana Printer | 209 |
| Hayes 1200 Modem | 489 |

## COMM. 64 DISK

| | |
|---|---|
| Temple Of Apshai | $33 |
| Upper Reaches Apshai | 18 |
| Jump Man | 33 |
| Zork I | 33 |
| Zork II | 33 |
| Zork III | 33 |
| Frogger | 30 |
| Jawbreaker | 24 |
| Ft. Apocalyse | 30 |
| Pharoh's Curse | 30 |
| Starter Pack D/C | $22 |
| Word Machine | 17 |
| Pet Emulator | 17 |
| Gen Ledger | 69 |
| Mail List Mgr. | 43 |
| Hes 6502 | 23 |

**New Educational Programs**
**Purchases can be made by check, money order, C.O.D. Carte Blanche and Diners Club. 1-702-367-2215**

ATARI

# YORK 10™ CASSETTES

**DATA TRAC BLANK CASSETTES**

**C-05, C-06, C-10, C-12, C-20, C-24, C-32**

**From the leading supplier of Computer Cassettes, new, longer length C-12's (6 minutes per side) provide the extra few feet needed for some 16K programs.**

**BASF-LHD (DPS) world standard tape.**
**Premium 5 screw shell with leader.**
**Error Free • Money back guarantee.**

Call: 213/700-0330 for IMMEDIATE SHIPMENT on Credit Card Orders
*ORDER NOW... ...MAIL TO...* YORK 10™ 9525 Vassar Ave. #G Chatsworth, CA 91311

ORDER FORM

| ITEM | 1 DOZ. | 2 DOZ. | TOTAL |
|---|---|---|---|
| C-05 | ☐ 7.00 | ☐ 13.00 | |
| C-06 | ☐ 7.00 | ☐ 13.00 | |
| C-10 | ☐ 7.50 | ☐ 14.00 | |
| C-12 | ☐ 7.50 | ☐ 14.00 | |
| C-20 | ☐ 8.75 | ☐ 16.50 | |
| C-24 | ☐ 9.00 | ☐ 17.00 | |
| C-32 | ☐ 11.00 | ☐ 21.00 | |
| Hard Box | ☐ 2.50 | ☐ 4.00 | |
| Storage Caddy @ 2.95 ea. Qty | | | |
| SUB TOTAL | | | |
| Calif. residents add sales tax | | | |
| Shipping handling (any quantity) | | | 3.50 |
| Outside 48 States, ADD $1 per doz. cassettes or boxes | | | |
| TOTAL | | | |

**FREE 1 STORAGE CADDY with every 4 doz. cassettes purchased.** (does not apply to 500 quantity offer)

Each cassette includes 2 labels only. Boxes sold separately. In Cont. U.S. shipment by U.P.S. If Parcel Post preferred, check here. ☐

**500 C-12's or C-10's ... 38¢ ea** w/labels, add 4¢ ea /shipping $17 (Free Caddy offer does not apply)

Check or M.O. enclosed ☐ Send Quantity Discounts ☐
Charge to credit card: VISA ☐ MASTERCARD ☐

Card No. _____ Exp. _____
Name _____
Address _____
City _____ State/Zip _____
Signature _____ Phone _____
Computer make & model _____ Disk? (y/n) _____ #G

## VIC And 64 Spreadsheet

*BEC*, a spreadsheet analysis program for the entrepreneur, has been developed by Lawco, Ltd., for the VIC and 64.

The program computes the break-even point for new products by using the product's expense data. It also uses the product's sales price to compute a break-even sales quantity or the quantity produced to figure a break-even sales price.

Total fixed costs, total variable costs, and sales revenue at the break-even point are also computed. Multiple break-even points can be produced by varying the sales price, the production quantity, and/or the cost figures.

*BEC* is available at $79.95 on tape and $89.95 on disk.

*Lawco, Ltd.*
*P.O. Box 1337*
*Cupertino, CA 95015*
*(408) 733-0739*

## Action Game For 64 With Disk Drive

*Crazy Conveyors*, produced by Bytes and Bits, is a machine language action game for the Commodore 64 with disk drive.

The game uses multicolor sprites, custom characters in 11 different colors for building blocks, ladders, fire poles, rotating pulleys, moving conveyors, and bonus boxes. *Crazy Conveyors* also uses three-part harmony and is playable with joystick or keyboard.

The game contains the Screen Creator which expands the game disk and other disks for additional screens.

The price for the game is $29.95.

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

## Cassette Copier For VIC And 64

T & M Products has produced a Data Cassette Copier, which allows all cassettes to be duplicated by interfacing two Datassette recorders with a Commodore 64 or a VIC-20 computer.

The Data Copier will duplicate all programs and load machine language programs without using VICMON. A Micro-speaker plus LED allows you to monitor the data by sight and sound.

The Data Copier is available for $24.95 plus $2 shipping and handling. Power is supplied by the computer at the cassette port, so no batteries are required.

*T & M Products*
*P.O. Box 1172*
*De Soto, TX 75115*

## Math Drill Program

*Let's Learn Math*, a menu-driven addition and subtraction drill program for the Commodore 64,

has been released by Micro-Systems Software.

The program has four levels of difficulty and is designed for youngsters age 6 to 12. All problems are solved column by column. The need for pencil and paper is eliminated. Right and wrong answers are flagged, and the correct answer is displayed if the entry was wrong. At the end of each session, a report of the number of correct and incorrect entries is displayed.

*Let's Learn Math* is available on tape for $12.95.

*Micro-Systems Software*
*4017 Adams #263*
*Indianapolis, IN 46205*

---

COMPUTE!'s GAZETTE *welcomes announcements of new products for VIC-20 and Commodore 64 computers, especially products aimed at beginning to intermediate users. Please send press releases and photos well in advance to: Tony Roberts, Assistant Managing Editor,* COMPUTE!'s GAZETTE, *P.O. Box 5406, Greensboro, NC 27403.*

---

*New product releases are selected from submissions for reasons of timeliness, available space, and general interest to our readers. We regret that we are unable to select all new product submissions for publication. Readers should be aware that we present here some edited version of material submitted by vendors and are unable to vouch for its accuracy at time of publication.*

# COMPUTE!'s Gazette Back Issues

**JULY 1983:** Commodore 64 Video Update, Snake Escape, Alfabug, VIC Marquee, Word Hunt, VIC Timepiece, product reviews, Learning To Program In BASIC, Quickfind, 64 Paddle Reader, Machine Language For Beginners, Enlivening Programs With Sound, Using Joysticks On The 64, Simple Answers To Common Questions, VICreations — Speedy Variables, 64 Explorer.

**AUGUST 1983:** Your First Hour With A Computer, Should You Join A Users Group?, Guide To Commodore Users Groups, The Viper, Cylon Zap, product reviews, VIC/64 Mailing List, Word Spell, Global Scan For VIC/64, Machine Language For Beginners, VIC Title Screens, 64 Hi-Res Graphics Made Easy, VIC/64 Four-Speed Brake, Disk Menu, Using A 1540 Disk Drive With The 64, Playing Computer Music, Simple Answers To Common Questions, HOTWARE, VICreations — Caring For Disk Drives/Cassettes, 64 Explorer, News & Products.

**OCTOBER 1983:** The Anatomy of Computers, Telegaming Today And Tomorrow, Commodore's Public Domain Programs, Oil Tycoon, Re-Beep, product reviews, Aardvark Attack, Word Match, A SHIFTy Solution: The WAIT Command, Program Transfers, Machine Language For Beginners, Improved Paddle Reader Routine, How To Use Tape And Disk Files, Understanding 64 Sound — Part 1, Speeding Up The VIC, Simple Answers To Common Questions, HOTWARE, Horizons 64 — Improving 64 Video Quality, VICreations — Using The VIC's Clock, News & Products.

**DECEMBER 1983:** A Survival Guide For Beginners, Telecommuting: Dawn Of The Electronic Cottage, The Inner World Of Computers — Part 2, Getting Started With A Disk Drive — Part 2, Spike, Space Duel, Bowling Champ, Saucer Shooter For The VIC-20, Budget Planner, The Note Name Game, Spelling Bee, Educational Games: A Kid's View, Disk File Manager, VIC Music Writer, Thinking, VIC Billboard, Tricks For Saving Memory, Easy Screen Formatting, Power BASIC: Foolproof Input For The VIC and 64, Sprites Made Easy, Sprite Creation On The 64, Machine Language For Beginners, Simple Answers To Common Questions, HOTWARE, VICreations: Custom Characters On The Expanded VIC, Horizons 64, MLX — Machine Language Entry For The Commodore 64, The Beginner's Corner: Computer Choreography, Computing For Kids: Your Wish Is My Command, The Automatic Proofreader.

**JANUARY 1984:** Word Processing In The Home, SpeedScript Word Processor For VIC And 64, The Inner World Of Computers — Part 3, Getting Started With A Disk Drive — Part 3, Cave-In For VIC-20, Hardhat Climber, Tetracrystals Of Veluria, Canyon Cruiser, Computing For Families: New Family Learning Games, 64 Electronic Notepad, Alpha-Shoot, The Beginner's Corner: Built-In Functions, Graph Plotter, 64 BASIC Aid, LIST Freezer, Machine Language For Beginners: Addressing, HOTWARE, VICreations: Using The Dynamic Keyboard, Horizons 64, Simple Answers To Common Questions, The Automatic Proofreader, MLX — Machine Language Entry For The Commodore 64 And VIC-20.

Back issues of July and August 1983 are $2.50 each. Issues from October forward are $3. Bulk rates are 6 issues for $15 or 12 issues for $30. All prices include freight in the U.S. Outside the U.S. add $1 per magazine order for surface postage. $4 per magazine for air mail postage. ALL BACK ISSUES ARE SUBJECT TO AVAILABILITY.

In the continental U.S. call
**TOLL FREE 800-334-0868**
(in North Carolina call 919-275-9809)

Or write to:
**COMPUTE!'s Gazette for Commodore Back Issues**
**P.O. Box 5406**
**Greensboro, North Carolina 27403, USA**

Prepayment required in U.S. funds. MasterCard, VISA, and American Express accepted. North Carolina residents please add 4% sales tax.

# A Beginner's Guide To Typing In Programs

## What Is A Program?

A computer cannot perform any task by itself. Like a car without gas, a computer has *potential*, but without a program, it isn't going anywhere. Most of the programs published in *COMPUTE!'s Gazette* for Commodore are written in a computer language called BASIC. BASIC is easy to learn and is built into all VIC-20s and Commodore 64s.

## BASIC Programs

Each month, *COMPUTE!'s Gazette* for Commodore publishes programs for both the VIC and 64. To start out, type in only programs written for your machine, e.g., "VIC Version" if you have a VIC-20. Later, when you gain experience with your computer's BASIC, you can try typing in and converting certain programs from another computer to yours.

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

## Brackets And Special Characters

The exception to this typing rule is when you see the curved bracket, such as "{DOWN}". Anything within a set of brackets is a special character or characters that cannot easily be listed on a printer. When you come across such a special statement, refer to "How To Type In *COMPUTE!'s Gazette* Programs."

## About DATA Statements

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

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

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

## Get To Know Your Machine

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

## A Quick Review

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

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

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

*We regret that we are not able to respond to individual inquiries about programs, products, or services appearing in* COMPUTE!'s Gazette *for Commodore due to increasing publication activity. On those infrequent occasions when a published program contains a typo, the correction will appear in the magazine, usually within eight weeks. If you have specific questions about items or programs which you've seen in* COMPUTE!'s Gazette *for Commodore, please send them to Gazette Feedback, P.O. Box 5406, Greensboro, NC 27403.*

# How To Type In *COMPUTE!'s Gazette* Programs

Many of the programs which are listed in *COMPUTE!'s Gazette* contain special control characters (cursor control, color keys, inverse video, etc.). To make it easy to know exactly what to type when entering one of these programs into your computer, we have established the following listing conventions.

Generally, any VIC-20 or Commodore 64 program listings will contain bracketed words which spell out any special characters: {DOWN} would mean to press the cursor down key. {5 SPACES} would mean to press the space bar five times.

To indicate that a key should be *shifted* (hold down the SHIFT key while pressing the other key), the key would be underlined in our listings. For example, S would mean to type the S key while holding the shift key. This would appear on your screen as a "heart" symbol. If you find an underlined key enclosed in braces (e.g., {10 N}), you should type the key as many times as indicated (in our example, you would enter ten shifted N's).

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

Rarely, you'll see a solitary letter of the alphabet enclosed in braces. These characters can be entered on the Commodore 64 by holding down the CTRL key while typing the letter in the braces. For example, {A} would indicate that you should press CTRL-A. You should never have to enter such a character on the VIC-20, but if you do, you would have to leave the quote mode (press RETURN and cursor back up to the position where the control character should go), press CTRL-9 (RVS ON), the letter in braces, and then CTRL-0 (RVS OFF).

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

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

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

Use the following table when entering cursor and color control keys:

| When You Read: | Press: | See: | When You Read: | Press: | See: | When You Read: | Press: | See: |
|---|---|---|---|---|---|---|---|---|
| {CLEAR} | SHIFT + CLR/HOME | | {CYN} | CTRL + 4 | | {7} | C= + 7 | |
| {HOME} | CLR/HOME | | {PUR} | CTRL + 5 | | {8} | C= + 8 | |
| {UP} | SHIFT + CRSR ↑ | | {GRN} | CTRL + 6 | | {F1} | f1 | |
| {DOWN} | CRSR ↑ | | {BLU} | CTRL + 7 | | {F2} | SHIFT + f1 | |
| {LEFT} | SHIFT + CRSR ← | | {YEL} | CTRL + 8 | | {F3} | f3 | |
| {RIGHT} | CRSR ← | | {1} | C= + 1 | | {F4} | SHIFT + f3 | |
| {RVS} | CTRL + 9 | | {2} | C= + 2 | | {F5} | f5 | |
| {OFF} | CTRL + 0 | | {3} | C= + 3 | | {F6} | SHIFT + f5 | |
| {BLK} | CTRL + 1 | | {4} | C= + 4 | | {F7} | f7 | |
| {WHT} | CTRL + 2 | | {5} | C= + 5 | | {F8} | SHIFT + f7 | |
| {RED} | CTRL + 3 | | {6} | C= + 6 | | | | |

# MLX Machine Language Entry Program

## For Commodore 64 And VIC-20

Charles Brannon, Program Editor

**MLX is a labor-saving utility that allows almost failsafe entry of machine language programs published in COMPUTE!'s GAZETTE. You need to know nothing about machine language to use MLX—it was designed for everyone. There are separate versions for the Commodore 64 and expanded VIC-20 (at least 8K). MLX was conceived and written by Program Editor Charles Brannon. Important: MLX is required to type in the machine language programs in this issue.**

MLX is a new way to enter long machine language (ML) programs with a minimum of fuss. MLX lets you enter the numbers from a special list that looks similar to BASIC DATA statements. It checks your typing on a line-by-line basis. It won't let you enter illegal characters when you should be typing numbers. It won't let you enter numbers greater than 255 (forbidden in ML). It won't let you enter the wrong numbers on the wrong line. In addition, MLX creates a ready-to-use tape or disk file. You can then use the LOAD command to read the program into the computer, as with any program:

```
LOAD "filename",1,1   (for tape)
LOAD "filename",8,1   (for disk)
```

To start the program, you enter a SYS command that transfers control from BASIC to machine language. The starting SYS number always appears in the appropriate article.

## Using MLX

Type in and save the correct version of MLX for your computer (you'll want to use it in the future). When you're ready to type in an ML program, run MLX. MLX asks you for two numbers: the starting address and the ending address. These numbers are given in the article accompanying the ML program.

You'll see a prompt corresponding to the starting address. The prompt is the current line you are entering from the listing. It increases by six each time you enter a line. That's because each line has seven numbers—six actual data numbers plus a *checksum number*. The checksum verifies that you typed the previous six numbers correctly. If you enter any of the six numbers wrong, or enter the checksum wrong, the computer rings a buzzer and prompts you to reenter the line. If you enter it correctly, a bell tone sounds and you continue to the next line.

MLX accepts only numbers as input. It you make a typing error, press the INST/DEL key; the entire number is deleted. You can press it as many times as necessary back to the start of the line. If you enter three-digit numbers as listed, the computer automatically prints the comma and goes on to accept the next number. If you enter less than three digits, you can press either the comma, SPACE bar, or RETURN key to advance to the next number. The checksum automatically appears in inverse video for emphasis.

## MLX Commands

When you finish typing an ML listing (assuming you type it all in one session) you can then save the completed program on tape or disk. Follow the screen instructions. If you get any errors while saving, you probably have a bad disk, or the disk is full, or you've made a typo when entering the MLX program itself.

You don't have to enter the whole ML program in one sitting. MLX lets you enter as much as you want, save it, and then reload the file from tape or disk later. MLX recognizes these commands:

| | |
|---|---|
| **SHIFT-S: Save** | **SHIFT-N: New Address** |
| **SHIFT-L: Load** | **SHIFT-D: Display** |

When you enter a command, MLX jumps out of the line you've been typing, so we recommend you do it at a new prompt. Use the Save command to save what you've been working on. It will save on tape or disk as if you've finished, but the tape or disk won't work, of course, until you finish the typing. Remember what address you stop at. The next time you run MLX, answer all the prompts as you did before, then insert the disk or tape. When you get to the entry prompt, press SHIFT-L to reload the partly completed file into memory. Then use the New Address command to resume typing.

To use the New Address command, press SHIFT-N and enter the address where you previously stopped. The prompt will change, and you can then continue typing. Always enter a New Address that matches up with one of the line numbers in the special listing, or else the checksum won't work. The Display command lets you display a section of your typing. After you press SHIFT-D, enter two addresses within the line number range of the listing. You can abort the listing by pressing any key.

The special MLX commands may seem a bit confusing, but as you work with MLX, they will become valuable. For example, what if you forgot where you stopped typing? Use the Display command to scan memory from the beginning to the end of the program. When you reach the end of your typing, the lines will contain a random pattern of numbers. When you see the end of your typing, press any key to stop the listing. Use the New Address command to continue typing from the proper location.

You can use the Save and Load commands to make copies of the completed program. Use Load to reload the tape or disk, then insert a new tape or disk and use Save to make a new copy.

Be sure to save MLX; it will be used for future ML programs in COMPUTE!'s GAZETTE.

*See program listings on page 177.*

# The Automatic Proofreader

"The Automatic Proofreader" will help you type in program listings from COMPUTE!'s Gazette without typing mistakes. It is a short error-checking program that hides itself in memory. When activated, it lets you know immediately after typing a line from a program listing if you have made a mistake. Please read these instructions carefully before typing any programs in COMPUTE!'s Gazette.

## Preparing The Proofreader

1. Using the listing below, type in the Proofreader. The same program works on both the VIC-20 and Commodore 64. Be very careful when entering the DATA statements — don't type an l instead of a 1, an O instead of a 0, extra commas, etc.

2. SAVE the Proofreader on tape or disk at least twice *before running it for the first time.* This is very important because the Proofreader erases this part of itself when you first type RUN.

3. After the Proofreader is SAVEd, type RUN. It will check itself for typing errors in the DATA statements and warn you if there's a mistake. Correct any errors and SAVE the corrected version. Keep a copy in a safe place — you'll need it again and again, every time you enter a program from COMPUTE!'s Gazette.

4. When a correct version of the Proofreader is RUN, it activates itself. You are now ready to enter a program listing. If you press RUN/STOP-RESTORE, the Proofreader is disabled. To reactivate it, just type the command SYS 886 and press RETURN.

## Using The Proofreader

All VIC and 64 listings in COMPUTE!'s Gazette now have a *checksum number* appended to the end of each line, for example ":rem 123". *Don't enter this statement when typing in a program.* It is just for your information. The rem makes the number harmless if someone does type it in. It will, however, use up memory if you enter it, and it will confuse the Proofreader, even if you entered the rest of the line correctly.

When you type in a line from a program listing and press RETURN, the Proofreader displays a number at the top of your screen. *This checksum number must match the checksum number in the printed listing.* If it doesn't, it means you typed the line differently than the way it is listed. Immediately recheck your typing. Remember, don't type the rem statement with the checksum number; it is published only so you can check it against the number which appears on your screen.

The Proofreader is not picky with spaces. It will not notice extra spaces or missing ones. This is for your convenience, since spacing is generally not important. But occasionally proper spacing *is* important, so be extra careful with spaces, since the Proofreader will catch practically everything else that can go wrong.

There's another thing to watch out for: if you enter the line by using abbreviations for commands, the checksum will not match up. But there is a way to make the Proofreader check it. After entering the line, LIST it. This eliminates the abbreviations. Then move the cursor up to the line and press RETURN. It should now match the checksum. You can check whole groups of lines this way.

## Special Tape SAVE Instructions

When you're done typing a listing, you must disable the Proofreader before SAVEing the program on tape. Disable the Proofreader by pressing RUN/STOP-RESTORE (hold down the RUN/STOP key and sharply hit the RESTORE key). This procedure is not necessary for disk SAVEs, *but you must disable the Proofreader this way before a tape SAVE.*

SAVE to tape erases the Proofreader from memory, so you'll have to LOAD and RUN it again if you want to type another listing. SAVE to disk does not erase the Proofreader.

## Replace Original Proofreader

If you typed in the original version of the Proofreader (October 1983 issue), you should replace it with the improved version below. We added a POKE to the original version to protect it from being erased when you LOAD another program from tape. The POKE does protect the Proofreader, and the Proofreader itself was not affected. However, a quirk in the VIC-20's operating system means that programs typed in with the Proofreader and SAVEd on tape cannot be LOADed properly later. If you LOAD a program SAVEd while the Proofreader was in memory, you see ?LOAD ERROR. This applies only to VIC tape SAVEs (disk SAVEs work OK, and the quirk was fixed in the Commodore 64).

If you have a program typed in with the original Proofreader and SAVEd on tape, follow this special LOAD procedure:

1. Turn the power off, then on.
2. LOAD the program from tape (disregard the ?LOAD ERROR).
3. Enter: POKE 45,PEEK(174):POKE 46,PEEK(175):CLR
4. ReSAVE the program to tape.

The program will LOAD fine in the future. We strongly recommend that you type in the new version of the Proofreader and discard the old one.

## Automatic Proofreader For VIC And 64

```
100 PRINT"{CLR}PLEASE WAIT...":FORI=886TO
    1018:READA:CK=CK+A:POKEI,A:NEXT
110 IF CK<>17539 THEN PRINT"{DOWN}YOU MAD
    E AN ERROR":PRINT"IN DATA STATEMENTS.
    ":END
120 SYS886:PRINT"{CLR}{2 DOWN}PROOFREADER
     ACTIVATED.":NEW
886 DATA 173,036,003,201,150,208
892 DATA 001,096,141,151,003,173
898 DATA 037,003,141,152,003,169
904 DATA 150,141,036,003,169,003
910 DATA 141,037,003,169,000,133
916 DATA 254,096,032,087,241,133
922 DATA 251,134,252,132,253,008
928 DATA 201,013,240,017,201,032
934 DATA 240,005,024,101,254,133
940 DATA 254,165,251,166,252,164
946 DATA 253,040,096,169,013,032
952 DATA 210,255,165,214,141,251
958 DATA 003,206,251,003,169,000
964 DATA 133,216,169,019,032,210
970 DATA 255,169,018,032,210,255
976 DATA 169,058,032,210,255,166
982 DATA 254,169,000,133,254,172
988 DATA 151,003,192,087,208,006
994 DATA 032,205,189,076,235,003
1000 DATA 032,205,221,169,032,032
1006 DATA 210,255,032,210,255,173
1012 DATA 251,003,133,214,076,173
1018 DATA 003
```

# Bug-Swatter:
## Modifications And Corrections

• In "Hardhat Climber" (January), the climber may accidentally jump when the joystick is pushed diagonally. Thanks to reader Stephen A. Ohayon for discovering this correction:

```
105 IF(JVAND16)=16THENJV=1:GOTO109:rem 62
```

If you want the climber to jump only when the joystick is in the neutral position, change the first 16 to a 31.

• The program listing of "Cave-In For VIC-20" (January) was inadvertently labeled "Cave-In For 64." The program will run only on the VIC-20.

• The Commodore 64 version of "MLX: Machine Language Entry Program" (December 1983) contained lines which were longer than the 80-character limit. Lines 160, 210, and 230 should be entered with abbreviated BASIC commands (P Shift-O for POKE, ? for PRINT). If you abbreviate, you will be able to type these lines within 80 characters, but the Proofreader rem numbers will appear to be incorrect. The program listing was corrected in the January version of MLX (see below).

• In correcting the problem described above, line 210 of MLX—64 Version (January) was split into two lines. But when line 215 was added, a closing parenthesis was omitted. To correct this, add a closing parenthesis to the end of line 215.

• Some readers have had difficulty using "The Assembler" (November 1983 "Machine Language For Beginners") to enter Programs 1 and 2 (December 1983), encountering ?EXTRA IGNORED errors on lines using indexed addressing. This is because Commodore BASIC will not accept commas in the middle of INPUT. Programs 1 and 2 were disassembled using the comma convention; readers who want to use The Assembler should type the program lines *without* commas. For example, instead of *STA 1024,Y* use *STA 1024Y*.

In addition, line 200 of The Assembler should have included an Automatic Proofreader rem number of 0.

• The printer used to generate GAZETTE program listings still places an occasional random question mark. Line 1018 of "Disk File Manager" (December 1983) contains one such superfluous character. To correct this, delete the question mark after FILE=. The program runs as listed, but the Proofreader checksum number will not be correct if the question mark is included.

• "Thinking" (December 1983) contains a minor typographical error. If the REM is removed from line 2 to convert Thinking to Thinking Harder, there are nine switches instead of six. But the prompt in line 132 says there are six. To correct this, change YOUR NUMBER (1–6) to YOUR NUMBER (1–"G$").

• "Foolproof INPUT" ("PowerBASIC," December 1983) as published does not solve the problems caused by commas and colons. It disables the cursor keys and certain other function keys, but commas will still cause an ?EXTRA IGNORED error. The solution is to POKE 198,1: POKE 631,34 before each INPUT. In a program with many INPUT statements, these two POKEs could be put into a subroutine. After the POKEs, the computer will be in quote mode, which allows entry of commas and colons.

• Readers who bowled a perfect game (300) in "Bowling Champ" (December 1983) may have been disappointed to find 290 as their score. Our thanks to reader David McDonnell, Jr., for finding this bug. To correct it, change these lines:

```
209 J=0:GOSUB430:GOSUB550:T(Z9)=T(Z9)+J
                                    :rem 83
210 K=J+176:IFK=186THENK=152:T(Z9)=T(Z9)-
    J*(U=0)                         :rem 8
211 PRINTC$(Z9);T(Z9):POKEL(Z9),K:POKEL(Z
    9)+CO,4:L(Z9)=L(Z9)+1           :rem 216
```

• "Spelling Bee For VIC" (December 1983) contains a minor error. Line 360 includes a PRINT color which is accessible on the Commodore 64, but not on the VIC. To remedy this, change Commodore-5 to CTRL-5 (purple).

• Readers Wesley Evans and Dick Sloss both discovered an error in the program "Sprite BASIC" from "Sprites Made Easy" (December 1983). The problem occurs in the colors of sprites 1–7. To correct this, make the following changes:

```
170 FOR I=49152 TO 49384:READN:POKE I,N:A
    =A+N:NEXT I                     :rem 189
180 IF A<>30780 THEN PRINT "ERROR IN DATA
     STATEMENTS"                    :rem 40
280 DATA 138,164,2,145,251,32,253,174,32,
    227,192,224,16,176,146,138,153
                                    :rem 179
300 DATA 23,208,32,253,174,32,227,192,224
    ,4,176,223,134,2,70,2,144       :rem 160
340 DATA 32,158,183,164,2,96        :rem 104
```

In addition, some readers have run into problems with the "Tie Fighter" program in the same article. The program runs as listed, but only if "Sprite BASIC" is loaded and run first.

# Tree Tutor For Tots

*(Article on page 60.)*

**BEFORE TYPING...**

**Before typing in programs, please refer to "How To Type COMPUTE!'s Gazette Programs," "A Beginner's Guide To Typing In Programs," and "The Automatic Proofreader" that appear before the Program Listings.**

## Program 1: Tree Tutor For VIC

```
2 PRINT"{CLR}{8 DOWN}{RVS}{5 RIGHT}*{UP}
  {LEFT}*TREE{RIGHT}TUTOR*{DOWN}{LEFT}*
  {DOWN}{12 LEFT}*{RIGHT}FOR{RIGHT}TOTS
  {RIGHT}*"                          :rem 121
4 POKE36869,255:POKE52,28:POKE56,28:CLR:F
  ORI=7168TO7679:POKEI,PEEK(I+25600):NEXT
                                     :rem 97
6 FORI=7168TO7263:READN:POKEI,N:NEXT:POKE
  36879,29:V=36878:M=36876:C=30720
                                    :rem 173
8 X=0:PRINT"{CLR}{BLU}{RVS}{2 SPACES}CHOO
  SE HIGHEST SUM"SPC(10)"(2-9)"    :rem 83
10 GETF$:F=VAL(F$):IFF<2ORF>9THEN10
                                    :rem 113
12 PRINT"{CLR}{RVS}WHEN DO YOU WANT TO
   {3 SPACES}SEE FRUIT?{2 DOWN}{4 LEFT}(1
   ) ALWAYS"SPC(12)"{DOWN}(2) IF WRONG"
                                    :rem 149
14 GETI$:I=VAL(I$):IFI<1ORI>2THEN14
                                    :rem 128
16 PRINT"{CLR}":POKE8183,205:POKE8184,228
   :POKE8185,206:FORB=38903TO38905:POKEB,
   10:NEXT                          :rem 216
18 FORL=1TO10:Q=240:IFI$="2"THENI=2
                                    :rem 60
20 T=INT(RND(.)*F)+1                :rem 94
22 B=INT(RND(.)*((F+1)-T)):IFT=T1ANDB=B1T
   HEN20                            :rem 166
24 PRINT"{HOME}{2 DOWN}";:FORY=1TO20:PRIN
   T"{18 SPACES}":NEXT:PRINT"{9 SPACES}";
                                    :rem 94
26 FORZ=38796TO38883:POKEZ,2:NEXT :rem 66
28 PRINT"{HOME}{2 DOWN}{GRN}{3 SPACES}HFH
   FHF":PRINT"{2 SPACES}HJJJJJJF":PRINT"
   {SPACE}HJJJJJJJJF":PRINT" GJJJJJJJJI"
                                    :rem 97
30 PRINT" HJJJJJJJJF":PRINT" GJJJJJJJJI":
   PRINT" HJJJJJJJJF":PRINT" GJJJJJJJJI"
                                    :rem 209
32 PRINT"{2 SPACES}GJJJJJJI":PRINT"
   {3 SPACES}GJJJJI":PRINT"{4 SPACES}GJJI
   ":PRINT"{5 SPACES}{BLK}JJ{DOWN}
   {2 LEFT}JJ{DOWN}{2 LEFT}JJ{DOWN}
   {2 LEFT}JJ";                     :rem 123
34 PRINT"{DOWN}{2 LEFT}JJ{DOWN}{2 LEFT}JJ
   {DOWN}{3 LEFT}HJJF"               :rem 0
36 X=X+1:PRINT"{HOME}{8 DOWN}{16 RIGHT}
   {BLK}"T"{4 LEFT}{2 DOWN}+"B"{4 LEFT}
   {DOWN}{RVS}***":PRINTSPC(17)"?
   {2 LEFT}";                       :rem 204
38 IFI=2THEN76                      :rem 78
40 POKE7751,11:POKE7751+C,10:IFT=1THEN58
                                    :rem 84
42 POKE7860,11:POKE7860+C,10:IFT=2THEN58
                                    :rem 89
```

```
44 POKE7885,11:POKE7885+C,10:IFT=3THEN58
                                    :rem 106
46 POKE7775,11:POKE7775+C,10:IFT=4THEN58
                                    :rem 105
48 POKE7815,11:POKE7815+C,10:IFT=5THEN58
                                    :rem 98
50 POKE7820,11:POKE7820+C,10:IFT=6THEN58
                                    :rem 84
52 POKE7903,11:POKE7903+C,10:IFT=7THEN58
                                    :rem 91
54 POKE7840,11:POKE7840+C,10:IFT=8THEN58
                                    :rem 94
56 POKE7928,11:POKE7928+C,10        :rem 64
58 IFB=0THEN76                      :rem 71
60 POKE8086,0:IFB=1THEN76           :rem 220
62 POKE8078,0:IFB=2THEN76           :rem 224
64 POKE8123,0:IFB=3THEN76           :rem 218
66 POKE8150,0:IFB=4THEN76           :rem 221
68 POKE8106,0:IFB=5THEN76           :rem 225
70 POKE8146,0:IFB=6THEN76           :rem 223
72 POKE8126,0:IFB=7THEN76           :rem 224
74 POKE8152,0                       :rem 198
76 POKE198,0                        :rem 154
78 GETA$:AN=VAL(A$):IFAN<1ORAN>9THEN78
                                    :rem 93
80 PRINTAN:FORZ=1TO500:NEXT:IFAN=T+BTHEN8
   6                                :rem 17
82 PRINT"{RVS}{9 DOWN}TRY AGAIN";:POKEV,5
   :POKEM,231:FORZ=1TO200:NEXT:POKEM,225
                                    :rem 42
84 FORZ=1TO200:NEXT:POKEV,0:I=0:GOTO36
                                    :rem 176
86 PRINT"{RVS}{9 DOWN}HOORAY!{2 SPACES}"S
   PC(7)L;:T1=T:B1=B:A=7700         :rem 212
88 PRINT"{HOME}{2 SPACES}";:FORB=1TO3:PRI
   NT"{BLK}{OFF}CAE{3 LEFT}";:FORZ=1TO75:
   NEXT:PRINT"BAD{3 LEFT}";         :rem 147
90 FORZ=1TO75:NEXT:PRINT"{3 SPACES}
   {2 LEFT}{DOWN}";:NEXT            :rem 170
92 FORB=1TO2:PRINT" {RED}{UP}@{UP}
   {2 LEFT}{BLK}CAE{3 LEFT}";:IFB=2THENPO
   KE7730,6:POKE7730+C,5:GOTO96     :rem 251
94 POKE7751,10:POKE7751+C,5:POKE7728,6:PO
   KE7728+C,5:POKE7729,8:POKE7729+C,5
                                    :rem 115
96 FORZ=1TO75:NEXT:PRINT"BAD{3 LEFT}";:FO
   RZ=1TO75:NEXT:PRINT"{3 SPACES}{DOWN}
   {2 LEFT}";:NEXT                  :rem 5
98 FORB=1TO13:PRINT" {RED}@{UP}{3 LEFT}
   {BLK}CAE{3 LEFT}";:FORZ=1TO75:NEXT:PRI
   NT"BAD{3 LEFT}";                 :rem 158
100 FORZ=1TO75:NEXT:PRINT"{3 SPACES}
    {DOWN}{2 LEFT}";:NEXT           :rem 210
102 PRINT"{UP}{LEFT}";:PRINT" CA{2 LEFT}"
    ;:GOSUB124:PRINT"BA{2 LEFT}";:GOSUB12
    4:PRINT" C{LEFT}";:GOSUB124     :rem 237
104 PRINT"B{LEFT}";:GOSUB124:PRINT" ";
                                    :rem 23
106 FORB=1TO(15-L):POKEM,Q:POKEA,32:A=A+2
    2:POKEA,0:POKEA+C,2:Q=Q-5:NEXT:POKEV,
    0:NEXT                          :rem 251
108 PRINT"{HOME}{8 DOWN}{11 RIGHT}
    {11 SPACES}{DOWN}{10 LEFT}{RVS}{BLK}Y
    OU GOT 10{DOWN}{10 LEFT}APPLES IN"
                                    :rem 240
110 PRINT"{RVS}{11 RIGHT}"X"TRIES.{DOWN}
    {4 LEFT}{2 SPACES}":FORZ=1TO300:NEXT
                                    :rem 34
112 PRINT"{HOME}{21 RIGHT}{BLK}U{LEFT}";:
    FORZ=1TO75:NEXT:PRINT"B{LEFT}";:FORZ=
    1TO75:NEXT                      :rem 157
114 PRINT" {DOWN}{2 LEFT}CA{2 LEFT}";:FOR
    Z=1TO75:NEXT:PRINT"BA{2 LEFT}";:FORZ=
    1TO75:NEXT                      :rem 144
116 FORB=1TO7:PRINT"{3 SPACES}{4 LEFT}
    {DOWN}CAE {4 LEFT}";:FORZ=1TO75:NEXT:
    PRINT"BAD{3 LEFT}";:FORZ=1TO75:NEXT
                                    :rem 215
118 NEXT:PRINT"{RVS}{6 DOWN}{LEFT}{BLU}HI
    T *{DOWN}{5 LEFT}TO PLAY{DOWN}
    {7 LEFT}AGAIN."                 :rem 12
120 GETP$:IFP$<>"*"THEN120          :rem 206
122 GOTO8                           :rem 6
124 POKEV,9:FORB=1TO2:POKEM,Q:POKEA,32:A=
    A+22:POKEA,0:POKEA+C,2:Q=Q-5:FORZ=1TO
    15:NEXT                         :rem 127
126 NEXT:RETURN                     :rem 242
128 DATA24,8,106,255,255,255,126,52,60,12
    6,187,199,239,126,40,40         :rem 147
130 DATA0,0,0,15,31,48,96,192,240,120,12,
    7,3,0,0,0,0,0,0,0,240,248,12,6,3
                                    :rem 137
132 DATA15,30,48,224,192,0,0,0,192,240,24
    8,252,252,254,255,255           :rem 18
134 DATA255,255,127,127,63,31,15,3,3,7,15
    ,31,63,63,127,255               :rem 84
136 DATA255,254,254,252,252,248,224,192,2
    55,255,255,255,255,255,255,255
                                    :rem 254
138 DATA245,105,170,170,170,170,170,105
                                    :rem 169
```

## Program 2: Tree Tutor For The 64

```
100 POKE53281,1:POKE53270,PEEK(53270)OR16
    :POKE53282,5:POKE53283,2        :rem 207
110 PRINT"{CLR}{11 DOWN}{12 RIGHT}";
                                    :rem 72
120 PRINT"{BLK}*{UP}{LEFT}*TREE{2 RIGHT}T
    UTOR*{DOWN}{LEFT}*{DOWN}{13 LEFT}*
    {2 RIGHT}FOR{RIGHT}TOTS{RIGHT}*"
                                    :rem 131
130 PRINT"{4 DOWN}{11 RIGHT} LOADING
    {2 SPACES}DATA"                 :rem 0
140 POKE56334,PEEK(56334)AND254:POKE1,PEE
    K(1)AND251                      :rem 182
150 POKE56,48:CLR:FORI=12288TO16383:POKEI
    ,PEEK(I+40960):NEXT             :rem 123
160 POKE1,PEEK(1)OR4:POKE56334,PEEK(56334
    )OR1                            :rem 134
170 FORI=12288TO12383:READN:POKEI,N:NEXT:
    POKE53281,1:C=54272             :rem 126
180 FORQ=CTOC+24:POKEQ,0:NEXT:POKEC+24,15
    :POKEC+5,17:POKEC+6,245         :rem 154
190 X=0:PRINT"{CLR}{BLK}{2 SPACES}{RVS} C
    HOOSE HIGHEST SUM "SPC(10)"(2-9)"
                                    :rem 38
200 GETF$:F=VAL(F$):IFF<2ORF>9THEN200
                                    :rem 211
210 PRINT"{CLR}{4 RIGHT}{RVS}WHEN DO YOU
    {SPACE}WANT TO SEE FRUIT?{4 DOWN}"
                                    :rem 6
220 PRINTSPC(12)"(1) ALWAYS{4 DOWN}"
                                    :rem 150
230 PRINTSPC(12)"(2) IF WRONG"      :rem 159
240 GETI$:I=VAL(I$):IFI<1ORI>2THEN240
                                    :rem 226
250 PRINT"{CLR}":POKE1938,77:POKE1940,100
    :POKE1942,78                    :rem 155
260 FORB=56210TO56214:POKEB,10:NEXT
                                    :rem 82
270 FORL=1TO10:Q=240:IFI$="2"THENI=2
                                    :rem 108
```

```
280 T=INT(RND(.)*F)+1                    :rem 150
290 B=INT(RND(.)*((F+1)-T)):IFT=T1ANDB=B1
    THEN280                              :rem 21
300 PRINT"{HOME}{2 DOWN}";:FORY=1TO21:PRI
    NT"{19 SPACES}":NEXT:PRINT"
    {12 SPACES}";                        :rem 140
310 FORZ=56016TO56176:POKEZ,2:NEXT:POKE53
    272,29                               :rem 93
320 PRINT"{HOME}{2 DOWN}{GRN}{3 SPACES}HF
    HFHFHFHF":PRINT"{2 SPACES}HJJJJJJJJJJ
    F"                                   :rem 252
330 PRINT" HJJJJJJJJJJJJF":PRINT" GJJJJJJ
    JJJJJJI"                             :rem 128
340 PRINT" HJJJJJJJJJJJJF":PRINT" GJJJJJJ
    JJJJJJI"                             :rem 129
350 PRINT" HJJJJJJJJJJJJF":PRINT" GJJJJJJ
    JJJJJJI"                             :rem 130
360 PRINT"{2 SPACES}GJJJJJJJJJJI":PRINT"
    {3 SPACES}GJJJJJJJJI":PRINT"
    {4 SPACES}GJJJJJJI"                  :rem 32
370 PRINT"{5 SPACES}GJJJJI{DOWN}{5 LEFT}
    {BLK}JJJJ{DOWN}{4 LEFT}JJJJ{DOWN}
    {4 LEFT}JJJJ";                       :rem 146
380 PRINT"{DOWN}{4 LEFT}JJJJ{DOWN}
    {4 LEFT}JJJJ{DOWN}{5 LEFT}HJJJJF"
                                         :rem 158
390 X=X+1:PRINT"{HOME}{8 DOWN}{16 RIGHT}
    {BLK}"T"{4 LEFT}{2 DOWN}+"B"{4 LEFT}
    {DOWN}***"                           :rem 143
400 PRINTSPC(17)"?{2 LEFT}";             :rem 184
410 IFI=2THEN600                         :rem 161
420 POKE1151,11:POKE1151+C,10:IFT=1THEN51
    0                                    :rem 151
430 POKE1226,11:POKE1226+C,10:IFT=2THEN51
    0                                    :rem 159
440 POKE1396,11:POKE1396+C,10:IFT=3THEN51
    0                                    :rem 177
450 POKE1196,11:POKE1196+C,10:IFT=4THEN51
    0                                    :rem 175
460 POKE1269,11:POKE1269+C,10:IFT=5THEN51
    0                                    :rem 179
470 POKE1278,11:POKE1278+C,10:IFT=6THEN51
    0                                    :rem 181
480 POKE1429,11:POKE1429+C,10:IFT=7THEN51
    0                                    :rem 179
490 POKE1314,11:POKE1314+C,10:IFT=8THEN51
    0                                    :rem 167
500 POKE1474,11:POKE1474+C,10:IFT=5THEN51
    0                                    :rem 170
510 IFB=0THEN600                         :rem 153
520 POKE1762,0:IFB=1THEN600              :rem 48
530 POKE1747,0:IFB=2THEN600              :rem 53
540 POKE1829,0:IFB=3THEN600              :rem 56
550 POKE1878,0:IFB=4THEN600              :rem 62
560 POKE1798,0:IFB=5THEN600              :rem 65
570 POKE1871,0:IFB=6THEN600              :rem 59
580 POKE1834,0:IFB=7THEN600              :rem 60
590 POKE1882,0                           :rem 252
600 POKE198,0:                           :rem 253
610 GETA$:AN=VAL(A$):IFAN<1ORAN>9THEN610
                                         :rem 173
620 PRINT(AN):FORZ=1TO500:NEXT:IFAN=T+BTH
    EN660                                :rem 192
630 PRINT"{RVS}{9 DOWN}TRY AGAIN{OFF}";:P
    OKEC,5:POKEC+1,5:FORZ=1TO200:NEXT
                                         :rem 78
640 POKEC+4,33:FORZ=1TO200:NEXT:POKEC+4,3
    2                                    :rem 48
650 I=0:GOTO390                          :rem 96
660 PRINT"{RVS}{8 DOWN}{RVS}{DOWN}HOORAY!
    {2 SPACES}"SPC(7)L;:T1=T:B1=B:A=1060
                                         :rem 13
670 PRINT"{HOME}{4 RIGHT}";:FORB=1TO3:PRI
    NT"{BLK}{OFF}CAE{3 LEFT}";:FORZ=1TO75
    :NEXT:PRINT"BAD{3 LEFT}";            :rem 52
680 FORZ=1TO75:NEXT:PRINT"{3 SPACES}
    {2 LEFT}{DOWN}";:NEXT                :rem 223
690 FORB=1TO3:PRINT" {RED}{UP}@{UP}
    {2 LEFT}{BLK}CAE{3 LEFT}";:IFB=2THENP
    OKE1112,6:POKE1112+C,5:GOTO720:rem 66
700 POKE1151,10:POKE1151+C,5:POKE1110,6:P
    OKE1110+C,5:POKE1111,8               :rem 53
710 POKE1111+C,5                         :rem 90
720 FORZ=1TO75:NEXT:PRINT"BAD{3 LEFT}";:F
    ORZ=1TO75:NEXT:PRINT"{3 SPACES}{DOWN}
    {2 LEFT}";:NEXT                      :rem 47
730 FORB=1TO27:PRINT" {RED}@{UP}{3 LEFT}
    {SPACE}{BLK}CAE{3 LEFT}";:FORZ=1TO75:
    NEXT:PRINT"BAD{3 LEFT}";             :rem 204
740 FORZ=1TO45:NEXT:PRINT"{3 SPACES}
    {DOWN}{2 LEFT}";:NEXT                :rem 217
750 PRINT"{UP}{LEFT}";:PRINT" CA{2 LEFT}"
    ;:GOSUB930:PRINT"BA{2 LEFT}";:GOSUB93
    0                                    :rem 137
760 PRINT"{2 SPACES}{LEFT}";:GOSUB930
                                         :rem 156
770 PRINT"{2 SPACES}{LEFT} ";:GOSUB930:PR
    INT"{DOWN}{3 LEFT}{3 SPACES}":rem 144
780 POKEC+4,17                            :rem 4
790 FORB=1TO14-L:POKEC+1,Q/4:POKEC,Q/4:PO
    KEA,32:A=A+40:POKEA,0:POKEA+C,2:Q=Q-5
                                         :rem 230
800 NEXT:POKEC+4,16:NEXT                 :rem 238
810 PRINT"{HOME}{8 DOWN}{22 RIGHT}
    {11 SPACES}{DOWN}{10 LEFT}{RVS}{BLK}
    {SPACE}YOU GOT 10";                  :rem 219
820 PRINT"{RVS}{DOWN}{11 LEFT}{3 SPACES}A
    PPLE{3 SPACES}";                     :rem 250
830 PRINT"{RVS}{DOWN}{11 LEFT}IN"X"{LEFT}
     TRIES.";:FORZ=1TO300:NEXT           :rem 166
840 PRINT"{OFF}{DOWN}{11 LEFT}{11 SPACES}
    "                                    :rem 207
850 PRINT"{HOME}{31 RIGHT}{BLK}U{LEFT}";:
    FORZ=1TO75:NEXT:PRINT"B{LEFT}";
                                         :rem 87
860 FORZ=1TO75:NEXT                      :rem 213
870 PRINT" {DOWN}{2 LEFT}CA{2 LEFT}";:FOR
    Z=1TO75:NEXT:PRINT"BA{2 LEFT}";:FORZ=
    1TO75:NEXT                           :rem 153
880 FORB=1TO7:PRINT"{3 SPACES}{4 LEFT}
    {DOWN}CAE {4 LEFT}";:FORZ=1TO75:NEXT:
    PRINT"BAD{3 LEFT}";                  :rem 110
890 FORZ=1TO75:NEXT                      :rem 216
900 NEXT:PRINT"{RVS}{6 DOWN}{LEFT}{BLU}HI
    T *{DOWN}{5 LEFT}TO PLAY{DOWN}
    {7 LEFT}AGAIN."                      :rem 11
910 GETP$:IFP$<>"*"THEN910               :rem 220
920 POKE53272,21:GOTO190                 :rem 105
930 FORB=1TO2::POKEA,32:A=A+40:POKEA,0
                                         :rem 11
940 FORZ=1TO15:NEXT:NEXT:RETURN          :rem 97
950 DATA24,8,106,255,255,255,126,52,60,12
    6,187,199,239,126,40,40              :rem 150
960 DATA0,0,0,15,31,48,96,192,240,120,12,
    7,3,0,0,0,0,0,0,240,248,12,6,3
                                         :rem 148
970 DATA15,30,48,224,192,0,0,0,192,240,24
    8,252,252,254,255,255                :rem 28
980 DATA255,255,127,127,63,31,15,3,3,7,15
    ,31,63,63,127,255                    :rem 93
990 DATA255,254,254,252,252,248,224,192,2
    55,255,255,255,255,255,255,255 :rem 6
1000 DATA245,105,170,170,170,170,170,105
                                         :rem 206
```

# VICreations

*(Article on page 124.)*

**BEFORE TYPING...**

**Before typing in programs, please refer to "How To Type COMPUTE!'s Gazette Programs," "A Beginner's Guide To Typing In Programs," and "The Automatic Proofreader" that appear before the Program Listings.**

```
1 REM REMEMBER TO CHNG{2 SPACES}"N", LINE
   #900 WHEN{3 SPACES}ADDING/DELETING DAT
  A{2 SPACES}STATEMENTS              :rem 63
10 PRINT"{CLR}{DOWN} DO YOU WISH TO SEE":
   PRINT" THE MENU?":PRINT"{DOWN} (PRESS
   {SPACE}Y OR N)"                   :rem 195
20 GETY$:IFY$=""THEN20                :rem 23
21 IFY$="Y"THEN800                    :rem 16
25 PRINT"{2 DOWN} LOADING DATABASE...":PR
   INT" PLEASE WAIT.":GOSUB900        :rem 146
30 S=0:PRINT"{CLR}{DOWN}ENTER SEARCH"SPC(
   4)"PRESS":PRINT"ARGUMENT"SPC(8)"F-KEY
   {2 DOWN}"                         :rem 225
31 PRINT"{DOWN}ARTICLE SUBJECT{3 SPACES}1
   {DOWN}":PRINT"ARTICLE NAME"SPC(6)"3
   {DOWN}"                           :rem 193
32 PRINT"MAGAZINE NAME"SPC(5)"5"      :rem 136
33 PRINT"{DOWN}TYPE OF COMPUTER{2 SPACES}
   7":PRINT"{DOWN}END PROGRAM"SPC(7)"8"
                                     :rem 186
40 GETX$                             :rem 192
41 IFX$="{F1}"THENS=3                :rem 104
42 IFX$="{F3}"THENS=2                :rem 105
43 IFX$="{F5}"THENS=1                :rem 106
44 IFX$="{F7}"THENS=6                :rem 113
45 IFX$="{F8}"THENPRINT"{CLR}{DOWN} END P
   ROGRAM":CLR:END                   :rem 64
46 IFS=0THEN40                       :rem 76
60 PRINT"{CLR}{DOWN} ENTER SUBJECT OF
   {DOWN}":PRINT" SEARCH:{2 DOWN}":INPUTS
   $                                 :rem 109
65 FORZ=1TON:IFA$(Z,S)=S$THENGOSUB300
                                     :rem 89
70 NEXTZ                              :rem 0
75 PRINT"{CLR}{DOWN} END OF DATA OR{DOWN}
   ":PRINT" SUBJECT NOT FOUND{DOWN}"
                                     :rem 178
76 PRINT" (CHECK SPELLING){2 DOWN}"
                                     :rem 109
77 GOSUB600:GOTO30                    :rem 90
300 PRINT"{CLR} SUBJECT FOUND:{2 DOWN}":P
    RINT"MAGAZINE:":PRINTA$(Z,1)     :rem 164
301 PRINT"{DOWN}ARTICLE:":PRINTA$(Z,2)
                                     :rem 233
302 PRINT"{DOWN}SUBJECT:":PRINTA$(Z,3):PR
    INT"{DOWN}DATE:":PRINTA$(Z,4):PRINT"
    {DOWN}PAGE NO.:"                 :rem 224
303 PRINTA$(Z,5):PRINT"{DOWN}COMPUTER:":P
    RINTA$(Z,6)                      :rem 146
305 GOSUB600:RETURN                  :rem 200
600 PRINT"{DOWN} (PRESS RETURN)"      :rem 54
601 GETY$:IFY$=""THEN601             :rem 129
602 RETURN                           :rem 120
800 PRINT"{CLR}{DOWN} RECORD FORMAT:
    {DOWN}":PRINT"1) MAGAZINE NAME{DOWN}"
    :PRINT"2) ARTICLE NAME{DOWN}"    :rem 95
801 PRINT"3) SUBJECT OF ARTICLE{DOWN}":PR
    INT"4) MONTH.YEAR{DOWN}":PRINT"5) PAG
    E NO.{DOWN}"                     :rem 64
802 PRINT"6) COMPUTER TYPE":PRINT"
    {2 DOWN} SEPARATE EACH ENTRY{DOWN}":P
    RINT" BY A COMMA{DOWN}"          :rem 26
803 GOSUB600                         :rem 177
804 PRINT"{CLR}{DOWN}WHEN PROMPTED TO":PR
    INT"{DOWN}ENTER SEARCH ARGUMENT,"
                                     :rem 48
805 PRINT"PRESS F-KEY FOR":PRINT"{DOWN}DE
    SIRED FUNCTION.{2 DOWN}"         :rem 16
806 GOSUB600                         :rem 180
807 PRINT"{CLR}{DOWN}WHEN PROMPTED TO
    {DOWN}":PRINT"ENTER SUBJECT OF{DOWN}"
                                     :rem 164
808 PRINT"SEARCH, ENTER NAME,{DOWN}":PRIN
    T"THEN PRESS RETURN.{2 DOWN}":GOSUB60
    0                                :rem 118
809 GOTO10                           :rem 59
900 N=5:DIMA$(N,6):FORR=1TON:FORC=1TO6:RE
    ADA$(R,C):NEXTC:NEXTR:RETURN     :rem 190
901 DATA GAZETTE,SPEEDSCRIPT,WORD PROCESS
    ING,1.84,38,VIC/64               :rem 212
902 DATA GAZETTE,CAVE-IN,GAME,1.84,80,VIC
                                     :rem 183
903 DATA GAZETTE,ELECTRONIC NOTEPAD,UTILI
    TY,1.84,112,64                   :rem 150
904 DATA GAZETTE,HORIZONS 64,GENERAL,1.84
    ,136,64                          :rem 90
905 DATA GAZETTE,MLX,M/L,1.84,171,VIC/64
                                     :rem 64
```

# Trenchfire

*(Article on page 52.)*

## Program 1: 64 Version

```
5 PRINT"{CLR}":POKE214,10:PRINT:PRINTTAB(
  13)"{RVS}TRENCH FIRE":FORJ=1TO999:NEXT
                                     :rem 32
10 POKE53281,11:POKE53282,12:POKE53283,13
   :Z=53270:POKEZ,PEEK(Z)OR16        :rem 180
20 POKE52,56:POKE56,56:CLR:POKE56334,PEEK
   (56334)AND254:POKE1,PEEK(1)AND251
                                     :rem 112
30 IFPEEK(14336)<>60THENFORI=0TO511:POKEI
   +14336,PEEK(I+53248):NEXT         :rem 194
35 POKE1,PEEK(1)OR4:POKE53280,0:V=53248
                                     :rem 10
40 POKE56334,PEEK(56334)OR1:POKE53272,(PE
   EK(53272)AND240)+14:POKE54296,0:rem 59
50 FORJ=0TO7:POKE14336+27*8+J,170:POKE143
   36+29*8+J,85:READQ:POKE14848+J,Q
                                     :rem 241
55 POKE14336+28*8+J,255:POKE14336+31*8+J,
   255-PEEK(14336+46*8+J):NEXT       :rem 239
57 DATA255,243,243,192,243,243,255,255
                                     :rem 142
60 FORJ=0TO15:READQ:POKE14856+J,Q:NEXT:DA
   TA254,254,252,240,232,156,235,247
                                     :rem 231
61 DATA127,127,63,15,23,57,215,239
                                     :rem 191
85 D=55296-1024:FORJ=1024TO1503:IFRND(1)>
   .9THENPOKEJ,31:POKEJ+D,8:J=J+1:rem 193
87 POKEJ,28:POKEJ+D,8:NEXT:Q$="SCORE"
                                     :rem 41
88 FORJ=1TO5:POKE1023+J,ASC(MID$(Q$,J,1))
   -64:POKE1023+J+D,0:NEXT:POKE1029,58
                                     :rem 195
```

```
89 FORJ=1029TO1035:POKEJ+D,0:NEXT:NS=3:GO
   SUB5000:SC=0:GOSUB5010          :rem 149
90 FORJ=1TO10:READQ:POKEQ-47,64:NEXT
                                   :rem 20
95 DATA1087,1090,1165,1212,1251,1290,1331
   ,1297,1371,1413                  :rem 6
110 FORJ=1TO16                     :rem 59
113 G=0                            :rem 73
115 IFJ>5THENIFJ/2=INT(J/2)THENF=F+40:G=4
   0                               :rem 25
120 READC                          :rem 242
130 T=1+J                          :rem 203
140 FORX=0TO18-T:M=1344+J*40+X:N=1344+J*4
   0+39-X                          :rem 148
150 Q=M:GOSUB500:Q=N:GOSUB500      :rem 37
160 NEXTX                          :rem 46
170 FORI=1364+J*40TO1444+J*40+FSTEP40
                                   :rem 90
180 Q=I-T-1:GOSUB500:Q=I+T:GOSUB500
                                   :rem 125
190 NEXT:I=I-40                    :rem 115
200 FORH=I-T-1 TOI+T               :rem 145
210 Q=H:GOSUB500:Q=H-G:GOSUB500:NEXT:NEXT
   :GOTO700                        :rem 135
300 DATA32,27,29,32,27,29,29,32,32,27,27,
   29,29,29,32,32                  :rem 197
500 IFQ<2024THENPOKEQ,C:POKEQ+D,8  :rem 36
501 RETURN                         :rem 118
700 IFPEEK(832)=33THEN800          :rem 163
705 FORJ=0TO2:FORI=0TO62:READQ:POKE832+J*
   64+I,Q:NEXT:NEXT                :rem 138
710 DATA33,0,0,64,128,0,140,64,0,158,64,0
   ,243,192,0,158,64,0,140,64,0    :rem 80
720 DATA64,128,0,33,0,0,0,0,0,0,0,0,0,0,0
   ,0,0,0,0,0,0                    :rem 238
730 DATA0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
   ,0,0,0,0                        :rem 20
735 DATA0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
   ,0,0,0,0                        :rem 25
740 DATA0,8,0,0,8,0,0,28,0,0,28,0,1,255,1
   92,3,255,224,127,127,127        :rem 135
750 DATA30,62,60,3,99,96,1,193,192,0,0,0,
   0,0,0,0,0,0,0,0,0               :rem 21
770 DATA0,128,0,32,32,0,0,2,0,10,128,2,40
   ,2,0,152,130,128,96,128,130     :rem 8
780 DATA128,150,136,0,1,215,64,215,64,128
   ,125,10,0,20,130,0,0,128,96,105,0
                                   :rem 67
785 DATA0,128,128,130,128,128,41,0,64,6,0
   ,64,10,2,128,128,8,2,0,8,0      :rem 248
790 FORJ=15232TO15296:POKEJ,0:NEXT:FORJ=1
   5253TO15273:READQ:POKEJ,Q:NEXT
                                   :rem 218
792 DATA0,112,0,0,136,0,1,36,0,1,116,0,1,
   36,0,0,136,0,0,112,0            :rem 149
800 V=53248:POKEV+21,0:POKE2040,14:POKEV+
   39,1:POKEV+28,0:POKEV+46,0      :rem 173
805 POKE2047,238:POKEV+16,0:POKEV+23,0:PO
   KEV+29,0:FORJ=1TO6:POKE2040+J,13
                                   :rem 81
810 POKEV+39+J,INT(RND(1)*3)+5:NEXT
                                   :rem 46
815 FORJ=54272TO54299:POKEJ,0:NEXT:rem 72
900 IFV<>53248THEN STOP            :rem 115
910 X=160:POKEV,X:POKEV+1,200:POKEV+29,1:
   POKEV+23,1:POKEV+21,1:LV=1:W=4:E=8
                                   :rem 42
920 POKE2047,238:IFPEEK(49172)<>206THENGO
   SUB2000                         :rem 41
925 POKE49152,6:POKE49153,6:POKEV+46,0
                                   :rem 183
930 SYS53000                       :rem 147
933 S=54272:POKES+24,15:POKES+5,1:POKES+6
   ,128:POKES+4,129:POKES+1,12     :rem 249
980 A=0                            :rem 79
990 IFA>11+LV*3THEN6000            :rem 49
995 IF(PEEK(56321)AND16)=0THENGOSUB6200
                                   :rem 28
1000 FORJ=1TO6:IFRND(1)<(1-LV/20)ORX(J)<>
   0THEN1050                       :rem 185
1005 POKE2040+J,13:X(J)=180:Y(J)=120:A=A+
   1                               :rem 22
1010 M%(J)=RND(1)*(5+LV/2)-2-LV/4:POKEV+3
   9+J,INT(RND(1)*3+0):R(J)=W      :rem 237
1020 POKEV+J*2,X(J):POKEV+1+J*2,Y(J):POKE
   V+21,PEEK(V+21)OR(2↑J):GOTO1080
                                   :rem 31
1050 NEXT                          :rem 5
1080 FORJ=1TO6:IFX(J)=0THENNEXT:GOTO1110
                                   :rem 14
1085 IFY(J)<160ORR(J)=ETHEN1095    :rem 50
1090 POKEV+29,PEEK(V+29)OR(2↑J):POKEV+23,
   PEEK(V+23)OR(2↑J):R(J)=E:M%(J)=M%(J)
   *2                              :rem 120
1092 FORN=S+7TOS+13:POKEN,0:NEXT   :rem 4
1093 POKES+24,15:POKES+12,207:POKES+13,0:
   POKES+8,10:POKES+11,33          :rem 175
1095 Y(J)=Y(J)+R(J):X(J)=X(J)+M%(J)
                                   :rem 161
1100 POKEV+J*2,X(J):POKEV+1+J*2,Y(J)
                                   :rem 67
1104 IFY(J)<222ANDPEEK(2040+J)=13THEN1107
                                   :rem 56
1105 X(J)=0:POKEV+21,PEEK(V+21)AND(255-2↑
   J)                              :rem 193
1106 POKEV+29,PEEK(V+29)AND(255-2↑J):POKE
   V+23,PEEK(V+23)AND(255-2↑J)     :rem 214
1107 NEXT                          :rem 8
1110 IF(PEEK(V+30)AND1)<>1THEN990:rem 125
1111 POKEV+21,1:POKES+11,0         :rem 217
1120 POKES+1,4:FORI=1TO10:POKES+24,15-ABS
   (6-I):FORJ=0TO7:POKEV+39,J      :rem 59
1130 FORH=1TO10:NEXT:NEXT:NEXT:POKES+24,1
   5:POKES+1,12                    :rem 87
1140 GOSUB5020:K=PEEK(V+30):GOTO1000
                                   :rem 93
2000 J=0:READT:IFT<999THENSTOP     :rem 156
2002 READQ:IFQ>=0THENPOKEJ+T,Q:J=J+1:GOTO
   2002                            :rem 56
2004 IFQ<>-99 THENGOTO2000         :rem 233
2006 RETURN                        :rem 168
2020 DATA49172,206,0,192,173,0,192,240,3,
   76,100,192                      :rem 247
2030 DATA173,1,192,141,0,192,162,1,254,32
   ,208,189,32,208,41,255,233,253,48,9
                                   :rem 202
2040 DATA222,32,208,222,32,208,222,32,208
                                   :rem 255
2050 DATA232,224,4,208,230,76,100,192,-1
                                   :rem 203
2080 DATA49252,173,1,220,41,4,208,15,173,
   0,208                           :rem 0
2090 DATA233,105,48,8,173,0,208,233,4,141
   ,0,208                          :rem 44
3000 DATA173,1,220,41,8,208,15,173,0,208
                                   :rem 203
3010 DATA233,218,16,8,173,0,208,105,4,141
   ,0,208,76,49,234,-1             :rem 164
3050 DATA53000,120,169,20,141,20,3,169,19
   2,141,21,3,88,96,-99            :rem 232
5000 IFNS>6THENNS=6:SC=SC+375:GOSUB5010
                                   :rem 153
```

```
5001 FORJ=1062TO1065-NS*3STEP-3:POKEJ,65:
     POKEJ+1,66:POKEJ+D,0:POKEJ+1+D,0:NEX
     T                              :rem 245
5002 RETURN                         :rem 167
5010 S$=STR$(SC):FORJ=1TOLEN(S$):POKE1029
     +J,ASC(MID$(S$,J,1)):NEXT      :rem 177
5013 O=INT(SC/1000):IFO>PTHENP=O:NS=NS+1:
     GOSUB5000:GOSUB5050            :rem 224
5016 RETURN                         :rem 172
5020 FORI=1TO2:POKE1064-NS*3+I,28:NEXT:NS
     =NS-1:IFNS>=0THENRETURN        :rem 87
5030 PRINT"{CLR}{10 DOWN}{14 RIGHT}GAME O
     VER"                           :rem 194
5031 POKE56334,PEEK(56334)AND254    :rem 18
5032 POKE788,49:POKE789,234         :rem 15
5033 POKE56334,PEEK(56334)OR1:POKE198,0:P
     OKE53249,0:POKE54296,0         :rem 122
5034 FORI=1TO2000:NEXT:SYS2048      :rem 79
5050 FORN=S+14TOS+20:POKEN,0:NEXT   :rem 46
5051 POKES+24,15:POKES+19,61:POKES+15,30:
     POKES+18,17:RETURN             :rem 46
6000 POKEV+21,1:FORJ=5TO1STEP-1:POKE49153
     ,J:FORI=132-J*20TO142-J*20:POKES+1,I
                                    :rem 112
6010 FORH=1TO40:NEXT:NEXT:NEXT:SC=SC+LV*7
     5:GOSUB5010:FORJ=1TO3000:NEXT:LV=LV+
     1                              :rem 43
6020 FORJ=2TO6:POKE49153,J:FORI=142-J*20T
     O132-J*20STEP-1:POKES+1,I:FORH=1TO40
                                    :rem 168
6030 NEXT:NEXT:NEXT:W=W+1:E=E+2     :rem 217
6040 GOTO980                        :rem 164
6200 M%=(166-PEEK(V))/7:POKEV+14,PEEK(V)+
     12+M%:POKEV+15,195:I=3         :rem 68
6210 POKEV+21,PEEK(V+21)OR128:FORJ=195TO1
     30+LV*2STEP-3                  :rem 225
6220 POKEV+15,J:I=I+1:IFI=5THENI=0:POKEV+
     14,PEEK(V+14)+M%               :rem 152
6230 K=PEEK(V+30):IF(KAND128)=0THENNEXT:G
     OTO6300                        :rem 151
6234 FORN=S+14TOS+20:POKEN,0:NEXT   :rem 51
6235 POKES+24,15:POKES+19,14:POKES+15,2:P
     OKES+18,129                    :rem 26
6240 FORJ=1TO6:IF(KAND(2↑J))=0THENNEXT
                                    :rem 130
6250 POKE2040+J,15:SC=SC+25:GOSUB5010
                                    :rem 126
6300 POKEV+21,PEEK(V+21)AND127:RETURN
                                    :rem 233
```

## Program 2: VIC Version

*(Use only with short MLX version and 8K expander.)*

```
4352 :169,147,032,210,255,169,214
4358 :008,141,015,144,169,032,003
4364 :162,000,157,242,030,232,067
4370 :224,242,208,248,169,032,117
4376 :162,000,157,228,031,232,066
4382 :224,022,208,248,162,000,126
4388 :189,162,017,157,242,150,185
4394 :232,224,242,208,245,162,075
4400 :000,189,148,018,157,228,020
4406 :151,232,224,022,208,245,112
4412 :169,000,141,060,003,169,090
4418 :064,141,046,145,169,141,004
4424 :141,020,003,169,017,141,051
4430 :021,003,169,192,141,046,138
4436 :145,076,170,018,162,000,143
4442 :189,242,150,041,015,201,160
4448 :000,240,009,168,200,152,097
4454 :201,008,208,002,169,005,183
4460 :157,242,150,232,224,242,075
4466 :208,230,162,000,189,228,107
4472 :151,041,015,168,200,152,079
4478 :201,008,208,002,169,005,207
4484 :157,228,151,232,224,022,122
4490 :208,234,096,238,060,003,209
4496 :172,060,003,192,010,208,021
4502 :008,032,088,017,160,000,199
4508 :141,060,003,076,191,234,093
4514 :007,007,007,007,007,007,204
4520 :007,007,007,007,000,000,196
4526 :007,007,007,007,007,007,216
4532 :007,007,007,007,006,006,220
4538 :006,006,006,006,006,006,222
4544 :006,007,000,000,007,006,218
4550 :006,006,006,006,006,006,234
4556 :006,006,005,005,005,005,236
4562 :005,005,005,005,006,007,243
4568 :007,007,007,006,005,005,253
4574 :005,005,005,005,005,005,252
4580 :007,007,007,007,007,007,014
4586 :007,005,006,006,006,006,014
4592 :006,006,005,007,007,007,022
4598 :007,007,007,007,006,006,030
4604 :006,006,006,006,007,005,032
4610 :006,006,006,006,006,006,038
4616 :005,007,006,006,006,006,044
4622 :006,006,006,006,006,006,050
4628 :006,006,007,005,005,005,054
4634 :005,005,005,005,005,007,058
4640 :006,006,006,006,006,006,068
4646 :005,005,005,005,005,006,069
4652 :007,005,005,005,005,005,076
4658 :005,005,005,007,006,005,083
4664 :005,005,005,005,005,005,086
4670 :005,005,005,006,007,007,097
4676 :007,007,007,007,007,007,110
4682 :007,007,006,005,005,005,109
4688 :005,005,007,007,007,007,118
4694 :005,006,007,007,007,007,125
4700 :007,007,007,007,007,007,134
4706 :006,005,007,007,007,007,137
4712 :007,007,007,007,005,006,143
4718 :006,006,006,006,006,006,146
4724 :006,006,006,006,006,005,151
4730 :007,007,007,007,006,006,162
4736 :007,007,005,006,006,006,165
4742 :006,006,006,006,006,006,170
4748 :006,006,006,005,007,007,177
4754 :006,006,006,006,007,007,184
4760 :005,005,005,005,005,005,182
4766 :005,005,005,005,005,005,188
4772 :005,005,007,007,006,006,200
4778 :169,010,141,065,003,169,215
4784 :008,141,066,003,169,014,065
4790 :141,067,003,169,004,141,195
4796 :069,003,169,002,141,070,130
4802 :003,169,008,141,071,003,077
4808 :169,015,141,014,144,169,084
4814 :170,141,013,144,169,050,125
4820 :141,128,022,162,000,169,066
4826 :037,157,000,030,232,224,130
4832 :242,208,248,162,000,189,249
4838 :000,128,157,000,028,232,007
4844 :224,000,208,245,189,000,078
4850 :129,157,000,029,232,224,245
4856 :000,208,245,189,131,023,020
4862 :157,000,029,232,224,064,192
4868 :208,245,169,255,141,005,003
4874 :144,169,000,141,072,003,027
4880 :141,073,003,032,118,019,146
```

```
4886 :162,000,169,037,157,000,035
4892 :030,232,224,242,208,248,188
4898 :162,000,189,086,023,142,124
4904 :062,003,170,169,046,157,135
4910 :000,030,169,001,157,000,147
4916 :150,174,062,003,232,224,129
4922 :015,208,231,162,000,189,095
4928 :079,023,157,000,030,169,010
4934 :004,157,000,150,232,224,069
4940 :007,208,240,169,003,141,076
4946 :074,003,032,078,021,162,196
4952 :005,169,005,141,061,003,216
4958 :170,169,033,157,212,031,098
4964 :169,232,141,075,003,169,121
4970 :003,141,076,003,169,000,242
4976 :141,078,003,076,195,019,112
4982 :162,006,160,006,024,032,252
4988 :240,255,162,000,189,099,045
4994 :023,032,210,255,232,224,082
5000 :011,208,245,162,008,160,162
5006 :001,024,032,240,255,162,088
5012 :000,189,110,023,032,210,200
5018 :255,232,224,021,208,245,059
5024 :169,000,141,072,003,141,174
5030 :073,003,173,017,145,041,106
5036 :032,240,249,032,010,022,245
5042 :032,010,022,032,010,022,050
5048 :032,010,022,173,017,145,071
5054 :041,032,208,249,096,162,210
5060 :000,142,068,003,032,065,250
5066 :020,174,068,003,032,025,012
5072 :022,032,117,022,162,170,221
5078 :142,013,144,032,243,019,039
5084 :173,135,003,201,000,240,204
5090 :003,032,196,020,238,068,015
5096 :003,174,068,003,224,003,195
5102 :208,213,076,195,019,173,098
5108 :141,002,041,001,201,001,119
5114 :240,054,165,198,201,000,084
5120 :240,047,169,000,133,198,019
5126 :173,119,002,201,133,208,074
5132 :006,169,100,141,128,022,066
5138 :096,201,134,208,006,169,064
5144 :050,141,128,022,096,201,150
5150 :135,208,006,169,025,141,202
5156 :128,022,096,201,136,208,059
5162 :006,169,010,141,128,022,006
5168 :096,096,120,032,159,255,038
5174 :173,141,002,041,001,201,101
5180 :001,240,244,088,096,169,130
5186 :127,141,034,145,173,032,206
5192 :145,162,255,142,034,145,187
5198 :041,128,208,021,174,061,199
5204 :003,169,032,157,212,031,176
5210 :232,224,010,144,002,162,096
5216 :009,142,061,003,076,221,096
5222 :020,173,017,145,041,016,002
5228 :208,021,174,061,003,169,232
5234 :032,157,212,031,202,224,204
5240 :255,208,002,162,000,142,121
5246 :061,003,076,221,020,173,168
5252 :135,003,201,001,240,082,026
5258 :173,017,145,041,032,208,242
5264 :075,173,061,003,010,010,220
5270 :010,141,133,003,162,001,088
5276 :142,135,003,174,133,003,234
5282 :024,105,008,141,134,003,065
5288 :189,255,022,170,189,242,211
5294 :030,201,032,240,008,169,086
5300 :000,141,135,003,076,109,132
5306 :021,169,035,157,242,030,072
5312 :142,132,003,096,174,132,103
5318 :003,169,032,157,242,030,063
5324 :238,133,003,174,133,003,120
5330 :236,134,003,144,209,162,074
5336 :000,142,135,003,096,189,013
5342 :212,031,201,032,208,008,146
5348 :169,033,157,212,031,076,138
5354 :131,020,206,074,003,032,188
5360 :078,021,032,210,022,174,009
5366 :061,003,169,032,157,212,112
5372 :031,169,033,162,005,157,041
5378 :212,031,142,061,003,173,112
5384 :065,003,170,169,032,157,092
5390 :008,031,173,066,003,170,209
5396 :169,032,157,008,031,173,078
5402 :067,003,170,169,032,157,112
5408 :008,031,169,010,141,065,200
5414 :003,169,008,141,066,003,172
5420 :169,014,141,067,003,169,095
5426 :004,141,069,003,169,002,182
5432 :141,070,003,169,008,141,076
5438 :071,003,173,074,003,201,075
5444 :000,240,001,096,032,118,043
5450 :019,076,022,019,162,007,123
5456 :169,037,236,074,003,240,071
5462 :007,157,014,030,202,076,060
5468 :082,021,224,000,240,010,157
5474 :169,036,157,014,030,202,194
5480 :224,000,208,248,096,032,144
5486 :243,021,142,062,003,189,002
5492 :065,003,170,169,038,157,206
5498 :008,031,160,255,140,011,215
5504 :144,032,241,022,136,192,127
5510 :080,208,245,169,000,141,209
5516 :011,144,032,132,022,174,143
5522 :062,003,189,065,003,170,126
5528 :169,032,157,008,031,174,211
5534 :062,003,165,162,041,007,086
5540 :024,105,001,157,069,003,011
5546 :024,105,005,157,065,003,017
5552 :238,078,003,173,078,003,237
5558 :201,025,240,001,096,206,183
5564 :128,022,169,000,141,078,214
5570 :003,173,128,022,201,007,216
5576 :208,003,238,128,022,120,151
5582 :169,234,141,149,017,141,033
5588 :150,017,088,032,229,022,238
5594 :032,010,022,032,010,022,090
5600 :032,010,022,032,235,022,065
5606 :120,169,208,141,149,017,010
5612 :169,008,141,150,017,088,041
5618 :096,138,056,233,022,205,224
5624 :065,003,208,003,162,000,177
5630 :096,205,066,003,208,003,067
5636 :162,001,096,162,002,096,011
5642 :162,000,160,000,200,192,212
5648 :000,208,251,232,224,000,163
5654 :208,246,096,142,063,003,012
5660 :189,065,003,170,169,032,144
5666 :157,008,031,174,063,003,214
5672 :173,061,003,221,069,003,058
5678 :176,009,222,065,003,222,231
5684 :069,003,076,063,022,254,027
5690 :065,003,254,069,003,189,129
5696 :065,003,024,105,022,170,197
5702 :224,220,144,020,165,162,237
5708 :041,007,174,063,003,024,132
5714 :105,001,157,069,003,024,185
5720 :105,005,170,076,104,022,058
5726 :189,008,031,201,033,208,252
5732 :003,076,236,020,169,034,126
```

```
5738 :157,008,031,138,174,063,165
5744 :003,157,065,003,096,162,086
5750 :000,160,000,200,192,000,158
5756 :208,251,232,224,030,208,253
5762 :246,096,173,072,003,024,232
5768 :105,010,141,072,003,144,099
5774 :003,238,073,003,162,000,109
5780 :160,006,024,032,240,255,097
5786 :173,073,003,174,072,003,140
5792 :032,205,221,173,072,003,098
5798 :205,075,003,208,038,173,100
5804 :073,003,205,076,003,208,228
5810 :030,173,075,003,024,105,076
5816 :232,141,075,003,173,076,116
5822 :003,105,003,141,076,003,009
5828 :173,074,003,201,007,240,126
5834 :006,238,074,003,032,078,121
5840 :021,096,162,255,142,013,129
5846 :144,032,241,022,202,224,055
5852 :080,208,245,169,170,141,209
5858 :013,144,096,162,220,142,235
5864 :013,144,096,162,170,142,191
5870 :013,144,096,140,077,003,199
5876 :160,000,200,192,000,208,236
5882 :251,172,077,003,096,204,029
5888 :182,161,139,118,096,075,003
5894 :054,205,183,162,140,119,101
5900 :097,076,054,206,184,163,024
5906 :141,119,098,076,054,207,201
5912 :185,163,142,120,098,076,040
5918 :054,208,186,164,142,120,136
5924 :098,076,054,209,187,165,057
5930 :143,121,099,077,055,210,235
5936 :188,166,143,121,099,077,074
5942 :055,211,189,166,144,122,173
5948 :099,077,055,212,190,167,092
5954 :145,122,100,077,055,213,010
5960 :191,168,146,123,101,078,111
5966 :055,019,003,015,018,005,193
5972 :037,048,027,048,077,099,164
5978 :118,142,167,181,197,210,081
5984 :221,234,240,159,084,082,092
5990 :069,078,067,072,070,073,019
5996 :082,069,030,080,082,069,008
6002 :083,083,037,066,085,084,040
6008 :084,079,078,037,084,079,049
6014 :037,080,076,065,089,255,216
6020 :255,255,255,255,255,255,126
6026 :255,231,231,231,195,066,067
6032 :066,000,126,126,126,102,178
6038 :000,000,102,126,126,255,247
6044 :255,126,126,126,126,255,146
6050 :255,024,024,024,060,189,226
6056 :189,255,129,000,000,000,229
6062 :000,000,000,000,000,126,044
6068 :165,219,165,165,219,165,254
6074 :126,013,013,013,013,013,121
```

# Guess America!

*(Article on page 64.)*

**BEFORE TYPING...**

**Before typing in programs, please refer to "How To Type COMPUTE!'s Gazette Programs," "A Beginner's Guide To Typing In Programs," and "The Automatic Proofreader" that appear before the Program Listings.**

```
40 DIME$(25):REM # OF LETTERS      :rem 88
50 GOSUB 20000                     :rem 215
60 PRINT"{HOME}{6 RIGHT}{8 DOWN}GUESS AME
   RICA!"                          :rem 26
70 FOR X=1TO2000:NEXT X            :rem 74
80 PRINT"{CLR}"                    :rem 204
90 REM WAGON ACROSS SCREEN"        :rem 118
95 GOSUB20000                      :rem 224
100 PRINT"{HOME}"                  :rem 117
110 FOR X=24 TO 5 STEP -1{2 SPACES}
                                   :rem 230
120 PRINT"{9 DOWN}{24 RIGHT}"{27 SPACES}
                                   :rem 181
130 PRINT TAB(X)"{5 SPACES}UIII"
    {6 SPACES}                     :rem 21
140 PRINT TAB(X)"{B}III{@}GH H"
    {6 SPACES}                     :rem 251
150 PRINT TAB(X)" H H {Y}Q{T}Q";
                                   :rem 174
160 FOR A= 1 TO 90:NEXT A          :rem 243
170 PRINT"{10 LEFT}{10 SPACES}";   :rem 198
180 PRINT"{10 LEFT}{UP}{10 SPACES}";
                                   :rem 88
190 PRINT"{10 LEFT}{UP}{10 SPACES}"
                                   :rem 30
200 FOR A=1 TO 20:NEXT A           :rem 231
210 PRINT"{HOME}"                  :rem 119
220 NEXT X                         :rem 43
230 PRINT"{10 DOWN}{5 RIGHT}{5 SPACES}UII
    I"                             :rem 209
235 PRINT"{5 RIGHT}{B}III{@}GH H
                                   :rem 239
240 PRINT"{5 RIGHT} H H {Y}Q{T}Q"
                                   :rem 132
250 PRINT"{HOME}{7 DOWN}{4 RIGHT}CALIFORN
    IA'S GOLD"                     :rem 222
260 FOR X=1TO3000:NEXT X           :rem 124
1500 REM - GAME DIRECTIONS         :rem 229
1505 PRINT"{CLR}"                  :rem 47
1510 PRINT"{6 SPACES}G U E S S{3 SPACES}A
     M E R I C A !"                :rem 50
1520 PRINT:PRINT                   :rem 28
1530 PRINT"{2 SPACES}THE GAME YOU ARE GOI
     NG TO PLAY IS "               :rem 83
1540 PRINT"BASED ON THE WORD GAME 'JUMBLE
     '.{2 SPACES}YOU"              :rem 6
1550 PRINT"WILL HAVE 15 SECONDS TO UNSCRA
     MBLE THE"                     :rem 221
1560 PRINT"JUMBLED LETTERS AND ENTER YOUR
      ANSWER."                     :rem 97
1570 PRINT:PRINT                   :rem 33
1580 PRINT"{2 SPACES}IF YOU ARE CORRECT Y
     OU WILL BE GIVEN"             :rem 74
1590 PRINT"10 POINTS.{2 SPACES}IF YOU ARE
      WRONG CLUES WILL"            :rem 177
1600 PRINT"BE GIVEN BUT YOU WILL RECEIVE
     {SPACE}FEWER"                 :rem 52
1610 PRINT"POINTS FOR A CORRECT ANSWER."
                                   :rem 174
1620 PRINT:PRINT                   :rem 29
1630 PRINT"{10 SPACES}0 CLUES - 10 PTS."
                                   :rem 250
1640 PRINT"{10 SPACES}1 CLUE{2 SPACES}-
     {2 SPACES}7 PTS."             :rem 127
1650 PRINT"{10 SPACES}2 CLUES -{2 SPACES}
     5 PTS."                       :rem 210
1660 PRINT"{10 SPACES}3 CLUES -{2 SPACES}
     2 PTS."                       :rem 209
1670 PRINT:PRINT                   :rem 34
1680 PRINT"{10 SPACES}PRESS 'C' TO CONTIN
     UE"                           :rem 198
```

```
1690 GET Z$:IF Z$=""THEN1690          :rem 245
1695 IF Z$<>"C" THEN 1690              :rem 226
1700 PRINT"{CLR}"                      :rem 44
1710 PRINT:PRINT                       :rem 29
1720 PRINT"{2 SPACES}YOUR TOTAL NUMBER OF
     POINTS AT THE END";               :rem 49
1721 PRINT"OF THE GAME WILL DETERMINE HOW
     FAR"                              :rem 200
1722 PRINT"ACROSS THE UNITED STATES YOUR
     {SPACE}WAGON"                     :rem 177
1723 PRINT"WILL TRAVEL."               :rem 210
1724 PRINT:PRINT                       :rem 34
1725 PRINT"WILL YOU MAKE IT TO CALIFORNIA
     'S GOLD?"                         :rem 234
1726 PRINT:PRINT                       :rem 36
1727 PRINTSPC(10)"PRESS 'C' TO CONTINUE"
                                       :rem 96
1728 GET Z$:IFZ$="" THEN 1728          :rem 249
1729 IF Z$<>"C" THEN 1728              :rem 226
1731 PRINT "{HOME}{11 DOWN}{16 RIGHT}
     {5 SPACES}":FOR A=1TO150:NEXT A
                                       :rem 254
1735 GOSUB 20000                       :rem 66
1736 PRINT"{HOME}{20 RIGHT}{DOWN}MAXIMUM
     {SPACE}SCORE: 50 "                :rem 67
1739 PRINT"{HOME}{17 DOWN}"            :rem 217
1740 PRINT"{32 RIGHT}10 PTS.":FOR X=1TO50
     0:NEXT X                          :rem 179
1750 PRINT"{UP}{21 RIGHT}20 PTS.":FOR X=1
      TO 500:NEXT X                     :rem 7
1760 PRINT"{UP}{12 RIGHT}30 PTS.":FOR X=1
     TO500:NEXT X                      :rem 4
1770 PRINT"{UP}{2 SPACES}40 PTS.":FOR X=1
     TO500:NEXT X                      :rem 170
1771 FOR X=24 TO 5 STEP -1             :rem 36
1772 PRINT"{HOME}{11 DOWN}{24 RIGHT}"
                                       :rem 40
1773 PRINT TAB(X)"{5 SPACES}UIII"      :rem 83
1774 PRINT TAB(X)"{B}III{@}GH H"
                                       :rem 57
1775 PRINT TAB(X)" H H {Y}Q{T}Q";
                                       :rem 236
1776 FOR A= 1 TO 90:NEXT A             :rem 49
1777 PRINT"{10 LEFT}{10 SPACES}";      :rem 4
1778 PRINT"{10 LEFT}{UP}{10 SPACES}";
                                       :rem 150
1779 PRINT"{10 LEFT}{UP}{10 SPACES}"
                                       :rem 92
1780 FOR A=1 TO 20:NEXT A              :rem 37
1781 PRINT"{HOME}"                     :rem 181
1782 NEXT X                            :rem 105
1783 PRINT"{11 DOWN}{5 RIGHT}{5 SPACES}UI
     II"                               :rem 32
1784 PRINT"{5 RIGHT}{B}III{@}GH H
                                       :rem 41
1785 PRINT"{5 RIGHT} H H {Y}Q{T}Q"
                                       :rem 195
1790 FOR X=1TO1500:NEXT X              :rem 184
2900 PRINT "{CLR}{2 RIGHT}{2 DOWN}REMEMBE
     R..."                             :rem 100
2903 PRINT"{5 RIGHT}{6 DOWN}PRESS {RVS}RE
     TURN{OFF} AFTER TYPING IN"        :rem 139
2905 PRINT"{2 DOWN}{12 RIGHT}YOUR ANSWER.
     "                                 :rem 108
2906 PRINT"{5 DOWN}{3 RIGHT}PRESS {RVS}DE
     L{OFF} KEY TO CORRECT SPELLING."
                                       :rem 126
2910 FOR X=1TO 2000:NEXT X             :rem 175
3000 REM INFORMATION                   :rem 237
3001 DATA GUESS                        :rem 101
3005 DATAWORD,CLUE,CLUE CLUE           :rem 241
3010 DATA FLORIDA,IN THE SOUTH,BOUGHT FRO
     M SPAIN FOR $5 MILLION            :rem 14
3015 DATA GROWS CITRUS FRUIT           :rem 217
3020 DATA LOUISIANA,GREATEST LAND DEAL IN
      HISTORY,COST $15 MILLION         :rem 16
3030 DATA BOUGHT FROM FRANCE IN 1803
                                       :rem 239
3040 DATA CALIFORNIA,GOLD RUSH - 1849,ON
     {SPACE}WEST COAST, MOVIE CAPITAL OF
     {SPACE}U.S.                       :rem 25
3050 DATA ALAMO, OLD SPANISH MISSION, WAR
      WITH MEXICO,REMEMBER THE...:rem 140
3060 DATA JEFFERSON, BOUGHT LOUISIANA TER
     RITORY,PRESIDENT                  :rem 205
3065 DATA DECLARATION OF INDEPENDENCE
                                       :rem 5
3070 DATA OREGON,TRAIL,IN THE NORTHWEST,F
     ROM ENGLAND IN 1846 BY TREATY
                                       :rem 255
3080 DATA TEXAS, LONE STAR REPUBLIC, OIL
     {SPACE}CAPITAL OF U.S., DALLAS
                                       :rem 216
3090 DATA FRONTIER, MOVED WEST,PLACE WHER
     E SETTLEMENT ENDS, WILDERNESS BEGINS
                                       :rem 88
3100 DATA ALASKA,EXTREME NORTH,ESKIMOS LI
     VE HERE,BOUGHT FROM RUSSIA IN 1867
                                       :rem 100
3110 DATA HAWAII, HAD A MONARCHY, TROPICA
     L PACIFIC PARADISE, PEARL HARBOR
                                       :rem 77
3120 DATA INDIANS,WARS AGAINST SETTLERS,A
     MERICAN NATIVES,FOUGHT WITH ARROWS
                                       :rem 101
3130 DATA WASHINGTON,GENERAL,CITY NAMED A
     FTER HIM,FIRST PRESIDENT          :rem 137
3140 DATA JACKSON, PRESIDENT,FOR THE COMM
     ON MAN, FROM THE WEST             :rem 226
3150 DATA PIONEER,1ST PERSON INTO AN AREA
     , MOVED WEST, FOUGHT INDIANS:rem 155
3160 DATA GOLD,FOUND AT SUTTERS MILL -
     {2 SPACES}CA.,DISCOVERED IN 1849, PR
     ECIOUS METAL                      :rem 24
3170 DATA MISSISSIPPI,WESTERN BORDER OF U
     .S. IN 1783,BETWEEN U.S. AND LOUISIA
     NA                                :rem 9
3180 DATA RIVER                        :rem 110
3190 DATA LINCOLN, IN NEBRASKA, BORN IN L
     OG CABIN, PRESIDENT DURING CIVIL WAR
                                       :rem 55
3210 DATA SLAVERY,NORTH OPPOSED IT,SOUTH
     {SPACE}FAVORED IT,CIVIL WAR ENDED IT
                                       :rem 97
3230 DATA KENNEDY, 20TH CENTURY PRESIDENT
     , CUBAN MISSILE CRISIS, ASSASSINATED
                                       :rem 25
3240 DATA PILGRIMS,EARLY SETTLERS,IN MASS
     ACHUSETTS,STARTED THANKSGIVING
                                       :rem 228
3260 DATA COLUMBUS, SAILED FOR SPAIN,THOU
     GHT EARTH WAS ROUND,DISCOVERED AMERI
     CA                                :rem 203
3270 DATA REVOLUTION,WAR,FOUGHT AGAINST E
     NGLAND,1776                       :rem 13
3280 DATA CROCKETT,HERO WHO DIED AT THE A
     LAMO,COONSKIN CAP,DAVY            :rem 72
3290 DATA ROOSEVELT, PRESIDENT DURING WOR
     LD WAR II, RELATED TO TEDDY :rem 156
3295 DATA MOST TERMS                   :rem 187
```

```
3300 DATA GETTYSBURG,IN CIVIL WAR,BATTLEF
     IELD,LINCOLN'S FAMOUS ADDRESS
                                   :rem 230
3310 DATA STATES,U.S. IS MADE UP OF THEM,
     POLITICAL UNITS, THERE ARE 50:rem 65
3320 DATA AMENDMENT, THERE ARE 26 OF THEM
     ,ADDITIONS TO CONSTITUTION    :rem 67
3325 DATA ERA DIDN'T MAKE IT       :rem 20
3330 DATA FRANKLIN,COLONIAL STATESMAN,INV
     ENTOR,BENJAMIN                :rem 118
3340 DATA EISENHOWER, FAMOUS GENERAL,IN W
     ORLD WAR II, PRESIDENT BEFORE KENNED
     Y                             :rem 52
3350 DATA MORMONS,RELIGIOUS GROUP,LED BY
     {SPACE}BRIGHAM YOUNG,SETTLED IN UTAH
                                   :rem 174
3360 DATA CARTER,20TH CENTURY PRESIDENT,F
     ROM GEORGIA,FAMOUS FOR PEANUTS:rem 5
3365 DATA BUFFALO,ANIMAL,HUNTED BY INDIAN
     S,HIDE USED FOR TENTS         :rem 73
3370 DATA SCOTT,U.S. GENERAL IN MEXICAN W
     AR,CAPTURED MEXICO CITY,WINFIELD
                                   :rem 40
3800 DATA *,*,*,*                  :rem 17
3995 P=0                           :rem 151
3999 C=0                           :rem 142
4000 REM RANDOM GENERATION OF DATA :rem 4
4005 C=C+1:REM WORD COUNTER        :rem 98
4010 DM=0                          :rem 195
4020 DM=DM+1                       :rem 129
4030 READ A$,A$,A$,A$              :rem 251
4040 IF A$<>"*" THEN 4020          :rem 153
4050 DM=DM-1                       :rem 134
4060 RESTORE                       :rem 238
4070 R=INT(RND(1)*DM)+1            :rem 19
4080 FOR T=1TOR-1                  :rem 200
4090 READ A$,A$,A$,A$              :rem 1
4100 NEXT T                        :rem 88
4110 READ W$,C$(1),C$(2),C$(3)     :rem 159
5000 REM SCRAMBLED WORD ROUTINE    :rem 152
5010 FL=0                          :rem 197
5020 L=LEN(W$):S$=""               :rem 45
5040 FOR M=1TOL                    :rem 90
5050 E$(M)=MID$(W$,M,1)            :rem 174
5060 NEXT M                        :rem 87
5080 FOR M=1TOL                    :rem 94
5090 R=INT((L-M+1)*RND(1)+1)       :rem 248
5100 S$=S$+E$(R)                   :rem 40
5110 H$=E$(R)                      :rem 124
5120 E$(R)=E$(L-M+1)               :rem 237
5130 E$(L-M+1)=H$                  :rem 78
5140 NEXTM                         :rem 86
5150 IFL=1 THEN 6000               :rem 9
5160 IFS$<> W$ THEN 6000           :rem 188
5170 FL=FL+1                       :rem 138
5180 IFFL>=5 THEN 6000             :rem 148
5190 GOTO5020                      :rem 207
6000 REM INITIALIZE SCREEN VARIABLE
     {6 SPACES}LOCATION            :rem 251
6010 SS$="{HOME}{4 RIGHT}{5 DOWN}":REM SC
     RAMBLED                       :rem 153
6020 WW$="{HOME}{2 RIGHT}{7 DOWN}":REM CO
     RRECT WORD                    :rem 75
6030 CC$(1)="{HOME}{5 RIGHT}{11 DOWN}":RE
     M CLUE 1                      :rem 77
6040 CC$(2)="{HOME}{5 RIGHT}{13 DOWN}":RE
     M CLUE 2                      :rem 114
6050 CC$(3)="{HOME}{5 RIGHT}{15 DOWN}":RE
     M CLUE 3                      :rem 151
6060 REM BEGIN GAME                :rem 47
6070 GOSUB 20000                   :rem 63
6075 PRINT"{HOME}{DOWN}{17 RIGHT}PLAYING
     {SPACE}FOR: 10 PTS.           :rem 77
6080 PRINTSS$S$                    :rem 156
6100 PRINT WW$;                    :rem 97
6111 GOSUB 7000                    :rem 16
6115 IF LEFT$(X$,LEN(W$))=W$ THEN P=P+10:
      REM SCORE                    :rem 57
6120 IFLEFT$(X$,LEN(W$))=W$GOTO8000:rem 4
6125 PRINT"{HOME}{DOWN}{30 RIGHT} 7"
                                   :rem 96
6130 PRINTCC$(1)C$(1)              :rem 108
6140 PRINTWW$;                     :rem 101
6150 REM INPUT X$                  :rem 188
6151 GOSUB 7000                    :rem 20
6155 IF LEFT$(X$,LEN(W$))=W$ THEN P=P+7
                                   :rem 121
6160 IF LEFT$(X$,LEN(W$))=W$GOTO8000
                                    :rem 8
6165 PRINT"{HOME}{DOWN}{30 RIGHT} 5"
                                   :rem 98
6170 PRINTCC$(2)C$(2)              :rem 114
6180 PRINTWW$;                     :rem 105
6191 GOSUB 7000                    :rem 24
6200 IF LEFT$(X$,LEN(W$))=W$ THEN P=P+5
                                   :rem 110
6205 IF LEFT$(X$,LEN(W$))=W$ GOTO 8000
                                    :rem 8
6207 PRINT"{HOME}{DOWN}{30 RIGHT} 2"
                                   :rem 92
6210 PRINTCC$(3)C$(3)              :rem 111
6220 PRINTWW$;                     :rem 100
6231 GOSUB 7000                    :rem 19
6235 IF LEFT$(X$,LEN(W$))=W$ THEN P=P+2
                                   :rem 115
6240 IF LEFT$(X$,LEN(W$))=W$ GOTO 8000
                                    :rem 7
6245 FOR X=1TO1000:NEXT X          :rem 179
6249 PRINT"{HOME}{DOWN}{30 RIGHT} 0"
                                   :rem 96
6250 PRINT"{HOME}{4 RIGHT}{7 DOWN}"W$" ←
     ← ←"                          :rem 120
6260 PRINT"{HOME}{RIGHT}{20 DOWN}PRESS"
                                   :rem 176
6261 PRINT"{RIGHT}'C' TO"          :rem 241
6262 PRINT"{RIGHT}CONTINUE"        :rem 35
6270 GET Z$:IFZ$=""THEN 6270       :rem 243
6280 IF C=5 THEN GOTO{2 SPACES}9000:REM T
     O END                         :rem 221
6290 IF C<5 THEN 4000:REM NEXT QUES.
                                   :rem 208
7000 REM 15 SEC. TIME DELAY FOR ANS.
                                   :rem 175
7005 PRINTWW$ "? ";                :rem 233
7010 POKE204,0:REM TURNS CURSOR ON:rem 30
7020 X$=""                         :rem 198
7030 FOR T=1 TO 1000 :REM 15 SEC.:rem 100
7040 GET R$:IFR$=CHR$(13) THEN 7090:REM R
     ETURN KEY BEING HIT           :rem 131
7050 IF R$=""THEN 7080:REM NO ANSWER BEIN
     G ENTERED                      :rem 71
7055 IF R$=CHR$(20) THEN 7005      :rem 194
7056 IF R$=CHR$(157) THEN 7005     :rem 254
7060 X$=X$+R$:REM ACCUMULATES INPUTS
                                   :rem 219
7070 PRINTR$;:REM EACH LETTER AS INPUT
                                   :rem 47
7080 NEXT T:REM TIME COUNT         :rem 56
7090 POKE 204,1:REM TURNS OFF CURSOR
                                   :rem 101
7105 PRINT " "                     :rem 158
```

```
7120 RETURN                              :rem 170
8000 REM CORRECT RESPONSE SEQUENCE
                                         :rem 134
8010 GOSUB 20000                         :rem 59
8030 PRINT"{HOME}{3 RIGHT}{3 DOWN}CORRECT
     "                                   :rem 75
8034 FOR X= 5 TO 28 STEP 2               :rem 251
8041 PRINT TAB(X)"CORRECT"               :rem 48
8050 NEXT X                              :rem 100
8060 PRINT"{HOME}{5 RIGHT}{12 DOWN}
     {5 SPACES}UIII"                     :rem 63
8070 PRINT"{5 RIGHT}{B}III{@}GH
     {SHIFT-SPACE}H"                     :rem 230
8080 PRINT"{5 RIGHT} H H {Y}Q{T}Q"
                                         :rem 190
8090 PRINT"{HOME}{RIGHT}{20 DOWN}PRESS"
                                         :rem 179
8100 PRINT"{RIGHT}'C' TO"                :rem 235
8110 PRINT"{RIGHT}CONTINUE"              :rem 29
8120 GET Z$:IFZ$="" THEN8120             :rem 235
8125 IF Z$<>"C" THEN 8120                :rem 216
8130 IF C<5 THEN 4000:REM NEXT QUES.
                                         :rem 203
9000 REM GAME ENDING AND SCORE           :rem 203
9010 GOSUB 20000                         :rem 60
9015 F=F+1:REM FLASHING SCORE            :rem 218
9020 PRINT"{HOME}{2 DOWN}{24 RIGHT} "P" P
     TS."                                :rem 66
9040 FOR X=1 TO 200:NEXT X               :rem 128
9050 PRINT"{HOME}{2 DOWN}{24 RIGHT}
     {10 SPACES}"                        :rem 140
9060 FOR X=1TO200:NEXT X                 :rem 130
9070 IF F<5 THEN GOTO 9015               :rem 77
9080 PRINT"{HOME}{2 DOWN}{24 RIGHT} "P" P
     TS."                                :rem 72
9090 IF P>=40 THEN 9640                  :rem 146
9100 IF P>=30 THEN 9450                  :rem 136
9110 IF P>=20 THEN 9280                  :rem 137
9120 IF P<20 THEN 9130                   :rem 69
9130 REM P<20                            :rem 159
9140 FOR X=24 TO 23 STEP -1              :rem 82
9150 PRINT"{HOME}"                       :rem 179
9160 PRINT"{9 DOWN}{24 RIGHT}"           :rem 242
9170 PRINT TAB(X)"{5 SPACES}UIII"        :rem 82
9180 PRINT TAB(X)"{B}III{@}GH H"
                                         :rem 56
9190 PRINT TAB(X)" H H {Y}Q{T}Q";
                                         :rem 235
9200 FOR A= 1 TO 90:NEXT A               :rem 39
9210 PRINT"{10 LEFT}{10 SPACES}";:rem 250
9220 PRINT"{10 LEFT}{UP}{10 SPACES}";
                                         :rem 140
9230 PRINT"{10 LEFT}{UP}{10 SPACES}"
                                         :rem 82
9240 FOR A=1 TO 20:NEXT A                :rem 36
9250 PRINT"{HOME}"                       :rem 180
9260 NEXT X                              :rem 104
9261 PRINT"{HOME}"                       :rem 182
9262 PRINT"{9 DOWN}{24 RIGHT}"           :rem 245
9263 PRINT TAB(X)"{5 SPACES}UIII"        :rem 85
9264 PRINT TAB(X)"{B}III{@}GH H"
                                         :rem 59
9265 PRINT TAB(X)" H H {Y}Q{T}Q";
                                         :rem 238
9270 PRINT"{HOME}{16 DOWN}{13 RIGHT}STUCK
      ON EAST COAST."                    :rem 59
9272 FOR X=1 TO 5000:NEXT X              :rem 186
9275 GOTO 13000                          :rem 4
9280 REM P=20-29                         :rem 62
9290 PRINT"{HOME}"                       :rem 184
9300 FOR X=24 TO 16 STEP -1              :rem 82
9310 PRINT"{9 DOWN}{24 RIGHT}"           :rem 239
9320 PRINT TAB(X)"{5 SPACES}UIII"        :rem 79
9330 PRINT TAB(X)"{B}III{@}GH H"
                                         :rem 53
9340 PRINT TAB(X)" H H {Y}Q{T}Q";
                                         :rem 232
9350 FOR A= 1 TO 90:NEXT A               :rem 45
9365 PRINT"{10 LEFT}{10 SPACES}";        :rem 5
9370 PRINT"{10 LEFT}{UP}{10 SPACES}";
                                         :rem 146
9380 PRINT"{10 LEFT}{UP}{10 SPACES}"
                                         :rem 88
9390 FOR A=1 TO 20:NEXT A                :rem 42
9400 PRINT"{HOME}"                       :rem 177
9405 NEXT X                              :rem 105
9410 PRINT"{HOME}"                       :rem 178
9411 PRINT"{9 DOWN}{24 RIGHT}"           :rem 241
9412 PRINT TAB(X)"{5 SPACES}UIII"        :rem 81
9413 PRINT TAB(X)"{B}III{@}GH H"
                                         :rem 55
9414 PRINT TAB(X)" H H {Y}Q{T}Q";
                                         :rem 234
9420 PRINT"{HOME}{15 DOWN}{8 RIGHT}ALMOST
      TO THE MISSISSIPPI."               :rem 121
9430 FOR X=1TO5000:NEXT X                :rem 182
9440 GOTO 13000                          :rem 254
9445 REM 30-39 PTS.                      :rem 219
9446 PRINT"{HOME}"                       :rem 187
9450 FOR X =24 TO 11 STEP -1             :rem 83
9451 PRINT"{HOME}"                       :rem 183
9455 PRINT"{9 DOWN}{24 RIGHT}"           :rem 249
9460 PRINT TAB(X)"{5 SPACES}UIII"        :rem 84
9470 PRINT TAB(X)"{B}III{@}GH H"
                                         :rem 58
9480 PRINT TAB(X)" H H {Y}Q{T}Q";
                                         :rem 237
9490 FOR A=1TO90:NEXT A                  :rem 50
9500 PRINT"{10 LEFT}{10 SPACES}";:rem 252
9510 PRINT"{10 LEFT}{UP}{10 SPACES}";
                                         :rem 142
9520 PRINT"{10 LEFT}{UP}{10 SPACES}"
                                         :rem 84
9530 FOR A=1 TO 20:NEXT A                :rem 38
9540 PRINT"{HOME}"                       :rem 182
9550 NEXT X                              :rem 106
9560 PRINT"{HOME}"                       :rem 184
9570 PRINT"{9 DOWN}{24 RIGHT}"           :rem 247
9580 PRINT TAB(X)"{5 SPACES}UIII"        :rem 87
9590 PRINT TAB(X)"{B}III{@}GH H"
                                         :rem 61
9600 PRINT TAB(X)" H H {Y}Q{T}Q";
                                         :rem 231
9610 PRINT"{HOME}{15 DOWN}{3 RIGHT}MADE I
     T TO THE ROCKIES"                   :rem 82
9620 FOR X=1TO5000:NEXT X                :rem 183
9630 GOTO 13000                          :rem 255
9640 REM 40 OR MORE                      :rem 239
9650 FOR X=24 TO 5 STEP -1               :rem 40
9660 PRINT"{HOME}"                       :rem 185
9670 PRINT"{9 DOWN}{24 RIGHT}"           :rem 248
9680 PRINT TAB(X)"{5 SPACES}UIII"        :rem 88
9690 PRINT TAB(X)"{B}III{@}GH H"
                                         :rem 62
9700 PRINT TAB(X)" H H {Y}Q{T}Q";
                                         :rem 232
9710 FOR A= 1 TO 90:NEXT A               :rem 45
9720 PRINT"{10 LEFT}{10 SPACES}";  :rem 0
9730 PRINT"{10 LEFT}{UP}{10 SPACES}";
                                         :rem 146
9740 PRINT"{10 LEFT}{UP}{10 SPACES}"
                                         :rem 88
```

```
9750 FOR A=1 TO 20:NEXT A          :rem 42
9760 PRINT"{HOME}"                 :rem 186
9770 NEXT X                        :rem 110
9780 PRINT"{HOME}"                 :rem 188
9790 PRINT"{9 DOWN}{24 RIGHT}"     :rem 251
9800 PRINT TAB(X)"{5 SPACES}UIII"  :rem 82
9810 PRINT TAB(X)"{B}III{@}GH H"   :rem 56
9820 PRINT TAB(X)" H H {Y}Q{T}Q";
                                   :rem 235
9825 PRINT"{HOME}{6 DOWN}{4 RIGHT}HOORAY!
     !!!"                          :rem 236
9830 PRINT"{HOME}{16 DOWN}{3 RIGHT}CALIFO
     RNIA AND GOLD!"               :rem 17
9840 FOR X=1TO5000:NEXT X          :rem 187
9850 GOTO 13000                    :rem 3
13000 PRINT"{CLR}{2 DOWN}DO YOU WANT TO P
      LAY AGAIN?"                  :rem 188
13010 PRINT:PRINT:PRINT:PRINT:PRINT:PRINT
                                   :rem 101
13020 PRINT"{11 RIGHT}TYPE {RVS}P{OFF} TO
       PLAY":PRINT:PRINT           :rem 163
13025 PRINT"{11 RIGHT}TYPE {RVS}Q{OFF} TO
       QUIT"                       :rem 40
13030 GET Z$:IF Z$="" THEN 13030   :rem 67
13040 IF Z$="P" THEN 2900          :rem 208
13060 IF Z$<> "P" AND Z$ <> "Q" THEN 1303
      0                            :rem 155
14500 REM BYE!                     :rem 223
14510 GOSUB 20000                  :rem 109
14520 PRINT"{HOME}{5 DOWN}{4 RIGHT}BYE!
                                   :rem 136
14530 PRINT"{5 DOWN}{4 RIGHT}SEE YOU IN C
      ALIFORNIA!"                   :rem 1
14540 FOR X=1TO5000:NEXT X         :rem 228
14550 PRINT"{HOME}{22 DOWN}"       :rem 89
14999 END                          :rem 231
19999 REM MAP SUBROUTINE           :rem 231
20000 PRINT"{CLR}"                 :rem 86
20010 PRINT SPC(2)"ZZZ"            :rem 187
20020 PRINT SPC(3)"Z"SPC(1)"ZZZZZZ"SPC(25
      )"ZZ"                        :rem 103
20030 PRINT SPC(2)"Z"SPC(8)"ZZZZZZZZZZ"SP
      C(14)"Z"SPC(1)"Z"            :rem 128
20040 PRINT SPC(2)"Z"SPC(18)"ZZ"SPC(12)"Z
      "SPC(2)"Z"                   :rem 225
20050 PRINT SPC(2)"Z"SPC(19)"ZZZZ"SPC(7)"
      ZZ"SPC(2)"Z"                 :rem 69
20060 PRINT SPC(1)"Z"SPC(23)"Z"SPC(1)"Z"S
      PC(5)"Z"SPC(2)"Z"            :rem 92
20070 PRINT SPC(1)"Z"SPC(22)"Z"SPC(1)"Z"S
      PC(1)"Z"SPC(2)"ZZ"SPC(3)"Z":rem 186
20080 PRINT SPC(1)"Z"SPC(22)"Z"SPC(1)"Z"S
      PC(1)"Z"SPC(1)"ZZ"SPC(4)"Z":rem 187
20090 PRINT SPC(1)"Z"SPC(23)"Z"SPC(2)"ZZZ
      "SPC(4)"Z"                   :rem 140
20100 PRINT SPC(1)"Z"SPC(32)"Z"    :rem 192
20110 PRINT SPC(1)"Z"SPC(32)"Z"    :rem 193
20120 PRINT SPC(1)"Z"SPC(32)"Z"    :rem 194
20130 PRINT SPC(1)"Z"SPC(32)"Z"    :rem 195
20140 PRINT SPC(2)"Z"SPC(31)"Z"    :rem 196
20150 PRINT SPC(3)"Z"SPC(30)"Z"    :rem 197
20160 PRINT SPC(3)"ZZZ"SPC(27)"Z":rem 128
20170 PRINT SPC(6)"Z"SPC(25)"Z"    :rem 206
20180 PRINT SPC(7)"ZZZ"SPC(1)"Z"SPC(19)"Z
      "                            :rem 13
20190 PRINT SPC(10)"Z"SPC(1)"Z"SPC(13)"ZZ
      ZZ"SPC(1)"Z"                 :rem 146
20200 PRINT SPC(13)"Z"SPC(7)"ZZZ"SPC(1)"Z
      "SPC(4)"Z"SPC(1)"Z"          :rem 15
20210 PRINT SPC(14)"ZZZ"SPC(3)"Z"SPC(3)"Z
      "SPC(6)"Z"SPC(1)"Z"          :rem 17
20220 PRINT SPC(17)"Z"SPC(1)"Z"SPC(11)"Z"
      SPC(1)"Z"                    :rem 3
20230 PRINT SPC(18)"Z"SPC(13)"ZZ":rem 213
20500 RETURN                       :rem 215
```

# Sea Route To India

*(Article on page 66.)*

**BEFORE TYPING...**

**Before typing in programs, please refer to "How To Type COMPUTE!'s Gazette Programs," "A Beginner's Guide To Typing In Programs," and "The Automatic Proofreader" that appear before the Program Listings.**

```
70 DIM M%(7),M$(6):X=RND(-TI)      :rem 45
71 FG=1.6:KB=151:HP=72:PRINT"{CLR}":IF PE
   EK(1024)=32THENKB=197:HP=29     :rem 52
72 M$(0)="":M$(1)="STOPPED AT CANARY ISLA
   NDS":M$(5)="SIGHTED CALCUTTA"   :rem 21
73 DH$="{2 SPACES}{L}{DOWN}{3 LEFT}
   {*}{RVS}{3 SPACES}{*}{DOWN}
   {4 LEFT}{4 SPACES}{DOWN}{4 LEFT}
   {4 SPACES}{DOWN}{5 LEFT}{RVS}£
   {3 SPACES}{OFF}£{DOWN}{3 LEFT}{L}
   {DOWN}{4 LEFT}{*}{RVS}{6 SPACES}
   {OFF}£"                         :rem 122
74 M$(2)="CAPE VERDE ISLANDS":M$(3)="ROUN
   DED CAPE OF GOOD HOPE"          :rem 65
76 M$(4)="PICKED UP INDIAN PILOT":rem 116
77 YS$=" -{DOWN}{LEFT}{RVS}+{DOWN}{LEFT}
   {OFF}-{DOWN}{2 LEFT}{*}{RVS} {OFF}£
   ":HS$="{RED} -{DOWN}{LEFT}{RVS}Z{OFF}
   {DOWN}{LEFT}-{DOWN}{2 LEFT}{*}{RVS}
   {SPACE}{OFF}£"                  :rem 108
78 MS$="{RIGHT}"+CHR$(20)+"{DOWN}":MS$=MS
   $+MS$:MS$=MS$+MS$               :rem 12
80 WH$(1)="NO WIND AT ALL":WH$(2)="VERY C
   ALM":WH$(3)="FAIR WINDS"        :rem 120
82 WH$(4)="GOOD WINDS":WH$(5)="GOOD WINDS
   ":WH$(6)="STRONG WINDS"         :rem 250
84 M%(1)=50:M%(2)=100:M%(3)=150:M%(4)=200
   :M%(5)=250:M%(6)=300:M%(7)=50   :rem 225
90 GOSUB16000                      :rem 224
91 DT$="{2 LEFT}.{DOWN}{LEFT}.{2 LEFT}.
   {2 LEFT}.{DOWN}{LEFT}.{DOWN}{LEFT}.
   {DOWN}{LEFT}.{DOWN}{LEFT}.{2 LEFT}.
   {DOWN}{LEFT}.{DOWN}{LEFT}.{DOWN}{LEFT}
   .{DOWN}{OFF}.{RVS}{OFF}.{DOWN}{OFF}.
   {RVS}{OFF}.{RVS}{OFF}.{DOWN}{LEFT}.
   {DOWN}{LEFT}.{DOWN}{LEFT}.{DOWN}{LEFT}
   .{DOWN}{LEFT}."                 :rem 139
92 DT$=DT$+"{DOWN}{LEFT}.{DOWN}{LEFT}.
   {DOWN}{OFF}.{DOWN}{OFF}.{RVS}{OFF}.
   {RVS}{OFF}.{RVS}{OFF}.{UP}{LEFT}.{UP}
   {OFF}.{UP}{OFF}.{UP}{LEFT}.{UP}{OFF}.
   {UP}{LEFT}.{UP}{LEFT}.{UP}{LEFT}.{UP}
   {OFF}.{UP}{OFF}.{RVS}{OFF}.{RVS}{OFF}.
   {RVS}{OFF}."                    :rem 54
93 DT$=DT$+"{RVS}{OFF}.{UP}{OFF}.{UP}
   {LEFT}."                        :rem 15
104 DEF FNR(X)=INT(RND(1)*X+1)     :rem 176
105 F$="{2 SHIFT-SPACE}<.{2 @}V
    {13 SHIFT-SPACE}(.{2 I}N
    {34 SHIFT-SPACE}"              :rem 17
106 F$=F$+"{28 SHIFT-SPACE}"       :rem 151
107 F$=F$+"<.{2 +}V{40 SHIFT-SPACE}(,
    {2 +}N{15 SHIFT-SPACE}"        :rem 114
```

```
100 F$=F$+"{28 SHIFT-SPACE}"          :rem 153
110 D$="{HOME}{32 DOWN}"              :rem 174
120 S$="{3 SPACES}{M}{2 SPACES}{M}
    {DOWN}{5 LEFT}{3 +} {2 +}{DOWN}
    {6 LEFT}{3 £} {2 £}{DOWN}
    {6 LEFT}{3 +} {2 +}{DOWN}{4 LEFT}
    {G} {M} {DOWN}{6 LEFT}{*}{RVS}
    {SPACE}ZZZ {OFF}£"                :rem 135
122 SS$="{3 SPACES}{M}{2 SPACES}{M}
    {DOWN}{5 LEFT}{3 +} {2 +}{DOWN}
    {6 LEFT}{3 £} {2 £}{DOWN}
    {6 LEFT}{3 +} {2 +}{DOWN}{4 LEFT}
    {G} {M} {DOWN}{6 LEFT}{*}{RVS}
    {5 SPACES}{OFF}£                  :rem 44
125 GOSUB15000                        :rem 14
130 GOTO500                           :rem 98
500 ML=8:GOSUB10000:FORWK=1TO52:Z=FRE(0):
    FORI=1TO10:GETA$:NEXT             :rem 140
510 GOSUB14000:POKE53281,3:REMWEATHER,MIL
    ES                                :rem 170
520 GOSUB10000:REM LOG                 :rem 8
530 GOSUB11000:REM SITUATION          :rem 232
550 X=FNR(GG):IFGG=6THENX=2*FNR(4)-1
                                      :rem 167
555 IFML<1200ANDX=4THENX=1             :rem 87
560 ONXGOSUB1000,2000,1000,4000,5000,6000
    ,6000                             :rem 113
790 FD=FD-1:SP=SP-1:WT=WT-1:IFWK>30THENCH
    =CH-1                             :rem 129
800 NEXTWK                            :rem 121
1000 REM CATCH WHALE ROUTINE          :rem 159
1002 DZ=17+INT(8*RND(1))               :rem 53
1005 PRINT"{CLR}WHALES SIGHTED"       :rem 246
1006 PRINT"{DOWN}TRY YOUR LUCK? Y OR N"
                                      :rem 173
1007 A$="":GETA$:IFA$="N"THEN 1155:rem 37
1008 IFA$<>"Y"THEN1007                :rem 203
1010 PRINT"{CLR}{DOWN}{11 SPACES}W{DOWN}
    {LEFT}{+}*-{DOWN}{3 LEFT}{+}V"
                                      :rem 227
1020 PRINT"{8 SPACES}{*}{RVS}{3 SPACES}
    {OFF}£"                           :rem 192
1030 PRINT"{5}JK{SHIFT-SPACE}JKJKJKJKKJ
    KJKJKJKJKJKJK{BLK}"               :rem 36
1040 PRINT"{HOME}PRESS H{OFF} "       :rem 16
1050 GOSUB1200                         :rem 9
1055 IFPEEK(KB)<>HPTHEN1050           :rem 100
1058 DC=0:PRINTLEFT$(D$,3)TAB(13)" {DOWN}
    {LEFT}-{DOWN}{LEFT}V{DOWN}{LEFT}";
                                      :rem 56
1060 DC=DC+1:GOSUB1200:PRINTLEFT$(D$,3+DC
    )TAB(13)" {DOWN}{LEFT}-{DOWN}{LEFT}V
    {DOWN}{LEFT}";:GOTO1070           :rem 78
1070 IFDC<>DZ-6THEN1060               :rem 79
1100 B$="*":OPEN3,3:INPUT#3,B$:CLOSE3:IFL
    EFT$(B$,1)<>"{SHIFT-SPACE}"THEN1150
                                      :rem 230
1110 PRINTLEFT$(D$,3+DC)TAB(13)" {DOWN}
    {LEFT}-{DOWN}{LEFT}V{LEFT}{UP}
    {DOWN}{LEFT}-{DOWN}{LEFT}V {2 UP}MIS
    SED";:GOTO1155                    :rem 230
1150 PRINT"{7 UP}GOOD SHOT":FD=FD+2
                                      :rem 222
1155 PRINTLEFT$(D$,23) "{4 UP}PRESS {RED}
    RETURN{BLK}"                      :rem 147
1157 A$="":GETA$:IFA$<>CHR$(13)THEN1157
                                      :rem 142
1159 RETURN                           :rem 176
1200 PRINTLEFT$(D$,DZ)LEFT$(F$,39)
                                      :rem 196
1210 F$=MID$(F$,2)+LEFT$(F$,1)        :rem 20
1245 REM{4 SPACES}A$="":GETA$:IFA$<>"L"TH
    EN1245                            :rem 72
1250 RETURN                           :rem 168
2000 REMFOREIGN SHIPS                 :rem 55
2010 PRINT"{CLR}{RED}"TAB(25)S$       :rem 73
2020 PRINT"{BLK}SHIP SIGHTED"         :rem 97
2030 PRINT"{RVS}A{OFF}PPROACH OR {RVS}F
    {OFF}LEE"                         :rem 233
2040 A$="":GETA$:IFA$<>"A"ANDA$<>"F"THEN2
    040                               :rem 137
2050 IFA$="A"ANDRND(1)>.2THEN3000     :rem 70
2060 ES=.5:IFA$="F"THENES=.8:GOTO2100
                                      :rem 149
2070 PRINT"{DOWN}IT'S A PIRATE SHIP!":PRI
    NT"{DOWN}YOU TURN AND FLEE"       :rem 110
2100 REMFLEE ROUTINE                  :rem 233
2110 IFRND(1)>ESTHENPRINT"{2 DOWN}ALAS.":
    PRINT"{DOWN}THEY CATCH AND SINK YOU"
    :GOTO17000                         :rem 41
2199 GOTO3140                         :rem 214
3000 FL=0:REM RACE                    :rem 251
3001 PRINT"{CLR}ITS CAPTAIN CHALLENGES YO
    U TO A RACE{DOWN}":IFRND(1)>.5THEN30
    03                                :rem 192
3002 PRINT"3 PIECES OF HIS GOLD FOR 3
    {2 SPACES}BARRELS OF{2 SPACES}YOUR S
    UPPLIES.":GOTO3005                 :rem 8
3003 FL=1:PRINT"3 BARRELS OF HIS SUPPLIES
     AGAINST"                          :rem 4
3004 PRINT"3 PIECES OF YOUR GOLD."
                                      :rem 188
3005 PRINT"{DOWN}DO YOU ACCEPT? {RVS}Y
    {OFF} OR {RVS}N{OFF}?"            :rem 248
3006 A$="":GETA$:IFA$="Y"THEN3017     :rem 48
3007 MS$="{RIGHT}"+CHR$(20)+"{DOWN}":MS$=
    MS$+MS$:MS$=MS$+MS$               :rem 103
3008 IFA$="N"THENRETURN               :rem 157
3009 GOTO3006                         :rem 206
3010 YS$="{BLK} -{DOWN}{LEFT}{RVS}+{DOWN}
    {LEFT}{OFF}-{DOWN}{2 LEFT}{*}{RVS}
     {OFF}£":HS$="{RED} -{DOWN}{LEFT}
    {RVS}Z{OFF}{DOWN}{LEFT}-{DOWN}
    {2 LEFT}{*}{RVS} {OFF}£"  :rem 82
3017 MS$="{RIGHT}"+CHR$(20)+"{DOWN}":MS$=
    MS$+MS$:MS$=MS$+MS$               :rem 104
3050 PRINT"{CLR}{BLK}";               :rem 247
3060 PRINTTAB(36)YS$:PRINT"{2 DOWN}"TAB(3
    6)HS$                              :rem 52
3070 FORT=1TO1000:NEXT                :rem 80
3075 YX=INT(RND(1)*10)+25:HX=INT(RND(1)*9
    )+25:IFHX=YXTHENYX=YX+1           :rem 171
3080 MX=YX:W$="YOUR":IFYX<HXTHENMX=HX:W$=
    "HIS"                              :rem 4
3090 FORJ=1TOMX                       :rem 179
3092 IFYX<JTHEN3095                   :rem 148
3093 PRINT"{HOME}"MS$                 :rem 119
3095 IFHX<JTHEN3100                   :rem 121
3096 PRINT"{HOME}{6 DOWN}"MS$         :rem 224
3100 NEXTJ                            :rem 77
3110 PRINT"{BLK}{HOME}{15 DOWN}"W$" SHIP
    {SPACE}WINS"                      :rem 108
3120 IFMX=YXTHENGP=GP-(FL=0)*3:SP=SP+3*FL
    : CH=CH+2                         :rem 29
3130 IFMX=HXTHENGP=GP-FL*3:SP=SP+3*(FL=0)
    : CH=CH-2                         :rem 15
3140 PRINTLEFT$(D$,23) "PRESS {RED}RETURN
    {BLK}"                            :rem 75
3145 A$="":GETA$:IFA$<>CHR$(13)THEN3145
                                      :rem 140
3150 RETURN                           :rem 169
4000 REMRIVER                          :rem 48
```

```
4010 IFRND(1)>.7THEN RETURN       :rem 108
4020 PRINT"{CLR}YOU SPY A RIVER.":PRINT"
     {DOWN}WILL YOU GO ASHORE FOR FOOD AN
     D WATER?"                    :rem 103
4025 PRINT"{DOWN}{RVS}Y{OFF} OR {RVS}N
     {OFF}"                       :rem 61
4030 A$="":GETA$:IFA$<>"Y"ANDA$<>"N"THEN4
     030                          :rem 171
4040 IFA$="N"THEN CH=CH-2:RETURN :rem 134
4050 PRINT"{2 DOWN}YOU LAND AND REPLENISH
     ."                           :rem 131
4060 IFRND(1)>.5THEN4800          :rem 91
4070 PRINTLEFT$(D$,23) "PRESS {RED}RETURN
     {BLK}"                       :rem 78
4075 A$="":GETA$:IFA$<>CHR$(13)THEN4075
                                  :rem 146
4080 PRINT"{CLR}NATIVES APPEAR{3 SPACES}O
     {3 SPACES}O{DOWN}{6 LEFT}J{RVS}{RED}
      {BLK}{OFF}K {Z}{RVS}{GRN} {OFF}
     {BLK}{X}{DOWN}{6 LEFT}V{3 SPACES}V
     "                            :rem 108
4082 PRINT"{DOWN}{RVS}A{OFF}PPROACH OR
     {RVS}F{OFF}LEE?"             :rem 66
4083 A$="":GETA$:IFA$<>"A"ANDA$<>"F"THEN4
     083                          :rem 155
4085 IFA$="F"THEN4800             :rem 135
4090 IFRND(1)>.5THEN4300          :rem 89
4100 PRINT"{3 DOWN}THE NATIVES TRADE GOLD
      FOR YOUR TRINKETS"           :rem 4
4110 GP=GP+10:CH=CH+1:GOTO4800    :rem 229
4300 PRINT"{CLR}{2 SPACES}O{3 SPACES}O
     {DOWN}{6 LEFT}J{RVS}{YEL} {BLK}{OFF}
     K{Z}{RVS}{RED} {OFF}{BLK}{X}
     {DOWN}{6 LEFT}V - V";        :rem 138
4302 PRINT"{2 UP}{5 SPACES}O{3 SPACES}O
     {DOWN}{6 LEFT}J{RVS}{GRN} {BLK}{OFF}
     K{Z}{RVS}{1} {BLK}{OFF}{X}
     {DOWN}{6 LEFT}V - V"         :rem 197
4305 PRINT"{3 DOWN}MORE NATIVES APPEAR!":
     PRINT"{DOWN}RUN FOR THE SHIP!"
                                  :rem 37
4310 PRINT"{DOWN}TYPE {BLU}RUN{BLK} AND P
     RESS RETURN"                 :rem 208
4320 TI$="000000"                 :rem 43
4330 INPUTA$:IFA$<>"RUN"THEN4330  :rem 153
4340 IFTI<200THEN4500             :rem 189
4350 PRINT"{DOWN}TOO SLOW. YOU'RE DEAD.":
     GOTO17000                    :rem 117
4500 PRINT"{DOWN}WHEW! YOU SAVED YOUR SKI
     N BUT LOST YOUR"             :rem 119
4505 PRINT"FOOD AND WATER":CH=CH-1 :rem 8
4510 GOTO4810                     :rem 208
4800 FD=10:WT=10:CH=CH+1          :rem 96
4810 PRINTLEFT$(D$,23) "PRESS {RED}RETURN
     {BLK}"                       :rem 80
4820 A$="":GETA$:IFA$<>CHR$(13)THEN4820
                                  :rem 142
4840 RETURN                       :rem 176
5000 REMSTORRM                    :rem 144
5010 IFRND(1)>.5THENRETURN        :rem 107
5015 POKE53281,12                 :rem 140
5020 PRINT"{CLR}TERRIBLE STORM"   :rem 25
5025 IFRND(1)>.9THEN PRINT"{DOWN}SHIPWREC
     K AND PERISH":GOTO17000      :rem 48
5030 PRINT"{DOWN}YOU RIDE IT OUT, BUT LOS
     E SUPPLIES":PRINT"{DOWN}OVERBOARD."
                                  :rem 13
5040 SP=SP-4                      :rem 173
5050 PRINTLEFT$(D$,23) "PRESS {RED}RETURN
     {BLK}"                       :rem 77
5060 A$="":GETA$:IFA$<>CHR$(13)THEN5060
                                  :rem 136
5070 POKE53281,3:RETURN           :rem 119
6000 REM ARAB DHOWS               :rem 69
6010 PRINT"{CLR}HOSTILE WATERS"   :rem 25
6020 PRINT"{DOWN}ARAB TRADERS WILL TRY TO
      KEEP YOU OUT"               :rem 201
6030 PRINTLEFT$(D$,23) "PRESS {RED}RETURN
     {BLK}"                       :rem 76
6040 A$="":GETA$:IFA$<>CHR$(13)THEN6040
                                  :rem 134
6045 IFRND(1)>.5THENRETURN        :rem 116
6050 PRINT"{CLR}{DOWN}";TAB(8)DH$:PRINT"
     {HOME}{DOWN}"TAB(20)DH$      :rem 244
6060 PRINT"{HOME}{9 DOWN}ARAB DHOWS APPEA
     R"                           :rem 157
6070 PRINT"{DOWN}TYPE {BLU}FLEE{BLK} AND
     {SPACE}PRESS RETURN"         :rem 252
6072 TI$="000000"                 :rem 49
6074 INPUTA$:IFA$<>"FLEE"THEN6074:rem 206
6076 IFTI<200THEN6090             :rem 203
6080 PRINT"{DOWN}THEY SINK YOU.":GOTO1700
     0                            :rem 181
6090 PRINT"{DOWN}YOUR PILOT ESCAPES THEM.
     "                            :rem 232
6100 PRINTLEFT$(D$,23) "PRESS {RED}RETURN
     {BLK}"                       :rem 74
6110 A$="":GETA$:IFA$<>CHR$(13)THEN6110
                                  :rem 130
6120 RETURN                       :rem 169
10000 REM LOG BOOK                :rem 226
10002 GOSUB16000:Q=INT(ML/200)    :rem 200
10003 IFQ>1THENPRINTLEFT$(D$,5);"
     {9 RIGHT}";LEFT$(DT$,3*Q)    :rem 93
10005 PRINT"{HOME}{BLK}{2 SPACES}SHIP'S R
     ECORD"                       :rem 215
10006 WK$=RIGHT$("{2 SPACES}"+STR$(WK),4)
                                  :rem 204
10007 ML$=RIGHT$("{2 SPACES}"+STR$(ML),4)
                                  :rem 187
10008 FD$=RIGHT$("{2 SPACES}"+STR$(FD),4)
                                  :rem 158
10009 SP$=RIGHT$("{2 SPACES}"+STR$(SP),4)
                                  :rem 209
10010 GP$=RIGHT$("{2 SPACES}"+STR$(GP),4)
                                  :rem 177
10011 CH$=RIGHT$("{2 SPACES}"+STR$(CH),4)
                                  :rem 154
10012 WT$=RIGHT$("{2 SPACES}"+STR$(WT),4)
                                  :rem 219
10020 PRINTLEFT$(D$,14)TAB(20)"WEEKS OUT
     {4 SPACES}";WK$              :rem 95
10030 PRINTTAB(20)"MILES SAILED ";ML$
                                  :rem 115
10040 PRINTTAB(20)"FOOD{9 SPACES}";FD$
                                  :rem 97
10042 PRINTTAB(20)"WATER{8 SPACES}";WT$
                                  :rem 223
10045 PRINTTAB(20)"SUPPLIES{5 SPACES}";SP
     $                            :rem 204
10050 PRINTTAB(20)"GOLD{9 SPACES}";GP$
                                  :rem 109
10060 PRINTTAB(20)"CREW SPIRIT{2 SPACES}"
     ;CH$                         :rem 72
10070 PRINT"{DOWN}"TAB(14)M$(G)   :rem 112
10073 IFG=1ORG=2THENPRINTTAB(14)"TOOK ON
     {SPACE}FOOD & WATER";        :rem 9
10074 IFG=4THENPRINTTAB(14)"AT MALINDI"
                                  :rem 99
```

```
10075 IFG=5THENPRINTTAB(10)"{DOWN}{RVS}YO
      U MADE IT!";:PRINT" {RVS}CONGRATULA
      TIONS!{HOME}";                  :rem 7
10077 IFG=5THEN PRINT"HISTORY WAS WRONG."
      :END                            :rem 48
10080 PRINTLEFT$(D$,24)"PRESS C"      :rem 52
10090 A$="":GETA$:IFA$<>"C"THEN10090
                                      :rem 183
10095 RETURN                          :rem 223
11000 REM EVAL SITUATION              :rem 190
11010 IFFD<1THENPRINT"{CLR}OUT OF FOOD":P
      RINT"{DOWN}YOU DIE OF STARVATION.":
      GOTO17000                       :rem 229
11020 IFSP<1THENPRINT"{CLR}OUT OF SUPPLIE
      S":PRINT"{DOWN}YOU DIE":GOTO17000
                                      :rem 126
11030 IFCH<2THENPRINT"{CLR}CREW MUTINIES.
      ":PRINT"{DOWN}THEY FORCE YOU TO TUR
      N BACK."                        :rem 202
11035 IFCH<2THEN17000                 :rem 169
11100 RETURN                          :rem 211
14000 POKE53281,7: REM WEATHER        :rem 184
14002 WH=FNR(7):G=0:GG=5:CM=M%(WH)*FG
                                      :rem 137
14005 PRINT"{CLR}{6 DOWN}WEATHER":rem 212
14010 IFWH=7THEN14140                 :rem 193
14030 PRINTWH$(WH):IFWH<3THENCH=CH-1
                                      :rem 239
14034 IFML<800ANDML+CM>800THENG=1:WT=10:F
      D=FD+3:SP=SP+6:IFFD<10THENFD=10
                                      :rem 170
14036 IFML<1500ANDML+CM>1500THENG=2:WT=10
      :FD=FD+3:SP=SP+6:IFFD<10THENFD=10
                                      :rem 9
14038 IFML<5000ANDML+CM>5000THENG=3
                                      :rem 57
14039 IFML<6600ANDML+CM>6600THENG=4
                                      :rem 73
14040 IFML>6600THENGG=6               :rem 91
14042 ML=ML+CM:Q=INT(ML/2+.5):IFML>9000TH
      ENG=5                           :rem 2
14045 GOTO14155                       :rem 55
14140 PRINT"{DOWN}STEADY RAIN":PRINT"
      {DOWN}YOU REFILL WATER TANKS":WT=10
                                      :rem 46
14155 PRINTLEFT$(D$,23) "PRESS {RED}RETUR
      N{BLK}"                         :rem 131
14157 A$="":GETA$:IFA$<>CHR$(13)THEN14157
                                      :rem 246
14200 RETURN                          :rem 215
15000 POKE53281,13:{3 SPACES}REM SETUP
                                      :rem 103
15010 PRINT"{CLR}HENRY THE NAVIGATOR, PRI
      NCE OF PORTUGAL,"               :rem 137
15015 PRINT"BELIEVES THERE MUST BE A SEA-
      ROUTE TO "                      :rem 221
15017 PRINT"{DOWN}INDIA. HE HAS OFFERED A
       PRIZE FOR"                     :rem 137
15018 PRINT"{DOWN}FINDING IT. VASCO DA GA
      MA IS GOING TO"                 :rem 117
15019 PRINT"{DOWN}TRY. HIS SHIPS WILL BE
      {SPACE}READY SOON. BUT"         :rem 6
15020 PRINT"{DOWN}YOU HAVE A SHIP THAT CA
      N LEAVE TODAY."                 :rem 143
15021 PRINT"{2 DOWN}YOU DECIDE TO TRY YOU
      R LUCK."                        :rem 213
15022 PRINTLEFT$(D$,23) "PRESS {RED}RETUR
      N{BLK}"                         :rem 125
15023 A$="":GETA$:IFA$<>CHR$(13)THEN15023
                                      :rem 232
15025 WT=10:GP=20                     :rem 182
15030 PRINT"{CLR}OTHERS HAVE TRIED. SOME
      {SPACE}DIED IN STORMS,"         :rem 241
15031 PRINT"{DOWN}SOME OF THIRST OR STARV
      ATION. SOME WERE"               :rem 196
15032 PRINT"{DOWN}MURDERED BY PIRATES, OT
      HERS BY NATIVES."               :rem 204
15033 PRINT"{DOWN}UNHAPPY CREWS MUTINIED.
       ARAB TRADERS"                  :rem 63
15034 PRINT"{DOWN}HAVE KILLED TO PROTECT
      {SPACE}THEIR ROUTES."           :rem 8
15035 PRINT"{DOWN}ALL THESE COULD HAPPEN
      {SPACE}TO YOU."                 :rem 51
15036 PRINT"{2 DOWN}{RVS}WORDS OF ADVICE
      {OFF}: NOT ALL STRANGE SHIPS":rem 5
15037 PRINT"{DOWN}HOLD PIRATES. NATIVES C
      AN BE FRIENDLY."                :rem 77
15038 PRINT"{DOWN}FRESH FOOD, GOOD WEATHE
      R, AN INCREASE"                 :rem 237
15039 PRINT"{DOWN}IN GOLD WILL KEEP YOUR
      {SPACE}CREW HAPPY."             :rem 46
15050 PRINTLEFT$(D$,23) "PRESS {RED}RETUR
      N{BLK}"                         :rem 126
15060 A$="":GETA$:IFA$<>CHR$(13)THEN15060
                                      :rem 234
15130 PRINT"{CLR}{DOWN}"TAB(30){2 SPACES}
      SS$                             :rem 196
15140 PRINT"{3 DOWN}PRESS {RVS}L{OFF} TO
      {SPACE}SET SAIL FROM LISBON":rem 47
15150 A$="":GETA$:IFA$<>"L"THEN15150
                                      :rem 196
15155 FORI=1TO30:PRINT"{HOME}"MS$:PRINT"
      {UP}"MS$:{12 SPACES}:NEXT       :rem 96
15160 SP=50:CH=10:FD=10               :rem 7
15180 PRINTLEFT$(D$,23) "PRESS {RED}RETUR
      N{BLK}"                         :rem 130
15185 A$="":GETA$:IFA$<>CHR$(13)THEN15185
                                      :rem 250
15200 RETURN                          :rem 216
16000 POKE53281,3:PRINT"{CLR}":IFML=0THEN
       PRINT"{CLR}{BLK} SEA ROUTE TO INDI
      A":POKE53280,3                  :rem 82
16004 PRINT"{BLK}RRRRRRRRRRRRRRRRRRRRRRR
      RRRRRRRRRRR{YEL}"               :rem 130
16005 REM PRINT"{YEL}"                :rem 79
16010 PRINTTAB(8)"{RVS}{10 SPACES}{OFF}
      {7 SPACES}{RVS}{7 SPACES}" :rem 223
16012 PRINTTAB(8)"{RVS}{2 SPACES}{OFF}£
      {3 SPACES}[2 J] {RVS}{19 SPACES}"
                                      :rem 244
16014 PRINTTAB(8)"[2 U]{3 SPACES}[C]
      {2 SPACES}{RVS}{23 SPACES}":rem 107
16016 PRINTTAB(4)"[B]{3 SPACES}{RVS}£
      {3 SPACES}{OFF}{4 SPACES}{RVS}
      {19 SPACES}"                    :rem 73
16018 PRINTTAB(7)"{RVS}£{8 SPACES}{OFF}
      [*]{RVS}{12 SPACES}{OFF}£"
                                      :rem 169
16020 PRINTTAB(6)"{RVS}£{9 SPACES}{OFF}
      {SPACE}[*]{RVS}{2 SPACES}[*]
      {OFF}{2 SPACES}[*]{RVS}{4 SPACES}
      {OFF}£"                         :rem 3
16022 PRINTTAB(3)"[B]{2 SPACES}{RVS}
      {10 SPACES}[*]{OFF} [*]{RVS}
      {2 SPACES}[*]{OFF}{2 SPACES}{RVS}
      {3 SPACES}{OFF}£"               :rem 24
16024 PRINTTAB(6)"{RVS}{11 SPACES}[*]
      {OFF} [*]{RVS} {OFF}£{2 SPACES}
      {RVS}{2 SPACES}{OFF}£"          :rem 40
16026 PRINTTAB(6)"{RVS}{12 SPACES}[*]
      {OFF} £{3 SPACES}[*]{RVS} "
                                      :rem 75
```

```
16028 PRINTTAB(6)"£*³{RVS}{12 SPACES}
      {OFF}£{5 SPACES}£*³"          :rem 59
16030 PRINTTAB(10)"£*³{RVS}{7 SPACES}
      {OFF}£"                       :rem 128
16032 PRINTTAB(11)"{RVS}{6 SPACES}{OFF}£
      "                             :rem 164
16034 PRINTTAB(11)"{RVS}{6 SPACES}{OFF}"
                                    :rem 253
16036 PRINTTAB(11)"{RVS}{6 SPACES}{OFF}"
                                    :rem 255
16038 PRINTTAB(11)"{RVS}{6 SPACES}{OFF}"
                                     :rem 1
16039 PRINTTAB(11)"{RVS}{5 SPACES}{OFF}£
      "                             :rem 171
16040 PRINTTAB(11)"{RVS}{5 SPACES}{OFF}
      {RVS}£H³"                     :rem 192
16042 PRINTTAB(11)"{RVS}{4 SPACES}{OFF}£
       {RVS} "                      :rem 183
16044 PRINTTAB(11)"{RVS}{3 SPACES}{OFF}£
      {2 SPACES}{RVS} "             :rem 185
16046 PRINTTAB(11)"£*³{RVS} {OFF}£"
                                    :rem 136
16048 IFML>0THENPRINT"{BLK}";:RETURN
                                    :rem 126
16050 PRINTTAB(20)"{DOWN}{BLK}PRESS B
      {OFF} TO BEGIN";              :rem 156
16060 A$="":GETA$:IFA$<>"B"THEN16060
                                    :rem 188
16070 RETURN                        :rem 222
17000 PRINTLEFT$(D$,23) "PRESS {RED}RETUR
      N{BLK}"                       :rem 123
17010 A$="":GETA$:IFA$<>CHR$(13)THEN17010
                                    :rem 228
17020 PRINT"{CLR}{2 DOWN}ON MAY 20, 1498"
      :PRINT"{DOWN}VASCO DA GAMA REACHED
      {SPACE}CALCUTTA ON THE"       :rem 86
17025 PRINT"{DOWN}WEST COAST OF INDIA, AF
      TER{2 SPACES}"                :rem 54
17030 PRINT"{DOWN}A VOYAGE OF 11 MONTHS A
      ND 9500 MILES.{4 DOWN}"       :rem 70
17040 PRINT"{DOWN}PLAY AGAIN? Y OR N"
                                    :rem 251
17050 A$="":GETA$:IFA$="Y"THENRUN:rem 142
17060 IFA$<>"N"THEN17050            :rem 42
17070 PRINT"{CLR}BETTER LUCK ANOTHER TIME
      ."                            :rem 198
17080 END                           :rem 215
```

# The Inner World Of Computers

*(Article on page 110.)*

BEFORE TYPING...

Before typing in programs, please refer to "How To Type COMPUTE!'s Gazette Programs," "A Beginner's Guide To Typing In Programs," and "The Automatic Proofreader" that appear before the Program Listings.

```
10 PRINT"{CLR}{BLK} TO CREATE {RVS}BINAMI
   TE"                              :rem 7
20 PRINT" PERSON/ITEM PROFILE"    :rem 121
30 PRINT" PUT {RVS}.{OFF} IN ALPHA CELL"
                                    :rem 28
40 PRINT"{5 SPACES}{RVS}ABCDEFGHIJKLMNO":
   DIMD(16),NA$(16),B$(16)          :rem 112
50 PRINT"{5 SPACES}{RVS}PPPPPPPPPPPPPPP"
                                    :rem 120
60 PRINT"HOW MANY NAME/ITEMS":INPUTTTT:IFT
   T>14THENTT=14                    :rem 43
70 FORI=0TOTT-1:READNA$(I):NEXT     :rem 201
80 PRINT"{CLR} IF ATTRIBUTE APPLIES"
                                    :rem 29
90 PRINT" PUT {RVS}.{OFF} IN ALPHA CELL"
                                    :rem 34
100 PRINT"IN ROW OPPOSITE NAME #":rem 168
110 PRINT"{RVS}NAME{OFF} ABCDEFGHIJKLMNO"
                                    :rem 96
120 INPUT" #{3 SPACES}{RVS}OOOOOOOOOOOOO
   O{17 LEFT}";B$:L=LEN(B$)         :rem 36
130 XP=L-1:FORI=1TOL:DM$=MID$(B$,I,1):IFD
   M$>"1"THENPRINT"{2 UP}{2 SPACES}":DM$
   ="":GOTO120                      :rem 102
140 IFDM$="."THENDM$="1"            :rem 30
150 D=D+(VAL(DM$))*2↑XP:XP=XP-1:NEXT
                                    :rem 192
160 GOSUB300:T=T+1                  :rem 37
170 PRINT"{UP} "TAB(2)T:B$="":D=0:D$="":I
   F T<TT THEN 120                  :rem 154
180 PRINT"{5 SPACES}ABCDEFGHIJKLMNO":PRIN
   T"TO FIND ATTRIBUTES":PRINT"TYPE NAME
    #'S"                            :rem 131
190 PRINT"( 1..."TT") ANY ORDER"    :rem 199
200 PRINT:INPUT"#";AL:AL=AL-1:PRINT"{UP}"
   :IFAL<0ORAL>TTTHEN240            :rem 62
210 FORJ=0TO14                      :rem 57
220 IF(D(AL)AND2↑J)=2↑J THENPRINTCHR$(79-
   J);                              :rem 138
230 NEXT:GOTO200                    :rem 217
240 FORI=0TOTT-1:PRINTLEFT$(NA$(I),5)"
   {RVS}"B$(I):PRINT"{RVS}"D(I):NEXT
                                    :rem 187
250 PRINT"{5 SPACES}ABCDEFGHIJKLMNO"
                                    :rem 160
260 END:REM*SAVE DATA*              :rem 42
300 B$(T)=B$:D(T)=D:                :rem 31
310 IFLEN(B$(T))<15THENB$(T)=LEFT$("
   {16 SPACES}",15-LEN(B$(T)))+B$(T)
                                    :rem 139
320 RETURN                          :rem 117
400 DATAMARYMARY,BOPEEP,BOYBLUE,MSMUFFIT,
   5NAME,6NAME,7NAME,8NAME,9NAME,10NAME
                                    :rem 30
410 DATA11NAME,12NAME,13NAME,14NAME,15NAM
   E,16NAME                         :rem 172
```

# Getting Started With A Disk Drive

*(Article on page 106.)*

```
1 FORI=828TO883:READA:POKEI,A:NEXTI
                                    :rem 254
10 REM"D=DSAVE"@BACK2",D0:?DS$:CATALOGD0
                                    :rem 159
20 BB=PEEK(44)+27:POKE995,BB        :rem 85
30 POKE998,PEEK(55):POKE999,PEEK(56):POKE
   55,0:POKE56,BB:CLR               :rem 55
40 BB=PEEK(995)                     :rem 66
50 N=PEEK(999)-BB-1:BA=BB*256:MA=828
                                    :rem 12
60 DIMBM%(35,24)                    :rem 63
70 FORJ=0TO7:TA(J)=2↑J:NEXT        :rem 217
80 PRINT"{CLR}{3 RIGHT}{RVS}BACKUP 1541
   {OFF}"                           :rem 72
90 PRINT"{DOWN}'GOTO10000' IF PROGRAM QUI
   TS ABNORMALLY"                   :rem 241
```

```
100 PRINT"{DOWN}"N"BUFFERS AVAILABLE"
                                    :rem 147
110 OPEN1,8,15                      :rem 235
200 REM *** MAIN FUNCTIONS ****     :rem 122
210 GOSUB1000                       :rem 212
220 D$="S":GOSUB3200:I2$=IR$        :rem 36
230 IFDR$<>"2A"THENPRINT"{RVS}ILLEGAL DOS
    1.0 DISK{OFF}":GOTO10000        :rem 177
240 IFI2$=I1$THENPRINT"{RVS}SOURCE AND DE
    STINATION HAVE SAME ID CODE{OFF}":GOT
    O10000                          :rem 127
250 GOSUB2500                       :rem 222
260 T=TS:S=0:NU=1:T1=T:S1=S         :rem 179
270 PRINT#1,"I0":OPEN3,8,3,"#"      :rem 88
280 PRINT"READING BLOCK #";         :rem 46
290 IFBM%(T1,S1)=0THENGOSUB2000:NU=NU+1:I
    FNU>NTHEN320                    :rem 23
300 S1=S1+1:IFS1>20THENS1=0:T1=T1+1
                                    :rem 30
310 IFT1<TF+1THEN290                :rem 164
320 PRINT"{DOWN}"                   :rem 119
330 CLOSE3                          :rem 63
340 D$="D":GOSUB3200:IFIR$<>I1$THENGOTO34
    0                               :rem 226
350 PRINT#1,"I0":OPEN3,8,3,"#"      :rem 87
360 PRINT"WRITING BUFFER #";        :rem 166
370 NU=1:T1=T:S1=S                  :rem 73
380 IFBM%(T1,S1)=0THENGOSUB2200:NU=NU+1:I
    FNU>NTHEN410                    :rem 25
390 S1=S1+1:IFS1>20THENS1=0:T1=T1+1
                                    :rem 39
400 IFT1<TF+1THEN380                :rem 164
410 PRINT"{DOWN}"                   :rem 119
420 CLOSE3                          :rem 63
430 S=S1+1:IFS>20THENS=0:T1=T1+1    :rem 143
440 T=T1:IFT>TFTHEN500              :rem 103
450 D$="S":GOSUB3200:IFIR$<>I2$THEN450
                                    :rem 189
460 NU=1:T1=T:S1=S:GOTO270          :rem 85
500 REM FINISHED XFERS              :rem 75
510 CLOSE1                          :rem 61
520 POKE55,PEEK(998):POKE56,PEEK(999):CLR
                                    :rem 184
530 PRINT"{2 DOWN}BACKUP COMPLETE"
                                    :rem 154
540 OPEN1,8,0,"$0"                  :rem 128
550 GET#1,A$:IFA$<>"{RVS}"THEN550   :rem 38
560 PRINTA$;:GOTO610                :rem 210
570 GET#1,A$:SS=ST:A=LEN(A$):IFATHENA=ASC
    (A$)                            :rem 182
580 GET#1,B$:SS=ST:B=LEN(B$):IFBTHENA=ASC
    (B$)                            :rem 188
590 IFSSTHEN660                     :rem 158
600 IFA=1ANDB=1THENGOSUB630         :rem 159
610 GET#1,A$:IFA$=""THENPRINT:GOTO570
                                    :rem 214
620 PRINTA$;:GOTO610                :rem 207
630 GET#1,A$:SS=ST:A=LEN(A$):IFATHENA=ASC
    (A$)                            :rem 179
640 GET#1,B$:SS=ST:B=LEN(B$):IFBTHENB=ASC
    (B$)                            :rem 186
650 N=B*256+A:PRINTN;:RETURN        :rem 5
660 CLOSE1                          :rem 67
670 END                             :rem 116
1000 REM HEADER DEST DISK           :rem 169
1010 PRINT"{DOWN}INSERT DESTINATION DISK
     {SPACE}TO BE FORMATTED"        :rem 182
1020 INPUT"{2 DOWN}DISK NAME{3 RIGHT}
     {SHIFT-SPACE}{16 SPACES}{19 LEFT}";D
     N$                             :rem 148
1030 IFDN$="{SHIFT-SPACE}"THENPRINT"
     {3 UP}";:GOTO1020              :rem 78
1040 IFLEN(DN$)>16THENCLR:GOTO40    :rem 198
1050 F=0:FORJ=1TOLEN(DN$):S1$=MID$(DN$,J,
     1)                             :rem 210
1060 IFS1$="{SHIFT-SPACE}"ORS1$=CHR$(34)T
     HENF=1                         :rem 65
1070 NEXTJ:IFFTHENPRINT"{3 UP}";:GOTO1020
                                    :rem 132
1080 INPUT"{DOWN}UNIQUE DISK ID{3 RIGHT}
     {SHIFT-SPACE}{20 SPACES}{23 LEFT}";I
     1$                             :rem 40
1090 IFI1$="{SHIFT-SPACE}"THENPRINT"
     {2 UP}";:GOTO1080              :rem 177
1100 IFLEN(I1$)<>2THENPRINT"{2 UP}";:GOTO
     1080                           :rem 100
1110 PRINT#1,"N0:"+DN$+","+I1$      :rem 17
1120 GOSUB3000                      :rem 7
1130 IFERTHENPRINTER$:GOTO10000     :rem 198
1140 RETURN                         :rem 166
2000 REM READ BLOCK T1,S1 TO BUFFER # NU
                                    :rem 133
2010 C=.                            :rem 113
2020 PRINT#1,"U1";3;0;T1;S1         :rem 243
2030 GOSUB3000:IFNOTERTHEN2060      :rem 80
2040 C=C+1:IFC<3GOTO2020            :rem 93
2050 PRINTER$:FORJ=(BB+NU)*256TO(BB+NU)*2
     56+255:POKEJ,.:NEXTJ:GOTO2100
                                    :rem 251
2060 PRINT#1,"B-P";3;0              :rem 177
2070 IFNU<>0THENPRINT"{3 SPACES}{3 LEFT}"
     ;RIGHT$("{2 SPACES}"+STR$(NU),3);"
     {3 LEFT}";                     :rem 26
2080 POKE996,PEEK(3):POKE997,PEEK(4):POKE
     4,BB+NU:SYSMA                  :rem 64
2085 POKE3,PEEK(996):POKE4,PEEK(997)
                                    :rem 99
2090 IFST<>.ANDST<>64THENGOSUB3000:GOTO20
     50                             :rem 179
2100 RETURN                         :rem 163
2200 REM WRITE BLOCK T1,S1 FROM BUFFER #
     {SPACE}NU                      :rem 135
2210 C=.                            :rem 115
2220 PRINT#1,"B-A";0;T1;S1:PRINT#1,"B-P";
     3;0                            :rem 212
2230 PRINT"{3 SPACES}{3 LEFT}";RIGHT$("
     {2 SPACES}"+STR$(NU),3);"{3 LEFT}";
                                    :rem 13
2240 POKE996,PEEK(3):POKE997,PEEK(4):POKE
     4,BB+NU:SYSMA+3                :rem 156
2245 POKE3,PEEK(996):POKE4,PEEK(997)
                                    :rem 97
2250 IFST<>.ANDST<>64THENPRINT"{RVS}IEEE
     {SPACE}WRITE ERROR"ST"{OFF}":GOTO100
     00                             :rem 37
2260 PRINT#1,"U2";3;0;T1;S1         :rem 250
2270 GOSUB3000:IFNOTERTHEN2300      :rem 83
2280 C=C+1:IFC<3THEN2260            :rem 95
2290 PRINT"{RVS}UNRECOVERABLE WRITE ERROR
     "ER$:GOTO10000                 :rem 177
2300 RETURN                         :rem 165
2500 REM GET BAM TO BM%(T,S)        :rem 214
2510 TS=1:TF=.                      :rem 28
2520 PRINT#1,"I0":OPEN3,8,3,"#"     :rem 136
2530 S9=0                           :rem 195
2540 PRINT"{DOWN}TRACK #{3 SPACES}BLOCKS
     {SPACE}TO XFER"                :rem 219
2550 PRINT"[24 T]"                  :rem 229
2560 NU=0:T1=18:S1=0:C0$=CHR$(.):GOSUB200
     0                              :rem 119
2570 BY=4                           :rem 218
2580 T%=(BY-4)/4+1                  :rem 145
```

```
2590 PRINT"{2 SPACES}";T%;          :rem 144
2600 IFPEEK(BA+BY)=.THENFORJ=.TO20:BM%(T%
     ,J)=.:NEXT:BY=BY+4:GOTO2650 :rem 175
2610 S=0                            :rem 137
2620 BY=BY+1:A0=PEEK(BA+BY):FORJ=.TO7:BM%
     (T%,S)=A0ANDTA(J):S=S+1:NEXT:rem 202
2630 IFS<22THEN2620                 :rem 70
2640 BY=BY+1                        :rem 155
2650 ES=21:IFT%>17THENES=19         :rem 91
2660 IFT%>24THENES=18               :rem 231
2670 IFT%>30THENES=17               :rem 228
2680 FORJ=ESTO24:BM%(T%,J)=-1:NEXT:rem 33
2690 SM=.:FORJ=.TO20:IFBM%(T%,J)=.THENSM=
     SM+1                           :rem 231
2700 NEXT:PRINTTAB(12);SM:S9=S9+SM
                                    :rem 143
2710 IFSM=.ANDTS=T%THENTS=TS+1:GOTO2730
                                    :rem 233
2720 IFSM<>.THENTF=T%               :rem 33
2730 IFBY<143THEN2580               :rem 200
2740 CLOSE3                         :rem 118
2750 PRINT"START =";TS;" FINISH =";TF
                                    :rem 158
2760 PRINT"{DOWN}A TOTAL OF";S9;"BLOCKS T
     O XFER"                        :rem 231
2770 S8=90+25+(.650+.980)*S9        :rem 136
2780 S7=INT(S8/60):PRINT"APPROX";S7":"INT
     (S8-S7*60);"FOR COPY"          :rem 203
2790 RETURN                         :rem 178
3000 REM READ ERR CH TO ER,ER$      :rem 88
3010 INPUT#1,E0$,E1$,E2$,E3$:ER$=E0$+","+
     E1$+","+E2$+","+E3$            :rem 176
3020 ER=LEN(E0$):IFERTHENER=VAL(E0$)
                                    :rem 146
3030 RETURN                         :rem 166
3200 REM INSTRUCT TO SWAP TO DISK GIVEN I
     N D$                           :rem 73
3210 IFD$="D"THENS1$="DESTINATION":GOTO32
     30                             :rem 87
3220 S1$="SOURCE"                   :rem 193
3230 PRINT"{DOWN}INSERT ";S1$;" DISK, PRE
     SS {RVS}SPACE{OFF}"            :rem 213
3240 GETA$:IFA$<>" "THEN3240        :rem 242
3250 OPEN2,8,0,"$0"                 :rem 178
3260 GOSUB3000:IFER>0THEN10000      :rem 252
3270 FORJ=1TO26:GET#2,A$:NEXTJ      :rem 57
3280 GET#2,A$:GET#2,B$:IR$=A$+B$    :rem 192
3290 GET#2,A$:GET#2,A$:GET#2,B$:DR$=A$+B$
                                    :rem 188
3300 CLOSE2:RETURN                  :rem 136
10000 REM DROP OUT                  :rem 2
10010 POKE55,PEEK(998):POKE56,PEEK(999):C
      LR:STOP                       :rem 147
15000 DATA 76,66,3,76,91,3,162,3,32,198,2
      55,160,0,132,3,32,207,255,145
                                    :rem 113
15010 DATA 3,165,144,208,3,200,208,244,32
      ,204,255,96,162,3,32,201,255,160
                                    :rem 245
15020 DATA 0,132,3,177,3,32,210,255,165,1
      44,208,3,200,208,244,32,204,255,96
                                    :rem 87
```

BEFORE TYPING...

Before typing in programs, please refer to "How To Type COMPUTE!'s Gazette Programs," "A Beginner's Guide To Typing In Programs," and "The Automatic Proofreader" that appear before the Program Listings.

# Dynamic SAVE For VIC And 64

*(Article on page 120.)*

## Program 1: Dynamic SAVE For Tape

```
60000 REM TAPE SAVER                :rem 133
60010 Q$=CHR$(34):N$="DYNAMIC SAVE"
                                    :rem 195
60019 REM USE EITHER 60020 OR 60021
                                    :rem 36
60020 POKE36879,27:C1$="{WHT}":C2$="{BLU}
      ":REM FOR VIC-20              :rem 14
60021 REM POKE53280,14:POKE53281,6:C1$="
      {BLU}":C2$="{7}":REM FOR COMMODOR
      E 64                          :rem 225
60030 PRINTC1$"{CLR}FORQ=1TO2:SAVE"Q$N$Q$
      ":NEXT"C2$                    :rem 139
60040 POKE631,19:POKE632,13         :rem 237
60050 POKE633,86:POKE634,101:POKE635,58:P
      OKE636,86:POKE637,101:POKE198,7:END
                                    :rem 172
```

## Program 2: Dynamic SAVE For Disk

```
60000 REM DISK SAVER                :rem 134
60010 Q$=CHR$(34):N$="DYNAMIC SAVE"
                                    :rem 195
60019 REM USE EITHER 60020 OR 60021
                                    :rem 36
60020 POKE36879,27:C1$="{WHT}":C2$="{BLU}
      ":REM FOR VIC-20              :rem 14
60021 REM POKE53280,14:POKE53281,6:C1$="
      {BLU}":C2$="{7}":REM FOR COMMODOR
      E 64                          :rem 225
60030 PRINTC1$"{CLR}SAVE"Q$"@0:"N$Q$",8:V
      ERIFY"Q$N$Q$",8"C2$           :rem 194
60040 POKE631,19:POKE632,13:POKE198,2:END
                                    :rem 103
```

# CUT-OFF!

*(Article on page 46.)*

## Program 1:
### Tiny MLX—Special VIC Version

```
100 POKE55,174:POKE56,23:CLR:POKE788,194
                                    :rem 76
210 S=6063:E=7658                   :rem 136
300 PRINT"{CLR}";CHR$(14):AD=S      :rem 56
310 PRINTRIGHT$("0000"+MID$(STR$(AD),2),5
    );":";:FORJ=1TO6                :rem 234
320 GOSUB570:IFN=-1THENJ=J+N:GOTO320
                                    :rem 228
480 IFN<0THENPRINT:GOTO310          :rem 168
490 A(J)=N:NEXTJ                    :rem 199
500 CKSUM=AD-INT(AD/256)*256:FORI=1TO6:CK
    SUM=(CKSUM+A(I))AND255:NEXT     :rem 200
510 PRINTCHR$(18);:GOSUB570:PRINTCHR$(20)
                                    :rem 234
515 IFN=CKSUMTHEN530                :rem 255
520 PRINT:PRINT"LINE ENTERED WRONG":PRINT
    "RE-ENTER":PRINT:GOSUB1000:GOTO310
                                    :rem 129
530 GOSUB2000                       :rem 218
540 FORI=1TO6:POKEAD+I-1,A(I):NEXT:rem 80
550 AD=AD+6:IFAD<ETHEN310           :rem 212
560 GOTO710                         :rem 108
```

```
570 N=0:Z=0                                :rem 88
580 PRINT"{+}";                            :rem 79
581 GETA$:IFA$=""THEN581                   :rem 95
585 PRINTCHR$(20);:A=ASC(A$):IFA=13ORA=44
    ORA=32THEN670                          :rem 229
590 IFA>128THENN=-A:RETURN                 :rem 137
600 IFA<>20 THEN 630                       :rem 10
610 GOSUB690:IFI=1ANDT=44THENN=-1:PRINT"
    {LEFT} {LEFT}";:GOTO690                :rem 172
620 GOTO570                                :rem 109
630 IFA<48ORA>57THEN580                    :rem 105
640 PRINTA$;:N=N*10+A-48                   :rem 106
650 IFN>255 THEN A=20:GOSUB1000:GOTO600
                                           :rem 229
660 Z=Z+1:IFZ<3THEN580                     :rem 71
670 IFZ=0THENGOSUB1000:GOTO570             :rem 114
680 PRINT",";:RETURN                       :rem 240
690 S%=PEEK(209)+256*PEEK(210)+PEEK(211)
                                           :rem 149
692 FORI=1TO3:T=PEEK(S%-I)                 :rem 68
695 IFT<>44ANDT<>58THENPOKES%-I,32:NEXT
                                           :rem 205
700 PRINTLEFT$("{3 LEFT}",I-1);:RETURN
                                           :rem 7
710 PRINT"{CLR}{RVS}*** SAVE ***{3 DOWN}"
                                           :rem 236
720 INPUT"{DOWN} FILENAME";F$              :rem 228
730 PRINT:PRINT"{2 DOWN}{RVS}T{OFF}APE OR
    {RVS}D{OFF}ISK: (T/D)"                 :rem 228
740 GETA$:IFA$<>"T"ANDA$<>"D"THEN740
                                           :rem 36
750 DV=1-7*(A$="D"):IFDV=8THENF$="0:"+F$
                                           :rem 158
760 T$=F$:ZK=PEEK(53)+256*PEEK(54)-LEN(T$
    ):POKE782,ZK/256                       :rem 3
762 POKE781,ZK-PEEK(782)*256:POKE780,LEN(
    T$):SYS65469                           :rem 109
763 POKE780,1:POKE781,DV:POKE782,1:SYS654
    66                                     :rem 69
765 POKE254,S/256:POKE253,S-PEEK(254)*256
    :POKE780,253                           :rem 12
766 POKE782,E/256:POKE781,E-PEEK(782)*256
    :SYS65496                              :rem 124
770 IF(PEEK(783)AND1)OR(ST AND191)THEN780
                                           :rem 111
775 PRINT"{DOWN}DONE.":END                 :rem 106
780 PRINT"{DOWN}ERROR ON SAVE.{2 SPACES}T
    RY AGAIN.":IFDV=1THEN720               :rem 171
781 OPEN15,8,15:INPUT#15,E1$,E2$:PRINTE1$
    ;E2$:CLOSE15:GOTO720                   :rem 103
782 GOTO720                                :rem 115
845 POKE780,1:POKE781,DV:POKE782,1:SYS654
    66                                     :rem 70
1000 REM BELL TONE                         :rem 250
1001 POKE36878,15:POKE36874,190            :rem 206
1002 FORW=1TO300:NEXTW                     :rem 117
1003 POKE36878,0:POKE36874,0:RETURN
                                           :rem 74
2000 REM BELL SOUND                        :rem 78
2001 FORW=15TO0STEP-1:POKE36878,W:POKE368
     76,240:NEXTW                          :rem 22
2002 POKE36876,0:RETURN                    :rem 119
```

## Program 2: CUT-OFF!—VIC Version

```
6063 :032,081,025,076,181,023,081
6069 :173,066,003,024,105,001,041
6075 :141,066,003,173,067,003,128
6081 :105,000,141,067,003,174,171
6087 :060,003,032,020,027,169,254
6093 :001,141,065,003,032,078,013
6099 :027,162,002,161,247,201,243
6105 :032,240,032,032,029,027,097
6111 :032,143,029,032,090,026,063
6117 :173,068,003,201,000,240,146
6123 :080,173,069,003,201,000,249
6129 :240,073,032,097,024,032,227
6135 :092,028,076,181,023,032,167
6141 :029,027,169,000,141,065,172
6147 :003,032,227,027,162,000,198
6153 :161,247,201,032,240,032,154
6159 :032,029,027,032,143,029,051
6165 :032,090,026,173,068,003,157
6171 :201,000,240,029,173,069,227
6177 :003,201,000,240,022,032,019
6183 :097,024,032,092,028,076,132
6189 :181,023,032,029,027,173,254
6195 :031,145,041,032,240,006,034
6201 :076,181,023,076,007,029,193
6207 :173,031,145,041,032,240,213
6213 :249,162,250,032,020,027,041
6219 :173,031,145,041,032,208,193
6225 :249,173,031,145,041,032,240
6231 :240,249,162,250,032,020,016
6237 :027,076,181,023,169,147,204
6243 :032,210,255,169,025,141,163
6249 :015,144,162,000,169,160,243
6255 :157,000,030,169,000,157,112
6261 :000,150,232,224,022,208,185
6267 :241,162,000,169,160,157,244
6273 :228,031,169,000,157,228,174
6279 :151,232,224,022,208,241,189
6285 :169,000,133,253,169,030,127
6291 :133,254,169,000,133,251,063
6297 :169,150,133,252,162,000,251
6303 :169,160,160,000,145,253,022
6309 :169,000,145,251,160,021,143
6315 :169,160,145,253,169,000,043
6321 :145,251,024,165,253,105,096
6327 :022,133,253,165,254,105,091
6333 :000,133,254,024,165,251,248
6339 :105,022,133,251,165,252,099
6345 :105,000,133,252,232,224,123
6351 :023,208,205,169,004,162,210
6357 :007,157,000,150,232,224,215
6363 :015,208,248,162,000,189,017
6369 :072,025,240,006,157,007,220
6375 :030,232,208,245,169,006,097
6381 :141,248,150,169,002,141,064
6387 :001,151,169,081,141,248,010
6393 :030,169,087,141,001,031,196
6399 :169,000,141,066,003,141,007
6405 :067,003,169,007,141,075,211
6411 :003,169,020,141,074,003,165
6417 :169,001,141,070,003,169,058
6423 :031,141,071,003,169,248,174
6429 :141,072,003,169,030,141,073
6435 :073,003,024,162,000,160,201
6441 :002,032,240,255,174,061,037
6447 :003,173,062,003,032,205,013
6453 :221,024,162,000,160,016,124
6459 :032,240,255,174,063,003,058
6465 :173,064,003,032,205,221,251
6471 :096,131,149,148,173,143,143
6477 :134,134,161,000,169,000,163
6483 :141,061,003,141,062,003,238
6489 :141,063,003,141,064,003,248
6495 :169,081,141,077,003,169,223
6501 :087,141,076,003,032,097,025
6507 :024,169,004,162,002,157,113
6513 :154,150,232,224,020,208,077
```

```
6519 :248,162,002,157,198,150,012
6525 :232,224,020,208,248,141,174
6531 :018,151,162,000,189,052,191
6537 :026,240,006,157,156,030,240
6543 :232,208,245,162,000,189,155
6549 :071,026,240,006,157,200,081
6555 :030,232,208,245,169,048,063
6561 :141,060,003,141,018,031,043
6567 :162,100,032,020,027,173,169
6573 :031,145,041,012,201,008,099
6579 :240,033,201,004,240,010,139
6585 :173,031,145,041,032,240,079
6591 :063,076,167,025,173,060,243
6597 :003,056,233,001,201,047,226
6603 :240,028,141,060,003,141,048
6609 :018,031,076,167,025,173,187
6615 :060,003,024,105,001,201,097
6621 :058,240,020,141,060,003,231
6627 :141,018,031,076,167,025,173
6633 :169,057,141,060,003,141,036
6639 :018,031,076,167,025,169,213
6645 :048,141,060,003,141,018,144
6651 :031,076,167,025,173,031,242
6657 :145,041,032,240,249,162,102
6663 :250,032,020,027,173,060,057
6669 :003,056,233,048,170,169,180
6675 :050,141,060,003,224,000,241
6681 :240,013,173,060,003,056,058
6687 :233,005,141,060,003,202,163
6693 :076,023,026,032,097,024,059
6699 :169,010,141,068,003,141,063
6705 :069,003,096,013,015,022,011
6711 :005,032,019,020,009,003,143
6717 :011,032,021,016,047,004,192
6723 :015,023,014,000,020,015,154
6729 :032,016,009,003,011,032,176
6735 :004,009,006,006,009,003,116
6741 :021,012,020,025,000,173,080
6747 :070,003,205,072,003,240,172
6753 :003,076,230,026,173,071,164
6759 :003,205,073,003,208,121,204
6765 :173,075,003,174,074,003,099
6771 :201,014,240,012,201,007,022
6777 :240,015,201,013,240,018,080
6783 :201,011,240,021,224,044,100
6789 :240,024,076,230,026,224,185
6795 :020,240,017,076,230,026,236
6801 :224,012,240,010,076,230,169
6807 :026,224,021,240,003,076,229
6813 :230,026,173,061,003,024,162
6819 :109,066,003,141,061,003,034
6825 :173,062,003,105,000,141,141
6831 :062,003,173,067,003,024,251
6837 :109,062,003,141,062,003,049
6843 :174,068,003,202,142,068,076
6849 :003,173,063,003,024,109,056
6855 :066,003,141,063,003,173,136
6861 :064,003,105,000,141,064,070
6867 :003,173,067,003,024,109,078
6873 :064,003,141,064,003,174,154
6879 :069,003,202,142,069,003,199
6885 :096,173,065,003,010,170,234
6891 :189,061,003,024,109,066,175
6897 :003,157,061,003,189,062,204
6903 :003,105,000,157,062,003,065
6909 :173,067,003,024,125,062,195
6915 :003,157,062,003,174,065,211
6921 :003,189,068,003,056,233,049
6927 :001,157,068,003,096,160,244
6933 :000,200,208,253,202,208,068
6939 :248,096,174,065,003,188,033
6945 :076,003,138,010,170,181,099
6951 :247,157,070,003,181,248,177
6957 :157,071,003,152,129,247,036
6963 :181,248,024,105,120,149,110
6969 :248,224,002,208,008,169,148
6975 :006,129,247,032,180,029,174
6981 :096,169,002,129,247,032,232
6987 :180,029,096,162,127,142,043
6993 :034,145,173,032,145,041,139
6999 :128,074,074,141,078,003,073
7005 :162,255,142,034,145,173,236
7011 :031,145,041,028,013,078,179
7017 :003,074,074,174,065,003,242
7023 :201,014,240,018,201,007,024
7029 :240,038,201,013,240,058,139
7035 :201,011,240,078,189,074,148
7041 :003,076,111,027,157,074,065
7047 :003,138,010,170,056,189,189
7053 :070,003,233,022,149,247,097
7059 :189,071,003,233,000,149,024
7065 :248,076,226,027,157,074,193
7071 :003,138,010,170,024,189,181
7077 :070,003,105,001,149,247,228
7083 :189,071,003,105,000,149,176
7089 :248,076,226,027,157,074,217
7095 :003,138,010,170,024,189,205
7101 :070,003,105,022,149,247,017
7107 :189,071,003,105,000,149,200
7113 :248,076,226,027,157,074,241
7119 :003,138,010,170,056,189,005
7125 :070,003,233,001,149,247,148
7131 :189,071,003,233,000,149,096
7137 :248,096,174,065,003,165,208
7143 :197,201,012,240,018,201,076
7149 :021,240,038,201,044,240,253
7155 :058,201,020,240,078,189,005
7161 :074,003,076,232,027,157,050
7167 :074,003,138,010,170,056,194
7173 :189,070,003,233,022,149,159
7179 :247,189,071,003,233,000,242
7185 :149,248,076,091,028,157,254
7191 :074,003,138,010,170,024,186
7197 :189,070,003,105,001,149,034
7203 :247,189,071,003,105,000,138
7209 :149,248,076,091,028,157,022
7215 :074,003,138,010,170,024,210
7221 :189,070,003,105,022,149,079
7227 :247,189,071,003,105,000,162
7233 :149,248,076,091,028,157,046
7239 :074,003,138,010,170,056,010
7245 :189,070,003,233,001,149,210
7251 :247,189,071,003,233,000,058
7257 :149,248,096,162,000,189,165
7263 :231,028,240,006,157,159,148
7269 :030,232,208,245,162,000,210
7275 :189,244,028,240,006,157,203
7281 :203,030,232,208,245,162,169
7287 :000,189,000,029,240,006,071
7293 :157,054,031,232,208,245,028
7299 :162,000,189,000,029,240,239
7305 :006,157,064,031,232,208,067
7311 :245,169,004,162,000,157,112
7317 :155,150,232,224,020,208,114
7323 :248,162,000,157,199,150,047
7329 :232,224,020,208,248,162,231
7335 :000,157,053,151,232,224,216
7341 :020,208,248,024,162,014,081
7347 :160,008,032,240,255,174,024
7353 :069,003,169,000,032,205,151
7359 :221,024,162,014,160,018,022
7365 :032,240,255,174,068,003,201
```

```
7371 :169,000,032,205,221,173,235
7377 :031,145,041,032,208,249,147
7383 :173,031,145,041,032,240,109
7389 :249,162,000,032,020,027,199
7395 :032,097,024,096,016,018,254
7401 :005,019,019,032,002,021,075
7407 :020,020,015,014,000,020,072
7413 :015,032,003,015,014,020,088
7419 :009,014,021,005,000,012,056
7425 :009,022,005,019,061,000,117
7431 :032,097,024,162,000,189,255
7437 :106,029,240,006,157,160,199
7443 :030,232,208,245,162,000,128
7449 :189,116,029,240,006,157,250
7455 :203,030,232,208,245,162,087
7461 :000,189,129,029,240,006,118
7467 :157,056,031,232,208,245,204
7473 :169,004,162,000,157,155,184
7479 :150,232,224,020,208,248,113
7485 :162,000,157,199,150,232,193
7491 :224,020,208,248,162,000,161
7497 :157,053,151,232,224,020,142
7503 :208,248,173,031,145,041,157
7509 :032,208,249,173,031,145,155
7515 :041,032,240,249,162,250,041
7521 :032,020,027,032,081,025,058
7527 :076,181,023,007,001,013,148
7533 :005,032,015,022,005,018,206
7539 :000,016,018,005,019,019,192
7545 :032,002,021,020,020,015,231
7551 :014,000,020,015,032,016,224
7557 :012,001,025,032,001,007,211
7563 :001,009,014,000,169,220,040
7569 :141,013,144,169,015,141,000
7575 :014,144,162,000,032,020,011
7581 :027,173,014,144,056,233,036
7587 :001,141,014,144,162,100,213
7593 :032,020,027,201,000,208,145
7599 :238,141,013,144,096,173,212
7605 :065,003,201,000,240,022,200
7611 :169,200,141,012,144,169,254
7617 :015,141,014,144,174,060,229
7623 :003,032,020,027,169,000,194
7629 :141,012,144,096,169,180,179
7635 :141,012,144,169,015,141,065
7641 :014,144,174,060,003,032,132
7647 :020,027,169,000,141,012,080
7653 :144,096,013,013,013,013,009
```

## Program 3: CUT-OFF! 64 Version

```
49152 :032,200,193,076,006,192,187
49158 :173,066,003,024,105,001,122
49164 :141,066,003,173,067,003,209
49170 :105,000,141,067,003,174,252
49176 :060,003,032,145,195,169,116
49182 :001,141,065,003,032,203,219
49188 :195,162,002,161,247,201,236
49194 :032,240,032,032,154,195,215
49200 :032,119,197,032,215,194,069
49206 :173,068,003,201,000,240,227
49212 :083,173,069,003,201,000,077
49218 :240,076,032,190,192,032,060
49224 :074,196,076,006,192,032,136
49230 :154,195,169,000,141,065,034
49236 :003,032,203,195,162,000,167
49242 :161,247,201,032,240,032,235
49248 :032,154,195,032,119,197,057
49254 :032,215,194,173,068,003,019
49260 :201,000,240,032,173,069,055
49266 :003,201,000,240,025,032,103
49272 :190,192,032,074,196,076,112
49278 :006,192,032,154,195,173,110
49284 :001,220,045,000,220,041,147
49290 :016,240,006,076,006,192,162
49296 :076,251,196,173,001,220,037
49302 :045,000,220,041,016,240,200
49308 :246,162,250,032,145,195,162
49314 :173,001,220,045,000,220,053
49320 :041,016,208,246,173,001,085
49326 :220,045,000,220,041,016,204
49332 :240,246,162,250,032,145,231
49338 :195,076,006,192,169,147,203
49344 :032,210,255,169,015,141,246
49350 :033,208,169,005,141,032,018
49356 :208,162,000,169,160,157,036
49362 :000,004,169,000,157,000,028
49368 :216,232,224,040,208,241,097
49374 :162,000,169,160,157,192,038
49380 :007,169,000,157,192,219,204
49386 :232,224,040,208,241,169,068
49392 :000,133,253,169,004,133,164
49398 :254,169,000,133,251,169,198
49404 :216,133,252,162,000,169,160
49410 :160,160,000,145,253,169,121
49416 :000,145,251,160,039,169,004
49422 :160,145,253,169,000,145,118
49428 :251,024,165,253,105,040,090
49434 :133,253,165,254,105,000,168
49440 :133,254,024,165,251,105,196
49446 :040,133,251,165,252,105,216
49452 :000,133,252,232,224,025,142
49458 :208,205,169,012,162,016,054
49464 :157,000,216,232,224,024,141
49470 :208,248,169,131,141,016,207
49476 :004,169,149,141,017,004,040
49482 :169,148,141,018,004,169,211
49488 :173,141,019,004,169,143,217
49494 :141,020,004,169,134,141,183
49500 :021,004,141,022,004,169,197
49506 :161,141,023,004,169,006,090
49512 :141,199,217,169,002,141,205
49518 :209,217,169,081,141,199,102
49524 :005,169,087,141,209,005,220
49530 :169,000,141,066,003,141,130
49536 :067,003,169,007,141,075,078
49542 :003,169,011,141,074,003,023
49548 :169,209,141,070,003,169,133
49554 :005,141,071,003,169,199,222
49560 :141,072,003,169,005,141,171
49566 :073,003,169,152,032,210,029
49572 :255,024,162,000,160,007,004
49578 :032,240,255,174,061,003,167
49584 :173,062,003,032,205,189,072
49590 :024,162,000,160,029,032,077
49596 :240,255,174,063,003,173,072
49602 :064,003,032,205,189,096,015
49608 :169,000,141,061,003,141,203
49614 :062,003,141,063,003,141,107
49620 :064,003,169,081,141,077,235
49626 :003,169,087,141,076,003,185
49632 :032,190,192,169,012,162,213
49638 :009,157,240,216,232,224,028
49644 :030,208,248,162,009,157,026
49650 :064,217,232,224,030,208,193
49656 :248,141,163,217,162,000,155
49662 :189,171,194,240,006,157,187
49668 :249,004,232,208,245,162,080
49674 :000,189,193,194,240,006,064
49680 :157,073,005,232,208,245,168
49686 :169,048,141,060,003,141,072
```

```
49692 :163,005,162,100,032,145,123
49698 :195,173,001,220,041,015,167
49704 :201,014,240,033,201,013,230
49710 :240,010,173,001,220,041,219
49716 :016,240,063,076,030,194,159
49722 :173,060,003,056,233,001,072
49728 :201,047,240,028,141,060,013
49734 :003,141,163,005,076,030,232
49740 :194,173,060,003,024,105,123
49746 :001,201,058,240,020,141,231
49752 :060,003,141,163,005,076,024
49758 :030,194,169,057,141,060,233
49764 :003,141,163,005,076,030,006
49770 :194,169,048,141,060,003,209
49776 :141,163,005,076,030,194,209
49782 :173,001,220,041,016,240,041
49788 :249,162,250,032,145,195,133
49794 :173,060,003,056,233,048,191
49800 :170,169,050,141,060,003,217
49806 :224,000,240,013,173,060,084
49812 :003,056,233,005,141,060,134
49818 :003,202,076,142,194,032,035
49824 :190,192,169,010,141,068,162
49830 :003,141,069,003,096,013,235
49836 :015,022,005,032,010,015,015
49842 :025,019,020,009,003,011,009
49848 :032,021,016,047,004,015,063
49854 :023,014,000,020,015,032,038
49860 :003,008,015,015,019,005,005
49866 :032,019,011,009,012,012,041
49872 :032,012,005,022,005,012,040
49878 :000,173,070,003,205,072,225
49884 :003,240,003,076,099,195,068
49890 :173,071,003,205,073,003,242
49896 :208,121,173,075,003,174,218
49902 :074,003,201,014,240,012,014
49908 :201,007,240,015,201,013,153
49914 :240,018,201,011,240,021,213
49920 :224,013,240,024,076,099,164
49926 :195,224,011,240,017,076,001
49932 :099,195,224,014,240,010,026
49938 :076,099,195,224,007,240,091
49944 :003,076,099,195,173,061,119
49950 :003,024,109,066,003,141,120
49956 :061,003,173,062,003,105,187
49962 :000,141,062,003,173,067,232
49968 :003,024,109,062,003,141,134
49974 :062,003,174,068,003,202,054
49980 :142,068,003,173,063,003,000
49986 :024,109,066,003,141,063,216
49992 :003,173,064,003,105,000,164
49998 :141,064,003,173,067,003,017
50004 :024,109,064,003,141,064,233
50010 :003,174,069,003,202,142,171
50016 :069,003,096,173,065,003,249
50022 :010,170,189,061,003,024,047
50028 :109,066,003,157,061,003,251
50034 :189,062,003,105,000,157,118
50040 :062,003,173,067,003,024,196
50046 :125,062,003,157,062,003,026
50052 :174,065,003,189,068,003,122
50058 :056,233,001,157,068,003,144
50064 :096,160,000,200,208,253,037
50070 :202,208,248,096,174,065,119
50076 :003,188,076,003,138,010,062
50082 :170,181,247,157,070,003,222
50088 :181,248,157,071,003,152,212
50094 :129,247,181,248,024,105,084
50100 :212,149,248,224,002,208,199
50106 :008,169,006,129,247,032,009
50112 :170,197,096,169,002,129,187
50118 :247,032,170,197,096,174,090
50124 :065,003,189,000,220,041,210
50130 :015,201,014,240,018,201,131
50136 :007,240,038,201,013,240,187
50142 :058,201,011,240,078,189,231
50148 :074,003,076,211,195,157,176
50154 :074,003,138,010,170,056,173
50160 :189,070,003,233,040,149,156
50166 :247,189,071,003,233,000,221
50172 :149,248,076,073,196,157,127
50178 :074,003,138,010,170,024,165
50184 :189,070,003,105,001,149,013
50190 :247,189,071,003,105,000,117
50196 :149,248,076,073,196,157,151
50202 :074,003,138,010,170,024,189
50208 :189,070,003,105,040,149,076
50214 :247,189,071,003,105,000,141
50220 :149,248,076,073,196,157,175
50226 :074,003,138,010,170,056,245
50232 :189,070,003,233,001,149,189
50238 :247,189,071,003,233,000,037
50244 :149,248,076,073,196,096,138
50250 :162,000,189,219,196,240,056
50256 :006,157,255,004,232,208,174
50262 :245,162,000,189,232,196,086
50268 :240,006,157,079,005,232,043
50274 :208,245,162,000,189,244,122
50280 :196,240,006,157,017,006,214
50286 :232,208,245,162,000,189,122
50292 :244,196,240,006,157,033,224
50298 :006,232,208,245,169,012,226
50304 :162,009,157,240,216,232,120
50310 :224,030,208,248,162,009,247
50316 :157,064,217,232,224,030,040
50322 :208,248,162,001,157,008,162
50328 :218,232,224,037,208,248,039
50334 :024,162,013,160,015,032,052
50340 :240,255,174,069,003,169,050
50346 :000,032,205,189,024,162,014
50352 :013,160,031,032,240,255,139
50358 :174,068,003.169,000,032,116
50364 :205,189,173,001,220,045,253
50370 :000,220,041,016,208,246,157
50376 :173,001,220,045,000,220,091
50382 :041,016,240,246,162,000,143
50388 :032,145,195,032,190,192,230
50394 :096,016,018,005,019,019,135
50400 :032,002,021,020,020,015,078
50406 :014,000,020,015,032,003,058
50412 :015,014,020,009,014,021,073
50418 :005,000,012,009,022,005,039
50424 :019,061,000,032,190,192,230
50430 :162,000,189,082,197,240,100
50436 :006,157,000,005,232,208,100
50442 :245,162,000,189,092,197,127
50448 :240,006,157,071,005,232,215
50454 :208,245,169,012,162,001,051
50460 :157,240,216,232,224,030,103
50466 :208,248,162,001,157,064,106
50472 :217,232,224,037,208,248,182
50478 :173,001,220,045,000,220,193
50484 :041,016,208,246,162,250,207
50490 :032,145,195,173,001,220,056
50496 :045,000,220,041,016,240,114
50502 :246,162,250,032,145,195,076
50508 :032,200,193,076,006,192,007
50514 :007,001,013,005,032,015,155
50520 :022,005,018,000,016,018,167
50526 :005,019,019,032,002,021,192
50532 :020,020,015,014,032,020,221
50538 :015,032,016,012,001,025,207
```

```
50544 :032,001,007,001,009,014,176
50550 :000,169,015,141,024,212,167
50556 :169,129,141,004,212,169,180
50562 :009,141,005,212,169,100,254
50568 :141,000,212,169,012,141,043
50574 :001,212,169,015,141,032,200
50580 :208,162,080,032,145,195,202
50586 :056,233,001,201,000,208,085
50592 :241,169,000,141,004,212,159
50598 :141,005,212,096,169,008,029
50604 :141,024,212,169,016,141,107
50610 :005,212,169,128,141,006,071
50616 :212,169,010,162,000,024,249
50622 :109,065,003,232,224,010,065
50628 :208,247,141,001,212,169,150
50634 :037,141,000,212,169,033,026
50640 :141,004,212,174,060,003,034
50646 :032,145,195,169,000,141,128
50652 :004,212,141,005,212,141,167
50658 :006,212,096,013,013,013,067
```

# Beginner's Corner

*(Article on page 84.)*

BEFORE TYPING...

Before typing in programs, please refer to "How To Type COMPUTE!'s Gazette Programs," "A Beginner's Guide To Typing In Programs," and "The Automatic Proofreader" that appear before the Program Listings.

## Program 1: Averages—VIC Version

```
4 PRINT"{CLR}{3 DOWN}{BLU}":PRINT"
  {4 SPACES}************":PRINT"
  {4 SPACES}* AVERAGES *":PRINT"
  {4 SPACES}************"          :rem 157
6 FORI=1TO8:READN$(I):NEXT         :rem 138
8 DATASUE,ANN,RITA,JUNE,BOB,JOE,JOHN,BILL
                                   :rem 181
10 DEF FNF(X)=INT(X*RND(0)+1)      :rem 111
12 PRINT"{4 DOWN}CHOOSE:":PRINT"1 INSTRUC
   TIONS":PRINT"2 PROBLEMS"         :rem 5
14 GETC$:IFC$="2"THEN50            :rem 35
16 IFC$<>"1"THEN14                 :rem 224
18 PRINT"{CLR}{DOWN}TO CALCULATE THE":PRI
   NT"AVERAGE OF SEVERAL":PRINT"NUMBERS,
   {SPACE}FIRST ADD"               :rem 65
20 PRINT"THE NUMBERS THEN":PRINT"DIVIDE T
   HE TOTAL BY":PRINT"THE NUMBER OF ITEMS
   ."                              :rem 25
22 PRINT"{3 DOWN}{GRN}PRESS RETURN."
                                   :rem 33
24 GETR$:IFR$=""THEN24             :rem 17
26 IFASC(R$)<>13THEN24              :rem 8
28 PRINT"{CLR}{BLU}HERE IS AN EXAMPLE.
   {DOWN}":PRINT"{BLK}FIND THE AVERAGE OF
   ":PRINT"THESE NUMBERS:{DOWN}" :rem 159
30 GOSUB106:PRINT"{DOWN}{BLU}ADD FOR TOTA
   L":PRINT"{RED}SUM = ";T         :rem 209
31 PRINT"{2 DOWN}{BLU}DIVIDE BY NUMBER":P
   RINT"OF ITEMS"                  :rem 177
32 PRINTT;"/";N;" = ";T/N:PRINT"{2 DOWN}
   {GRN}PRESS F1 TO CONTINUE."     :rem 202
34 GETR$:IFR$<>"{F1}"THEN34        :rem 213
36 PRINT"{CLR}{BLU}NOW TRY A PROBLEM.":PR
   INT"{BLK}{DOWN}GIVEN THESE NUMBERS":GO
   SUB106                          :rem 106
38 INPUT"{BLU}TOTAL{RED}";S:IFS=T THEN42
                                   :rem 206
40 PRINT"{DOWN}{BLU}NO, THE TOTAL IS":PRI
   NTTAB(6);T                      :rem 227
42 PRINT"{DOWN}{BLU}NOW DIVIDE.":PRINT"
   {BLK}THE AVERAGE IS":INPUTA:IFABS(A-T/
   N)<.01THEN46                    :rem 239
44 PRINT"{DOWN}{BLU}NO, THE AVERAGE IS":G
   OTO32                           :rem 130
46 PRINT"{DOWN}{BLU}CORRECT!":PRINT"
   {DOWN}{GRN}PRESS F7 TO CONTINUE"
                                   :rem 234
48 GETR$:IFR$<>"{F7}"THEN48        :rem 226
50 P=FNF(3):ONP GOTO52,60,70        :rem 60
52 PRINT"{CLR}{BLU}A BOWLING TEAM HAD":PR
   INT"THE FOLLOWING SCORES":PRINT"FOR ON
   E GAME.{DOWN}":T=0              :rem 197
54 X=INT(2*RND(0))                 :rem 251
56 FORI=1TO4:S=115+FNF(40):T=T+S:PRINTN$(
   I+X*4);TAB(8)S:NEXT             :rem 179
58 PRINT"{DOWN}WHAT WAS THE TEAM'S"
                                   :rem 240
59 PRINT"AVERAGE SCORE FOR":PRINT"THE GAM
   E?":N=4:F=10:GOTO78             :rem 219
60 PRINT"{CLR}{BLU}A BASKETBALL TEAM WON"
   :PRINT"THE FOLLOWING NUMBER":PRINT"OF
   {SPACE}GAMES.{DOWN}":T=0        :rem 199
62 N=4+FNF(3):Y=1983-N             :rem 208
64 FORI=1TON:S=50+FNF(20):T=T+S:Y=Y+1:PRI
   NTY;TAB(9)S:NEXT                :rem 138
66 PRINT"{DOWN}WHAT WAS THE AVERAGE":PRIN
   T"NUMBER OF GAMES PER":PRINT"YEAR THE
   {SPACE}TEAM WON"                :rem 62
68 PRINT"DURING THESE YEARS?":F=6:GOTO78
                                   :rem 25
70 PRINT"{CLR}{BLU}A FULLBACK GAINED THE"
   :PRINT"FOLLOWING NUMBER OF":PRINT"YARD
   S IN SEVERAL"                   :rem 73
72 PRINT"FOOTBALL GAMES.{DOWN}":T=0:N=4+F
   NF(3)                           :rem 182
74 FORI=1TON:S=60+FNF(30):T=T+S:PRINTTAB(
   5)S:NEXT                        :rem 112
76 PRINT"{DOWN}WHAT WAS THE BACK'S":PRINT
   "AVERAGE YARDS GAINED":PRINT"PER GAME?
   ":F=10                          :rem 116
78 A=INT(T/N+.5):C=FNF(4):ONC GOTO80,82,8
   4,86                            :rem 36
80 PRINT"A";A:FORI=1TO3:A=A-FNF(F):PRINTC
   HR$(66+I);A:NEXT:GOTO88         :rem 12
82 PRINT"A ";A-FNF(F):PRINT"B ";A:A=A+FNF
   (F):PRINT"C ";A:A=A+FNF(F):PRINT"D ";A
   :GOTO88                          :rem 1
84 I=A-FNF(F):J=I-FNF(F):PRINT"A ";J:PRIN
   T"B ";I:PRINT"C ";A:PRINT"D ";A+FNF(F)
   :GOTO88                         :rem 47
86 I=A-FNF(F):J=I-FNF(F):K=J-FNF(F):PRINT
   "A ";K:PRINT"B ";J:PRINT"C ";I:PRINT"D
    ";A                            :rem 102
88 GETA$:IF(A$<"A")+(A$>"D")THEN88:rem 59
90 IFASC(A$)-64=C THEN98           :rem 60
92 PRINT"{DOWN}NO, THE ANSWER IS {RED}";C
   HR$(64+C):PRINT"{GRN}PRESS F3 TO CONTI
   NUE"                            :rem 27
94 GETA$:IFA$="{F3}"THEN50         :rem 123
96 GOTO94                          :rem 21
98 PRINT"CORRECT!":PRINT"{DOWN}{GRN}PRESS
   :":PRINT" F1 ANOTHER PROBLEM":PRINT" F
   7 END PROGRAM"                  :rem 145
100 GETA$:IFA$="{F1}"THEN50        :rem 158
102 IFA$<>"{F7}"THEN100            :rem 141
```

```
104 GOTO108                              :rem 103
106 N=FNF(3)+4:T=0:FORI=1TON:J=10+FNF(10)
    :PRINTTAB(6)J:T=T+J:NEXT:RETURN
                                         :rem 18
108 PRINT"{CLR}{BLU}":END                :rem 45
```

## Program 2: Averages—64 Version

```
20 POKE 53281,1                          :rem 241
30 PRINT "{CLR}{3 DOWN}{BLU}"            :rem 25
40 PRINT "{4 SPACES}***********"         :rem 45
50 PRINT "{4 SPACES}* AVERAGES *":rem 216
60 PRINT "{4 SPACES}***********"         :rem 47
70 FOR I=1 TO 8:READ N$(I):NEXT          :rem 187
80 DATA SUE,ANN,RITA,JUNE,BOB,JOE,JOHN,BI
   LL                                    :rem 229
90 DEF FNF(X)=INT(X*RND(0)+1)            :rem 119
100 PRINT "{4 DOWN}CHOOSE:"              :rem 161
110 PRINT "1 INSTRUCTIONS"               :rem 73
120 PRINT "2 PROBLEMS"                   :rem 250
130 GET C$                               :rem 219
140 IF C$="2" THEN 500                   :rem 2
150 IF C$<>"1" THEN 130                  :rem 62
160 PRINT "{CLR}{DOWN}TO CALCULATE THE"
                                         :rem 30
170 PRINT "AVERAGE OF SEVERAL"           :rem 11
180 PRINT "NUMBERS, FIRST ADD"           :rem 3
190 PRINT "THE NUMBERS THEN"             :rem 151
200 PRINT "DIVIDE THE TOTAL BY"          :rem 24
210 PRINT "THE NUMBER OF ITEMS."         :rem 83
220 PRINT "{3 DOWN}{GRN}PRESS RETURN."
                                         :rem 81
230 GET R$:IF R$="" THEN 230             :rem 111
240 IF ASC(R$)<>13 THEN 230              :rem 101
250 PRINT "{CLR}{BLU}HERE IS AN EXAMPLE.
    {DOWN}"                              :rem 180
260 PRINT "{BLK}FIND THE AVERAGE OF"
                                         :rem 139
270 PRINT "THESE NUMBERS:{DOWN}"         :rem 74
280 GOSUB 1200                           :rem 221
290 PRINT "{DOWN}{BLU}ADD FOR TOTAL"
                                         :rem 208
300 PRINT "{RED}SUM = ";T                :rem 65
310 PRINT "{2 DOWN}{BLU}DIVIDE BY NUMBER"
                                         :rem 191
320 PRINT "OF ITEMS"                     :rem 125
330 PRINT T;"/";N;" = ";T/N              :rem 118
340 PRINT "{2 DOWN}{GRN}PRESS F1 TO CONTI
    NUE."                                :rem 226
350 GET R$:IF R$<>"{F1}" THEN 350 :rem 55
360 PRINT "{CLR}{BLU}NOW TRY A PROBLEM."
                                         :rem 143
370 PRINT"{BLK}{DOWN}GIVEN THESE NUMBERS"
                                         :rem 26
380 GOSUB 1200                           :rem 222
390 INPUT "{BLU}TOTAL{RED}";S            :rem 189
400 IF S=T THEN 420                      :rem 204
410 PRINT "{DOWN}{BLU}NO, THE TOTAL IS":P
    RINT TAB(6)T                         :rem 217
420 PRINT "{DOWN}{BLU}NOW DIVIDE."
                                         :rem 110
430 PRINT "{BLK}THE AVERAGE IS ";:rem 171
440 INPUT A                              :rem 105
450 IF A=T/N THEN 470                    :rem 65
460 PRINT "{DOWN}{BLU}NO, THE AVERAGE IS"
    :PRINT "TOTAL/NUMBER =";T/N:GOTO 340
                                         :rem 182
470 PRINT "{DOWN}{BLU}CORRECT!"          :rem 207
480 PRINT "{DOWN}{GRN}PRESS F7 TO CONTINU
    E"                                   :rem 174
490 GET R$:IF R$<>"{F7}" THEN 490 :rem 68
500 P=FNF(3)                             :rem 128
510 ON P GOTO 520,640,770                :rem 227
520 PRINT"{CLR}{BLU}A BOWLING TEAM HAD"
                                         :rem 97
530 PRINT  THE FOLLOWING SCORES"         :rem 202
540 PRINT "FOR ONE GAME.{DOWN}":T=0
                                         :rem 135
550 X=INT(2*RND(0))                      :rem 44
560 FOR I=1 TO 4                         :rem 16
570 S=115+FNF(40):T=T+S                  :rem 26
580 PRINT N$(I+X*4);TAB(8);S             :rem 64
590 NEXT I                               :rem 38
600 PRINT "{DOWN}WHAT WAS THE TEAM'S"
                                         :rem 25
610 PRINT "AVERAGE SCORE FOR"            :rem 198
620 PRINT "THE GAME?"                    :rem 163
630 N=4:F=10:GOTO 900                    :rem 130
640 PRINT "{CLR}{BLU}A BASKETBALL TEAM WO
    N"                                   :rem 78
650 PRINT "THE FOLLOWING NUMBER"         :rem 199
660 PRINT "OF GAMES.{DOWN}":T=0          :rem 169
670 N=4+FNF(3):Y=1983-N                  :rem 5
680 FOR I=1 TO N                         :rem 45
690 S=50+FNF(20):T=T+S:Y=Y+1             :rem 110
700 PRINT Y;TAB(9)S                      :rem 108
710 NEXT I                               :rem 32
720 PRINT "{DOWN}WHAT WAS THE AVERAGE"
                                         :rem 118
730 PRINT "NUMBER OF GAMES PER"          :rem 29
740 PRINT "YEAR THE TEAM WON"            :rem 153
750 PRINT "DURING THESE YEARS?"          :rem 114
760 F=6:GOTO 900                         :rem 98
770 PRINT "{CLR}{BLU}A FULLBACK GAINED TH
    E"                                   :rem 47
780 PRINT "FOLLOWING NUMBER OF"          :rem 127
790 PRINT "YARDS IN SEVERAL"             :rem 157
800 PRINT "FOOTBALL GAMES.{DOWN}":T=0
                                         :rem 99
810 N=4+FNF(3)                           :rem 225
820 FOR I=1 TO N                         :rem 41
830 S=60+FNF(30):T=T+S                   :rem 231
840 PRINT TAB(5)S                        :rem 217
850 NEXT I                               :rem 37
860 PRINT "{DOWN}WHAT WAS THE FULLBACK'S"
                                         :rem 62
870 PRINT "AVERAGE NUMBER OF YARDS"
                                         :rem 76
880 PRINT "GAINED PER GAME?"             :rem 89
890 F=10                                 :rem 133
900 A=INT(T/N+.5)                        :rem 178
910 C=FNF(4):ON C GOTO 920,950,1000,1040
                                         :rem 15
920 PRINT "A";A                          :rem 41
930 FOR I=1 TO 3:A=A+FNF(F):PRINTCHR$(65+
    I);A:NEXT                            :rem 146
940 GOTO 1060                            :rem 157
950 PRINT "A ";A-FNF(F)                  :rem 202
960 PRINT "B ";A                         :rem 46
970 A=A+FNF(F):PRINT "C ";A              :rem 197
980 A=A+FNF(F):PRINT "D ";A              :rem 199
990 GOTO 1060                            :rem 162
1000 I=A-FNF(F):J=I-FNF(F)               :rem 206
1010 PRINT"A ";J:PRINT"B ";I:PRINT"C ";A
                                         :rem 244
1020 PRINT "D{SHIFT-SPACE}";A+FNF(F)
                                         :rem 144
1030 GOTO 1060                           :rem 196
1040 I=A-FNF(F):J=I-FNF(F):K=J-FNF(F)
                                         :rem 124
1050 PRINT"A ";K:PRINT"B{SHIFT-SPACE}";J:
```

```
    PRINT"C ";I:PRINT"D ";A      :rem 109
1060 GET A$                      :rem 12
1070 IF (A$<"A")+(A$>"D") THEN 1060
                                 :rem 107
1080 IF ASC(A$)-64=C THEN 1130   :rem 240
1090 PRINT "{DOWN}NO, THE ANSWER IS {RED}
    ";CHR$(64+C)                 :rem 67
1100 PRINT "{GRN}PRESS F3 TO CONTINUE"
                                 :rem 191
1110 GET A$:IF A$="{F3}"THEN 500   :rem 1
1120 GOTO 1110                   :rem 192
1130 PRINT "CORRECT!"            :rem 201
1140 PRINT"{DOWN}{GRN}PRESS:"    :rem 141
1150 PRINT " F1 ANOTHER PROBLEM"  :rem 49
1160 PRINT " F7 END PROGRAM"       :rem 5
1170 GET A$:IF A$="{F1}" THEN 500  :rem 6
1180 IF A$<>"{F7}" THEN 1170     :rem 252
1190 GOTO 1240                   :rem 203
1200 N=FNF(3)+4:T=0                :rem 6
1210 FOR I=1 TO N                 :rem 83
1220 J=10+FNF(10):PRINT TAB(6)J:T=T+J
                                 :rem 103
1230 NEXT I:RETURN               :rem 104
1240 PRINT"{CLR}{BLU}":END        :rem 91
```

# Machine Language For Beginners

*(Article on page 90.)*

## Program 2:
### Double Decker—VIC Version

```
10 I=12288                       :rem 236
20 READ A:CK=CK+A:IF A=256 THEN 40:rem 53
30 POKE I,A:I=I+1:GOTO 20        :rem 130
40 IF CK<>27447 THEN PRINT"ERROR IN DATA
   {SPACE}STATEMENTS":STOP       :rem 198
50 END                            :rem 60
12288 DATA 160,0,169,6,153,0      :rem 97
12294 DATA 148,153,0,149,200,208  :rem 40
12300 DATA 247,160,0,169,224,153  :rem 33
12306 DATA 0,16,153,228,17,200   :rem 184
12312 DATA 192,22,208,245,169,21  :rem 39
12318 DATA 133,71,169,16,133,72  :rem 251
12324 DATA 162,24,160,0,169,224  :rem 240
12330 DATA 145,71,200,145,71,202  :rem 25
12336 DATA 240,16,24,165,71,105  :rem 242
12342 DATA 22,133,71,165,72,105  :rem 240
12348 DATA 0,133,72,76,38,48     :rem 108
12354 DATA 169,20,133,204,32,155  :rem 36
12360 DATA 224,164,98,185,149,15  :rem 56
12366 DATA 201,224,240,244,169,90 :rem 92
12372 DATA 153,149,15,198,204,208:rem 100
12378 DATA 235,169,215,133,251,169
                                 :rem 156
12384 DATA 17,133,252,32,187,48    :rem 1
12390 DATA 32,197,48,165,197,201  :rem 55
12396 DATA 0,240,10,201,1,240    :rem 124
12402 DATA 21,201,60,240,84,208  :rem 231
12408 DATA 237,198,251,160,0,177  :rem 49
12414 DATA 251,201,32,240,16,230  :rem 20
12420 DATA 251,76,102,48,160,7   :rem 193
12426 DATA 177,251,201,32,240,25  :rem 33
12432 DATA 76,102,48,230,251,160  :rem 34
12438 DATA 6,169,32,145,251,165    :rem 1
12444 DATA 251,208,2,198,252,198  :rem 54
12450 DATA 251,32,187,48,76,102  :rem 251
12456 DATA 48,160,0,169,32,145   :rem 202
12462 DATA 251,230,251,208,2,230  :rem 27
12468 DATA 252,32,187,48,76,102    :rem 5
12474 DATA 48,160,5,169,120,145  :rem 253
12480 DATA 251,136,208,251,96,160 :rem 95
12486 DATA 0,136,208,253,96,256    :rem 5
```

## Program 3:
### Double Decker—64 Version

```
10 I=49152                       :rem 236
20 READ A:CK=CK+A:IF A=256 THEN 40:rem 53
30 POKE I,A:I=I+1:GOTO 20        :rem 130
40 IF CK<>29751 THEN PRINT"ERROR IN DATA
   {SPACE}STATEMENTS":STOP       :rem 198
50 END                            :rem 60
49152 DATA 160,0,169,8,153,0      :rem 99
49158 DATA 216,153,0,217,153,0   :rem 198
49164 DATA 218,153,0,219,200,208  :rem 42
49170 DATA 241,160,0,169,224,153  :rem 42
49176 DATA 0,4,153,192,7,200      :rem 99
49182 DATA 192,40,208,245,169,39  :rem 63
49188 DATA 133,71,169,4,133,72   :rem 215
49194 DATA 162,24,160,0,169,224  :rem 255
49200 DATA 145,71,200,145,71,202  :rem 31
49206 DATA 240,16,24,165,71,105  :rem 248
49212 DATA 40,133,71,165,72,105  :rem 246
49218 DATA 0,133,72,76,44,192    :rem 159
49224 DATA 169,20,133,204,32,158  :rem 45
49230 DATA 224,164,98,185,168,3   :rem 12
49236 DATA 201,224,240,244,169,90 :rem 98
49242 DATA 153,168,3,198,204,208  :rem 56
49248 DATA 235,169,169,133,251,169
                                 :rem 170
49254 DATA 7,133,252,32,193,192    :rem 3
49260 DATA 32,203,192,165,197,201 :rem 97
49266 DATA 56,240,10,201,8,240   :rem 196
49272 DATA 21,201,35,240,84,208  :rem 248
49278 DATA 237,198,251,160,0,177  :rem 64
49284 DATA 251,201,32,240,16,230  :rem 35
49290 DATA 251,76,108,192,160,7    :rem 6
49296 DATA 177,251,201,32,240,25  :rem 48
49302 DATA 76,108,192,230,251,160 :rem 94
49308 DATA 6,169,32,145,251,165    :rem 7
49314 DATA 251,208,2,198,252,198  :rem 60
49320 DATA 251,32,193,192,76,108  :rem 52
49326 DATA 192,160,0,169,32,145    :rem 0
49332 DATA 251,230,251,208,2,230  :rem 33
49338 DATA 252,32,193,192,76,108  :rem 62
49344 DATA 192,160,5,169,120,145  :rem 51
49350 DATA 251,136,208,251,96,160:rem 101
49356 DATA 0,136,208,253,96,96,256
                                 :rem 166
```

# Poker

*(Article on page 56.)*

> BEFORE TYPING...
>
> Before typing in programs, please refer to "How To Type COMPUTE!'s Gazette Programs," "A Beginner's Guide To Typing In Programs," and "The Automatic Proofreader" that appear before the Program Listings.

## Program 1: Poker—VIC Version

```
20 POKE36879,200:PRINT"{CLR}":FORA=828TO9
   98:READB:POKEA,B:NEXT         :rem 181
```

```
30 WK=4*(PEEK(36866)AND128)+64*(PEEK(3686
   9)AND112)                          :rem 223
40 CL=37888+4*(PEEK(36866)AND128)-WK
                                      :rem 220
50 IFWK=7680THENFORA=1TO12:READB:READC:PO
   KEB,C:NEXT                         :rem 55
60 DIMJ%(13,4):DIMG$(20):S1=36875:S2=S1+1
   :VL=S1+3:D1=0:SC=0:HD=0            :rem 97
70 G$(4)="{2 SPACES}*** POKER{2 SPACES}25
   6 ***{2 SPACES}"                   :rem 242
80 G$(7)=" IT'S YOU AGAINST VIC"      :rem 217
100 G$(9)="{2 SPACES}YOU WIN AS FOLLOWS:
    {SPACE}":G$(10)="{3 SPACES}ROYAL FLUS
    H-$250{2 SPACES}"                 :rem 175
120 G$(11)=" STRAIGHT FLUSH-$100
    {2 SPACES}":G$(12)="{3 SPACES}4 OF A
    {SPACE}KIND-$20{4 SPACES}"        :rem 185
140 G$(13)="{4 SPACES}FULL HOUSE-$10
    {4 SPACES}":G$(14)="{7 SPACES}FLUSH-$
    8{7 SPACES}"                      :rem 134
160 G$(15)="{5 SPACES}STRAIGHT-$5
    {5 SPACES}":G$(16)="{4 SPACES}3 OF A
    {SPACE}KIND-$4{4 SPACES}"         :rem 184
180 G$(17)="{6 SPACES}2 PAIR-$3{7 SPACES}
    ":G$(18)="{2 SPACES}PAIR, JACKS & UP-
    $1 "                              :rem 17
200 G$(20)="{2 SPACES}EACH HAND COSTS $1.
    ":N$="{HOME}{22 DOWN}"            :rem 34
210 B$=LEFT$(N$,20):JW$=LEFT$(N$,10)
                                      :rem 177
220 A=4:MM=220:G=50:PRINT"{BLK}":POKEVL,1
    5:D1=0                            :rem 98
230 FORB=1TO22:PRINTLEFT$(N$,A)RIGHT$(G$(
    A),B):POKES1,MM                   :rem 138
240 FORC=1TOG:NEXT:POKES1,0:NEXT:FORB=1TO
    D1:NEXT:IFA=20THENPOKEVL,0:GOTO310
                                      :rem 193
250 IFA=18THENA=20:MM=220:PRINT"{WHT}":G=
    50:D1=1500                        :rem 166
260 IFA>8ANDA<18THENA=A+1             :rem 225
270 IFA=7THENA=9:PRINT"{BLU}":G=40:D1=600
                                      :rem 145
280 IFA=5THENGOSUB5000:FORA=1TO600:NEXT:A
    =7:PRINT"{WHT}":MM=238:G=40:D1=600
                                      :rem 35
290 IFA=4THENA=5:PRINT"{RED}":MM=226
                                      :rem 144
300 GOTO230                           :rem 97
310 G$(4)="":G$(5)="":G$(7)="":G$(10)="":
    G$(20)=""                         :rem 16
320 AK$="{DOWN}{GRN}{RVS} {DOWN}{LEFT}
    {DOWN}{LEFT} ":D$=B$+"{21 SPACES}"
                                      :rem 23
340 E$=LEFT$(N$,15):F$=E$+"{21 SPACES}":X
    =RND(-TI)                         :rem 104
350 HD=HD+1:GOSUB4030:POKE36879,31:PRINTC
    HR$(147):SYS828                   :rem 100
360 PRINTLEFT$(N$,5)SPC(4)"{BLU}HIT {RVS}
    K{OFF} TO KEEP"                    :rem 3
370 PRINTLEFT$(N$,7)SPC(3)"HIT {RVS}C
    {OFF} TO CHANGE":GOSUB5050        :rem 227
500 X=INT(RND(1)*13)+1:Y=INT(RND(1)*4)+1:
    IFJ%(X,Y)=1THEN500                :rem 122
510 J%(X,Y)=1:K=K+1                    :rem 10
520 E=32:IFY=1THENG=88:H=0             :rem 32
530 IFY=2THENG=83:H=2                 :rem 254
540 IFY=3THENG=65:H=0                 :rem 254
550 IFY=4THENG=90:H=2                   :rem 0
560 IFX=10THENE=49:F=48:GOTO620       :rem 114
570 IFX>1ANDX<10THENF=X+48             :rem 91
580 IFX=11THENF=10                     :rem 54
590 IFX=12THENF=17                     :rem 63
600 IFX=13THENF=11                     :rem 50
610 IFX=1THENX=14:F=1                   :rem 3
620 IFK>5THENRETURN                   :rem 244
630 IFX=1THENX=14:F=1                   :rem 5
640 IFK=1THENCD=WK+199:PT(1)=X:ST(1)=G
                                       :rem 33
650 IFK=2THENCD=WK+203:PT(2)=X:ST(2)=G
                                       :rem 23
660 IFK=3THENCD=WK+207:PT(3)=X:ST(3)=G
                                       :rem 31
670 IFK=4THENCD=WK+211:PT(4)=X:ST(4)=G
                                       :rem 30
680 IFK=5THENCD=WK+215:PT(5)=X:ST(5)=G:GO
    SUB700:POKEVL,5:Z=250:GOTO810     :rem 87
690 GOSUB700:GOTO500                  :rem 190
700 POKECD,112:POKECD+CL,0:POKECD+1,64:PO
    KECD+1+CL,0:POKECD+2,110:POKECD+2+CL,
    0                                 :rem 197
710 FORA=(CD+24)TO(CD+68)STEP22:POKEA,93:
    POKEA+CL,0:NEXT                   :rem 149
720 FORA=(CD+22)TO(CD+66)STEP22:POKEA,93:
    POKEA+CL,0:NEXT                   :rem 146
730 POKECD+88,109:POKECD+88+CL,0:POKECD+8
    9,64:POKECD+89+CL,0:POKECD+90,125
                                      :rem 94
740 POKECD+90+CL,0:LF=1:WB=230        :rem 218
750 E1=E:F1=F:G1=G:H1=H:E=160:F=160:G=160
     :H=0                             :rem 33
760 LF=LF+1:POKES1,WB:POKEVL,14       :rem 168
770 POKECD+23,E:POKECD+23+CL,H:POKECD+45,
    F:POKECD+45+CL,H:POKECD+67,G:POKECD+6
    7+CL,H                            :rem 92
780 FORB=1TO100:NEXT:POKEVL,0:POKES1,0:IF
    LF=0THENRETURN                    :rem 106
790 IFLF=4THENLF=0:E=E1:G=G1:H=H1:F=F1:GO
    TO770                             :rem 69
800 H=H+3:WB=WB+5:GOTO760             :rem 220
810 POKE198,0:PRINTD$:PRINTB$;:PRINTTAB(3
    )CHR$(28)"KEEP OR CHANGE?":CT=0
                                      :rem 104
820 PRINTE$SPC(2)CHR$(30)"?":POKES1,Z
                                      :rem 164
830 FORA=1TO100:NEXT:PRINTE$SPC(2)" ":POK
    ES1,0:FORA=1TO50:NEXT             :rem 85
840 GETH$:IFH$=""THEN820              :rem 103
850 IFH$="C"ORH$="K"THEN870             :rem 3
860 GOTO820                           :rem 113
870 IFH$="K"THEN900                    :rem 46
880 IFH$="C"THENCT=CT+1:GOSUB500:PT(1)=X:
    ST(1)=G:E(1)=E:F(1)=F:G(1)=G:H(1)=H
                                      :rem 152
890 PRINTJW$SPC(2)AK$                  :rem 12
900 PRINTE$SPC(6)"?":FORA=1TO100:NEXT:PRI
    NTE$SPC(6)" ":FORA=1TO50:NEXT     :rem 46
910 GETI$:IFI$=""THEN900              :rem 102
920 IFI$="C"ORI$="K"THEN940             :rem 1
930 GOTO900                           :rem 110
940 IFI$="K"THEN970                    :rem 52
950 CT=CT+1:GOSUB500:PT(2)=X:ST(2)=G:E(2)
    =E:F(2)=F:G(2)=G:H(2)=H           :rem 174
960 PRINTJW$SPC(6)AK$                  :rem 14
970 PRINTE$SPC(10)"?":FORA=1TO100:NEXT:PR
    INTE$SPC(10)" ":FORA=1TO50:NEXT
                                      :rem 139
980 GETJ$:IFJ$=""THEN970              :rem 118
990 IFJ$="C"ORJ$="K"THEN1020           :rem 48
1000 GOTO970                          :rem 154
1020 IFJ$="K"THEN1050                 :rem 129
1030 CT=CT+1:GOSUB500:PT(3)=X:ST(3)=G:E(3
     )=E:F(3)=F:G(3)=G:H(3)=H         :rem 218
```

```
1040 PRINTJW$SPC(10)AK$:FORA=1TO1000:NEXT
     :IFCT=3THEN1500                    :rem 209
1050 PRINTE$SPC(14)"?":FORA=1TO100:NEXT:P
     RINTE$SPC(14)" ":FORA=1TO50:NEXT
                                        :rem 185
1060 GETK$:IFK$=""THEN1050              :rem 196
1070 IFK$="K"ORK$="C"THEN1090           :rem 95
1080 GOTO1050                           :rem 200
1090 IFK$="K"THEN1120                   :rem 135
1100 CT=CT+1:GOSUB500:PT(4)=X:ST(4)=G:E(4
     )=E:F(4)=F:G(4)=G:H(4)=H           :rem 222
1110 PRINTJW$SPC(14)AK$:IFCT=3THEN1500
                                        :rem 38
1120 PRINTE$SPC(18)"?":FORA=1TO100:NEXT:P
     RINTE$SPC(18)" ":FORA=1TO50:NEXT
                                        :rem 191
1130 GETL$:IFL$=""THEN1120              :rem 194
1140 IFL$="C"ORL$="K"THEN1160           :rem 93
1150 GOTO1120                           :rem 196
1160 IFL$="K"THEN1500                   :rem 136
1170 CT=CT+1:GOSUB500:CD=WK+215:PT(5)=X:S
     T(5)=G:E(5)=E:F(5)=F:G(5)=G:H(5)=H
                                        :rem 78
1180 PRINTJW$SPC(18)AK$                 :rem 108
1500 FORTV=1TO5:IFTV>5THEN1560          :rem 126
1510 IFE(TV)>0THEN1530                  :rem 252
1520 NEXTTV:IFTV=5THEN1560              :rem 145
1530 E=E(TV):F=F(TV):G=G(TV):H=H(TV)
                                        :rem 139
1540 CD=WK+195+TV*4:IFCD>WK+215THEN1560
                                        :rem 27
1550 GOSUB700:IFTV<5THENNEXTTV          :rem 222
1560 FORA=1TO5:E(A)=0:F(A)=0:G(A)=0:H(A)=
     0:NEXTA                            :rem 242

2000 PRINTD$F$:FORA=1TO5:POKE(1015+A),PT(
     A):NEXT                            :rem 145
2010 FORA=1TO5:POKE(1015+A),PT(A):NEXT
                                        :rem 249
2020 SYS908:FORA=1TO5:PT(A)=PEEK((1015+A)
     ):NEXT                             :rem 44
2110 IFPT(5)-PT(4)=1THENIFPT(4)-PT(3)=1TH
     ENIFPT(3)-PT(2)=1THENIFPT(2)-PT(1)=1
     THENSS=1                           :rem 124
2120 IFST(1)=ST(2)THENIFST(2)=ST(3)THENIF
     ST(3)=ST(4)THENIFST(4)=ST(5)THENFL=1
                                        :rem 9
2130 SYS960:XE=PEEK(1011):ZQ=PEEK(1012)
                                        :rem 13
2160 IFPT(1)=PT(2)THENIFPT(1)=PT(3)THENIF
     PT(1)=PT(4)THENFR=1                :rem 170
2170 IFPT(5)=PT(4)THENIFPT(5)=PT(3)THENIF
     PT(5)=PT(2)THENFR=1                :rem 183
2180 IFSS=1THENIFFL=1THENIFPT(5)=14THENSC
     =SC+249:Z$=G$(10):GOTO3030         :rem 99
2190 IFSS=1THENIFFL=1THENSC=SC+99:Z$=G$(4
     ):GOTO3030                         :rem 128
2200 IFFR=1THENSC=SC+19:Z$=G$(12):GOTO303
     0                                  :rem 211
2210 IFZQ=4THENIFFR<>1THENSC=SC+9:Z$=G$(1
     3):GOTO3030                        :rem 187
2220 IFFL=1THENSC=SC+7:Z$=G$(14):GOTO3030
                                        :rem 158
2230 IFSS=1THENSC=SC+4:Z$=G$(15):GOTO3030
                                        :rem 177
2240 IFZQ=3THENSC=SC+3:Z$=G$(16):GOTO3030
                                        :rem 185
2250 IFZQ=2THENSC=SC+2:Z$=G$(17):GOTO3030
                                        :rem 185
2260 IFZQ=1ANDXE>=11THENZ$=G$(18):GOTO303
     0                                  :rem 7
2270 SC=SC-1:Z$="{6 SPACES}LOUSY HAND!
     {5 SPACES}":QP=1                   :rem 236
3030 GOSUB5050:PRINTCHR$(156):IFQP=1THENP
     RINTCHR$(144)                      :rem 68
3040 FORA=1TO5:PRINTB$;Z$:UA=20:FORB=135T
     O243STEP12                         :rem 158
3050 IFQP=1THENFORB=243TO135STEP-12:UA=32
                                        :rem 14
3060 POKEVL,15:POKES1,B:POKES2,B:FORC=1TO
     UA:NEXT:NEXT                       :rem 224
3070 POKEVL,0:POKES1,0:POKES2,0:PRINTD$:F
     ORD=1TO100:NEXT:NEXT               :rem 178
3080 FORX=1TO13:FORY=1TO4:J%(X,Y)=0:NEXT:
     NEXT:K=0                           :rem 102
3090 FORA=1TO5:PT(A)=0:ST(A)=0:NEXT:SS=0:
     FL=0:ZQ=0:FR=0:K=0:XE=0:QP=0       :rem 5
3100 FORA=1TO1500:NEXT:GOTO350          :rem 71
4030 POKE36879,120:PRINT"{CLR}"LEFT$(N$,1
     1)SPC(8)"{BLK}{RVS}HAND";HD        :rem 59
4040 D=231:POKEVL,15:FORA=1TO3:FORB=120TO
     127:POKE36879,B:POKES1,D           :rem 215
4050 POKES2,D:FORC=1TO40:NEXT:D=D+1:NEXT:
     NEXT:POKES1,0:POKES2,0:RETURN:rem 57
5000 FORA=1TO3:FORB=200TO207:POKE36879,B:
     FORC=1TO50:NEXT:NEXT:NEXT:POKE36879,
     200:RETURN                         :rem 30
5050 PRINTLEFT$(N$,3)SPC(4)CHR$(28)CHR$(1
     8)"WINNINGS:"CHR$(146)"$";SC;"
     {2 SPACES}":RETURN                 :rem 84
6000 DATA160,5,162,22,169,160,157,255,15,
     157,227,17,136,208,3,32,131,3,152,15
     7,255,147                          :rem 188
6010 DATA157,227,149,202,208,232,160,5,16
     2,220,169,160,157,22,16,157,43,16,15
     7,8,17,157                         :rem 245
6020 DATA29,17,136,208,3,32,131,3,152,157
     ,22,148,157,43,148,157,8,149,157,29,
     149,32,134                         :rem 254
6030 DATA3,208,218,96,160,7,96,138,56,233
     ,22,170,96,162                     :rem 216
6040 DATA4,142,246,3,174,246,3,160,0,140,
     247,3,185,249,3,217,248,3,176,16,72,
     185,248                            :rem 98
6050 DATA3,153,249,3,104,153,248,3,169,1,
     141,247,3,200,202,208,228,173,247,3,
     240,5,206                          :rem 171
6060 DATA246,3,208,210,96,162           :rem 198
6070 DATA 0,142,245,3,172,245,3,185,248,3
     ,217,249,3,208,4,232,141,243,3,200,1
     92,4,208                           :rem 74
6080 DATA242,238,245,3,173,245,3,201,4,20
     8,226,142,244,3,96                 :rem 138
6090 DATA836,29,839,31,849,149,852,151,86
     4,30,867,30,870,31,873,31,883,150,88
     6,150,889                          :rem 26
6100 DATA151,892,151                    :rem 10
```

## Program 2: Poker—64 Version

```
20 POKE53281,1:POKE53280,14 :PRINT"{CLR}"
   :FORA=908TO998:READB:POKEA,B:NEXT
                                        :rem 65
30 WK=1024                              :rem 9
40 CL=54272                             :rem 52
60 DIMJ%(13,4):DIMG$(20):WA=CL+4:VL=CL+24
   :D1=0:SC=0:HD=0                      :rem 184
65 FOR T=CLTOCL+24:POKET,0:NEXT         :rem 199
70 G$(4)="{10 SPACES}*** POKER{2 SPACES}2
   56 ***{2 SPACES}"                    :rem 242
80 G$(5)="{7 SPACES}IT'S YOU AGAINST THE
   {SPACE}64"                           :rem 64
```

```
100 G$(9)="{10 SPACES}YOU WIN AS FOLLOWS:
    ":G$(10)="{11 SPACES}ROYAL FLUSH-$250
    "                                  :rem 175
120 G$(11)="{9 SPACES}STRAIGHT FLUSH-$100
    ":G$(12)="{11 SPACES}4 OF A KIND-$20"
                                       :rem 185
140 G$(13)="{12 SPACES}FULL HOUSE-$10
    {2 SPACES}":G$(14)="{15 SPACES}FLUSH-
    $8"                                :rem 134
160 G$(15)="{13 SPACES}STRAIGHT-$5":G$(16
    )="{12 SPACES}3 OF A KIND-$4":rem 184
180 G$(17)="{14 SPACES}2 PAIR-$3":G$(18)=
    "{9 SPACES}PAIR, JACKS & UP-$1"
                                       :rem 17
200 G$(20)="{10 SPACES}EACH HAND COSTS $1
    .":N$="{HOME}{26 DOWN}"            :rem 102
210 B$=LEFT$(N$,20):JW$=LEFT$(N$,10)
                                       :rem 177
220 A=4:MM=60:G=10:PRINT"{BLK}":D1=0:POKE
    CL,MM:POKECL+1,MM:POKECL+4,17 :rem 86
230 FORB=1TO40:PRINTLEFT$(N$,A)RIGHT$(G$(
    A),B)                              :rem 215
234 IFLEFT$(RIGHT$(G$(A),B),2)="
    {2 SPACES}"THEN240                 :rem 202
235 POKEVL,9:POKECL+5,17:POKECL+6,129:POK
    EVL,0                              :rem 68
240 NEXT:FOR I=1TOG:NEXT:FORB=1TOD1:NEXT:
    IFA=20THENPOKECL+4,16:GOTO310  :rem 1
250 IFA=18THENA=20:MM=90:PRINT"{RED}":G=3
    0:D1=300:GOTO 300                  :rem 99
260 IFA>8ANDA<18THENA=A+1              :rem 225
270 IFA=7THENA=9:PRINT"{BLU}":G=40:D1=200
                                       :rem 141
280 IFA=5THENGOSUB5000:FORA=1TO200:NEXT:A
    =9:PRINT"{BLU}":G=40:D1=200    :rem 137
290 IFA=4THENA=5:PRINT"{RED}":MM=90
                                       :rem 95
300 GOTO230                            :rem 97
310 G$(4)="":G$(5)="":G$(7)="":G$(10)="":
    G$(20)=""                          :rem 16
320 AK$="{2 DOWN}{GRN}{2 LEFT}{RVS}
    {2 SPACES}{DOWN}{LEFT}{RVS}{2 SPACES}
    {DOWN}{LEFT}{RVS}{2 SPACES}":D$=B$+"
    {40 SPACES}"                       :rem 134
340 E$=LEFT$(N$,16):F$=E$+"{21 SPACES}":X
    =RND(-TI)                          :rem 105
350 HD=HD+1:GOSUB4030:POKE53281,1 :PRINTC
    HR$(147):GOSUB 5100                :rem 200
360 PRINTLEFT$(N$,5)SPC(13)"{BLU}HIT
    {RVS}K{OFF} TO KEEP"               :rem 51
370 PRINTLEFT$(N$,7)SPC(12)"HIT {RVS}C
    {OFF} TO CHANGE":GOSUB5050     :rem 19
500 X=INT(RND(1)*13)+1:Y=INT(RND(1)*4)+1:
    IFJ%(X,Y)=1THEN500                 :rem 122
510 J%(X,Y)=1:K=K+1                    :rem 10
520 E=32:IFY=1THENG=88:H=0             :rem 32
530 IFY=2THENG=83:H=2                  :rem 254
540 IFY=3THENG=65:H=0                  :rem 254
550 IFY=4THENG=90:H=2                    :rem 0
560 IFX=10THENE=48:F=49:GOTO620    :rem 114
570 IFX>1ANDX<10THENF=X+48             :rem 91
580 IFX=11THENF=10                     :rem 54
590 IFX=12THENF=17                     :rem 63
600 IFX=13THENF=11                     :rem 50
610 IFX=1THENX=14:F=1                    :rem 3
620 IFK>5THENRETURN                    :rem 244
630 IFX=1THENX=14:F=1                    :rem 5
640 IFK=1THENCD=WK+397:L=1:PT(1)=X:ST(1)=
    G                                  :rem 21
650 IFK=2THENCD=WK+405:L=9:PT(2)=X:ST(2)=
    G                                  :rem 23
660 IFK=3THENCD=WK+413:L=17:PT(3)=X:ST(3)
    =G                                 :rem 73
670 IFK=4THENCD=WK+421:L=25:PT(4)=X:ST(4)
    =G                                 :rem 75
675 POKE CL+1,15                       :rem 78
680 IFK=5THENCD=WK+429:L=33:PT(5)=X:ST(5)
    =G:GOSUB700:POKEVL,15:Z=250:GOTO810
                                       :rem 184
690 GOSUB700:GOTO500                   :rem 190
700 PRINT"{HOME}{10 DOWN}";            :rem 96
701 PRINT TAB(L)"{BLK}{A}****{S}"
                                       :rem 203
703 PRINTTAB(L)"-{4 SPACES}-"          :rem 153
704 PRINTTAB(L)"-{4 SPACES}-"          :rem 154
705 PRINTTAB(L)"-{4 SPACES}-"          :rem 155
706 PRINTTAB(L)"{Z}****{X}"            :rem 76
740 LF=1:WB=55                         :rem 21
750 E1=E:F1=F:G1=G:H1=H:E=160:F=160:G=160
    :H=0                               :rem 33
760 POKE VL,8:LF=LF+1:POKECL+1,WB:POKECL,
    WB:POKECL+4,17                 :rem 138
770 POKE CD+46,E:POKE CD+46+CL,H:POKECD+4
    5,F:POKECD+45+CL,H             :rem 181
775 POKE CD+86,G:POKECD+86+CL,H:POKECD+87
    ,G:POKE CD+87+CL,H             :rem 209
777 IF F<> 49 THENPOKE CD+128,F:POKECD+CL
    +128,H                             :rem 98
778 IF F<> 49 THEN POKE CD+127,E:POKE CD+
    CL+127,H:GOTO780               :rem 114
779 F=49:POKE CD+127,F:POKECD+127+CL,H:PO
    KE CD+128,48:POKE CD+128+CL,H:rem 211
780 FORB=1TO100:NEXT:POKECL+4,16:IFLF=0TH
    ENRETURN                       :rem 164
790 IFLF=4THENLF=0:E=E1:G=G1:H=H1:F=F1:GO
    TO770                              :rem 69
800 H=H+8:WB=WB+5:GOTO760          :rem 225
810 POKE198,0:PRINTD$:PRINTB$;:PRINTTAB(1
    3)CHR$(28)"KEEP OR CHANGE?":CT=0
                                       :rem 153
820 PRINTE$SPC(4)CHR$(30)"?":POKECL+1,Z:P
    OKECL,Z                        :rem 139
830 FORA=1TO100:NEXT:PRINTE$SPC(4)" ":POK
    ECL+4,16:FORA=1TO50:NEXT       :rem 248
840 GETH$:IFH$=""THEN820           :rem 103
850 IFH$="C"ORH$="K"THEN870              :rem 3
860 GOTO820                        :rem 113
870 IFH$="K"THEN900                  :rem 46
880 IFH$="C"THENCT=CT+1:GOSUB500:PT(1)=X:
    ST(1)=G:E(1)=E:F(1)=F:G(1)=G:H(1)=H
                                       :rem 152
890 PRINTJW$SPC(4)AK$                  :rem 14
900 PRINTE$SPC(12)"?":FORA=1TO100:NEXT:PR
    INTE$SPC(12)" ":FORA=1TO50:NEXT
                                       :rem 136
910 GETI$:IFI$=""THEN900           :rem 102
920 IFI$="C"ORI$="K"THEN940              :rem 1
930 GOTO900                        :rem 110
940 IFI$="K"THEN970                  :rem 52
950 CT=CT+1:GOSUB500:PT(2)=X:ST(2)=G:E(2)
    =E:F(2)=F:G(2)=G:H(2)=H        :rem 174
960 PRINTJW$SPC(12)AK$               :rem 59
970 PRINTE$SPC(20)"?":FORA=1TO100:NEXT:PR
    INTE$SPC(20)" ":FORA=1TO50:NEXT
                                       :rem 141
980 GETJ$:IFJ$=""THEN970           :rem 118
990 IFJ$="C"ORJ$="K"THEN1020         :rem 48
1000 GOTO970                       :rem 154
1020 IFJ$="K"THEN1050              :rem 129
1030 CT=CT+1:GOSUB500:PT(3)=X:ST(3)=G:E(3
     )=E:F(3)=F:G(3)=G:H(3)=H      :rem 218
```

```
1040 PRINTJW$SPC(20)AK$:FORA=1TO1000:NEXT
     :IFCT=3THEN1500                    :rem 210
1050 PRINTE$SPC(28)"?":FORA=1TO100:NEXT:P
     RINTE$SPC(28)" ":FORA=1TO50:NEXT
                                        :rem 195
1060 GETK$:IFK$=""THEN1050              :rem 196
1070 IFK$="K"ORK$="C"THEN1090            :rem 95
1080 GOTO1050                           :rem 200
1090 IFK$="K"THEN1120                   :rem 135
1100 CT=CT+1:GOSUB500:PT(4)=X:ST(4)=G:E(4
     )=E:F(4)=F:G(4)=G:H(4)=H           :rem 222
1110 PRINTJW$SPC(28)AK$:IFCT=3THEN1500
                                        :rem 43
1120 PRINTE$SPC(36)"?":FORA=1TO100:NEXT:P
     RINTE$SPC(36)" ":FORA=1TO50:NEXT
                                        :rem 191
1130 GETL$:IFL$=""THEN1120              :rem 194
1140 IFL$="C"ORL$="K"THEN1160            :rem 93
1150 GOTO1120                           :rem 196
1160 IFL$="K"THEN1500                   :rem 136
1170 CT=CT+1:GOSUB500:CD=WK+215:PT(5)=X:S
     T(5)=G:E(5)=E:F(5)=F:G(5)=G:H(5)=H
                                        :rem 78
1180 PRINTJW$SPC(36)AK$                 :rem 108
1500 FORTV=1TO5:IFTV>5THEN1560          :rem 126
1510 IFE(TV)>0THEN1530                  :rem 252
1520 NEXTTV:IFTV=5THEN1560              :rem 145
1530 E=E(TV):F=F(TV):G=G(TV):H=H(TV)
                                        :rem 139
1540 CD=WK+389+TV*8:IFCD>WK+429THEN1560
                                        :rem 43
1550 GOSUB740:IFTV<5THENNEXTTV          :rem 226
1560 FORA=1TO5:E(A)=0:F(A)=0:G(A)=0:H(A)=
     0:NEXTA                            :rem 242
2000 PRINTD$F$:FORA=1TO5:POKE(1015+A),PT(
     A):NEXT                            :rem 145
2010 FORA=1TO5:POKE(1015+A),PT(A):NEXT
                                        :rem 249
2020 SYS908:FORA=1TO5:PT(A)=PEEK((1015+A)
     ):NEXT                              :rem 44
2110 YY=0:IFPT(5)-PT(4)=1THENIFPT(4)-PT(3
     )=1THENYY=1                        :rem 147
2115 IFYY=1THENIFPT(3)-PT(2)=1THENIFPT(2)
     -PT(1)=1THENSS=1                     :rem 9
2120 IFST(1)=ST(2)THENIFST(2)=ST(3)THENIF
     ST(3)=ST(4)THENIFST(4)=ST(5)THENFL=1
                                          :rem 9
2130 SYS960:XE=PEEK(1011):ZQ=PEEK(1012)
                                         :rem 13
2160 IFPT(1)=PT(2)THENIFPT(1)=PT(3)THENIF
     PT(1)=PT(4)THENFR=1                :rem 170
2170 IFPT(5)=PT(4)THENIFPT(5)=PT(3)THENIF
     PT(5)=PT(2)THENFR=1                :rem 183
2180 IFSS=1THENIFFL=1THENIFPT(5)=14THENSC
     =SC+249:Z$=G$(10):GOTO3030         :rem 99
2190 IFSS=1THENIFFL=1THENSC=SC+99:Z$=G$(4
     ):GOTO3030                         :rem 128
2200 IFFR=1THENSC=SC+19:Z$=G$(12):GOTO303
     0                                  :rem 211
2210 IFZQ=4THENIFFR<>1THENSC=SC+9:Z$=G$(1
     3):GOTO3030                        :rem 187
2220 IFFL=1THENSC=SC+7:Z$=G$(14):GOTO3030
                                        :rem 158
2230 IFSS=1THENSC=SC+4:Z$=G$(15):GOTO3030
                                        :rem 177
2240 IFZQ=3THENSC=SC+3:Z$=G$(16):GOTO3030
                                        :rem 185
2250 IFZQ=2THENSC=SC+2:Z$=G$(17):GOTO3030
                                        :rem 185
2260 IFZQ=1ANDXE>=11THENZ$=G$(18):GOTO303
     0                                   :rem 7
2270 SC=SC-1:Z$="{14 SPACES}LOUSY HAND!!
     {4 SPACES}":QP=1                   :rem 13
3030 GOSUB5050:PRINTCHR$(156):IFQP=1THENP
     RINTCHR$(144)                      :rem 68
3040 FORA=1TO5:PRINTB$;Z$:UA=20:FORB=135T
     O243STEP12                         :rem 158
3050 IFQP=1THENFORB=243TO135STEP-12:UA=32
                                        :rem 14
3060 POKECL+4,17:POKECL+1,B:POKECL,B:FORC
     =1TOUA:NEXT:NEXT                   :rem 159
3070 POKECL+4,16:PRINTD$:FORD=1TO100:NEXT
     :NEXT                              :rem 162
3080 FORX=1TO13:FORY=1TO4:J%(X,Y)=0:NEXT:
     NEXT:K=0                           :rem 102
3090 FORA=1TO5:PT(A)=0:ST(A)=0:NEXT:SS=0:
     FL=0:ZQ=0:FR=0:K=0:XE=0:QP=0   :rem 5
3100 FORA=1TO1500:NEXT:GOTO350      :rem 71
4030 POKE53281,1:PRINT"{CLR}"LEFT$(N$,11)
     SPC(16)"{BLK}{RVS}HAND";HD   :rem 250
4040 D=231:FORA=1TO3:FORB=0TO15:POKE53280
     ,B:POKECL+1,D:POKECL+21,D    :rem 241
4050 FORC=1TO40:NEXT:NEXT:NEXT:POKE53280,
     12:RETURN                          :rem 231
5000 FORA=1TO3:FORB=0 TO 15 :POKE53280,B:
     FORC=1TO50:NEXT:NEXT:NEXT:POKE53280,
     14                                  :rem 36
5010 RETURN                             :rem 166
5050 PRINTLEFT$(N$,3)SPC(13)CHR$(28)CHR$(
     18)"WINNINGS:"CHR$(146)"$";SC;"
     {2 SPACES}"                        :rem 106
5060 RETURN                             :rem 171
5100 Z=1:FOR T=1024 TO 1063:POKET+54272,T
     -1023:POKET,160:NEXT T             :rem 32
5110 FOR T=1024 TO 2024-40 STEP 40:Z=Z+1:
     POKET+54272,Z:POKET,160            :rem 27
5120 POKET+54311,Z:POKET+39,160:NEXTT
                                        :rem 222
5130 FORT=1984 TO 2023:POKET+54272,T-1984
     :POKET,160:NEXT T                  :rem 61
5140 RETURN                             :rem 170
6040 DATA162,4,142,246,3,174,246,3,160,0,
     140,247,3,185,249,3,217,248,3,176
                                        :rem 107
6045 DATA 16,72,185,248                 :rem 121
6050 DATA3,153,249,3,104,153,248,3,169,1,
     141,247,3,200,202,208,228,173
                                        :rem 156
6055 DATA 247,3,240,5,206,246           :rem 149
6060 DATA3,208,210,96                    :rem 57
6070 DATA162,0,142,245,3,172,245,3,185,24
     8,3,217,249,3,208,4,232,141,243
                                         :rem 4
6075 DATA 3,200,192,4,208               :rem 203
6080 DATA242,238,245,3,173,245,3,201,4,20
     8,226,142,244,3,96                 :rem 138
6090 DATA836,29,839,31,849,149,852,151,86
     4,30,867,30,870,31,873,31,883,150,88
     6                                  :rem 131
6100 DATA 150,889,151,892,151           :rem 161
```

**BEFORE TYPING...**

Before typing in programs, please refer to "How To Type COMPUTE!'s Gazette Programs," "A Beginner's Guide To Typing In Programs," and "The Automatic Proofreader" that appear before the Program Listings.

# MLX For VIC And 64

*(Article on page 145.)*

BEFORE TYPING...

Before typing in programs, please refer to "How To Type COMPUTE!'s Gazette Programs," "A Beginner's Guide To Typing In Programs," and "The Automatic Proofreader" that appear before the Program Listings.

## Program 1: MLX—64 Version

```
100 PRINT"{CLR}{CYN}";CHR$(142);CHR$(8);:
    POKE53281,1:POKE53280,1          :rem 73
101 POKE 788,52:REM DISABLE RUN/STOP
                                     :rem 119
110 PRINT"{RVS}{40 SPACES}";         :rem 176
120 PRINT"{RVS}{15 SPACES}{RIGHT}{OFF}
    £*£{RVS}{RIGHT} {RIGHT}{2 SPACES}
    £*£{OFF}£*£{RVS}£{RVS}
    {13 SPACES}";                    :rem 250
130 PRINT"{RVS}{15 SPACES}{RIGHT} £G£
    {RIGHT} {2 RIGHT} {OFF}£{RVS}££*£
    {OFF}£*£{RVS}{13 SPACES}";       :rem 35
140 PRINT"{RVS}{40 SPACES}"          :rem 120
200 PRINT"{2 DOWN}{PUR}{BLK}{3 SPACES}A F
    AILSAFE MACHINE LANGUAGE EDITOR
    {5 DOWN}"                        :rem 130
210 PRINT"£5£{2 UP}STARTING ADDRESS?
    {8 SPACES}{9 LEFT}";             :rem 143
215 INPUTS:F=1-F:C$=CHR$(31+119*F:rem 125

220 IFS<256OR(S>40960ANDS<49152)ORS>53247
    THENGOSUB3000:GOTO210            :rem 235
225 PRINT:PRINT:PRINT                :rem 180
230 PRINT"£5£{2 UP}ENDING ADDRESS?
    {8 SPACES}{9 LEFT}";:INPUTE:F=1-F:C$=
    CHR$(31+119*F)                   :rem 20
240 IFE<256OR(E>40960ANDE<49152)ORE>53247
    THENGOSUB3000:GOTO230            :rem 183
250 IFE<STHENPRINTC$;"{RVS}ENDING < START
    {2 SPACES}":GOSUB1000:GOTO 230
                                     :rem 176
260 PRINT:PRINT:PRINT                :rem 179
300 PRINT"{CLR}";CHR$(14):AD=S:POKEV+21,0
                                     :rem 225
310 PRINTRIGHT$("0000"+MID$(STR$(AD),2),5
    );":";:FORJ=1TO6                 :rem 234
320 GOSUB570:IFN=-1THENJ=J+N:GOTO320
                                     :rem 228
390 IFN=-211THEN 710                  :rem 62
400 IFN=-204THEN 790                  :rem 64
410 IFN=-206THENPRINT:INPUT"{DOWN}ENTER N
    EW ADDRESS";ZZ                   :rem 44
415 IFN=-206THENIFZZ<SORZZ>ETHENPRINT"
    {RVS}OUT OF RANGE":GOSUB1000:GOTO410
                                     :rem 225
417 IFN=-206THENAD=ZZ:PRINT:GOTO310
                                     :rem 238
420 IF N<>-196 THEN 480              :rem 133
430 PRINT:INPUT"DISPLAY:FROM";F:PRINT,"TO
    ";:INPUTT                        :rem 234
440 IFF<SORF>EORT<SORT>ETHENPRINT"AT LEAS
    T";S;"{LEFT}, NOT MORE THAN";E:GOTO43
    0                                :rem 159
450 FORI=FTOTSTEP6:PRINT:PRINTRIGHT$("000
    0"+MID$(STR$(I),2),5);":";       :rem 30
```

```
451 FORK=0TO5:N=PEEK(I+K):PRINTRIGHT$("00
    "+MID$(STR$(N),2),3);",";      :rem 66
460 GETA$:IFA$>""THENPRINT:PRINT:GOTO310
                                   :rem 25
470 NEXTK:PRINTCHR$(20);:NEXTI:PRINT:PRIN
    T:GOTO310                      :rem 50
480 IFN<0 THEN PRINT:GOTO310      :rem 168
490 A(J)=N:NEXTJ                  :rem 199
500 CKSUM=AD-INT(AD/256)*256:FORI=1TO6:CK
    SUM=(CKSUM+A(I))AND255:NEXT   :rem 200
510 PRINTCHR$(18);:GOSUB570:PRINTCHR$(20)
                                   :rem 234
515 IFN=CKSUMTHEN530              :rem 255
520 PRINT:PRINT"LINE ENTERED WRONG : RE-E
    NTER":PRINT:GOSUB1000:GOTO310:rem 176
530 GOSUB2000                     :rem 218
540 FORI=1TO6:POKEAD+I-1,A(I):NEXT:POKE54
    272,0:POKE54273,0             :rem 227
550 AD=AD+6:IF AD<E THEN 310      :rem 212
560 GOTO 710                      :rem 108
570 N=0:Z=0                        :rem 88
580 PRINT"{+}";                   :rem 79
581 GETA$:IFA$=""THEN581           :rem 95
585 PRINTCHR$(20);:A=ASC(A$):IFA=13ORA=44
    ORA=32THEN670                 :rem 229
590 IFA>128THENN=-A:RETURN        :rem 137
600 IFA<>20 THEN 630               :rem 10
610 GOSUB690:IFI=1ANDT=44THENN=-1:PRINT"
    {LEFT} {LEFT}";:GOTO690       :rem 172
620 GOTO570                       :rem 109
630 IFA<48ORA>57THEN580           :rem 105
640 PRINTA$;:N=N*10+A-48          :rem 106
650 IFN>255 THEN A=20:GOSUB1000:GOTO600
                                   :rem 229
660 Z=Z+1:IFZ<3THEN580             :rem 71
670 IFZ=0THENGOSUB1000:GOTO570    :rem 114
680 PRINT",";:RETURN              :rem 240
690 S%=PEEK(209)+256*PEEK(210)+PEEK(211)
                                   :rem 149
691 FORI=1TO3:T=PEEK(S%-I)         :rem 67
695 IFT<>44ANDT<>58THENPOKES%-I,32:NEXT
                                   :rem 205
700 PRINTLEFT$("{3 LEFT}",I-1);:RETURN
                                     :rem 7
710 PRINT"{CLR}{RVS}*** SAVE ***{3 DOWN}"
                                   :rem 236
720 INPUT"{DOWN} FILENAME";F$     :rem 228
730 PRINT:PRINT"{2 DOWN}{RVS}T{OFF}APE OR
     {RVS}D{OFF}ISK: (T/D)"       :rem 228
740 GETA$:IFA$<>"T"ANDA$<>"D"THEN740
                                   :rem 36
750 DV=1-7*(A$="D"):IFDV=8THENF$="0:"+F$
                                   :rem 158
760 T$=F$:ZK=PEEK(53)+256*PEEK(54)-LEN(T$
    ):POKE782,ZK/256               :rem 3
762 POKE781,ZK-PEEK(782)*256:POKE780,LEN(
    T$):SYS65469                  :rem 109
763 POKE780,1:POKE781,DV:POKE782,1:SYS654
    66                            :rem 69
765 POKE254,S/256:POKE253,S-PEEK(254)*256
    :POKE780,253                   :rem 12
766 POKE782,E/256:POKE781,E-PEEK(782)*256
    :SYS65496                     :rem 124
770 IF(PEEK(783)AND1)OR(ST AND191)THEN780
                                   :rem 111
775 PRINT"{DOWN}DONE.":END        :rem 106
780 PRINT"{DOWN}ERROR ON SAVE.{2 SPACES}T
    RY AGAIN.":IFDV=1THEN720      :rem 171
781 OPEN15,8,15:INPUT#15,E1$,E2$:PRINTE1$
    ;E2$:CLOSE15:GOTO720          :rem 103
790 PRINT"{CLR}{RVS}*** LOAD ***{2 DOWN}"
                                   :rem 212
800 INPUT"{2 DOWN} FILENAME";F$   :rem 244
810 PRINT:PRINT"{2 DOWN}{RVS}T{OFF}APE OR
     {RVS}D{OFF}ISK: (T/D)"       :rem 227
820 GETA$:IFA$<>"T"ANDA$<>"D"THEN820
                                   :rem 34
830 DV=1-7*(A$="D"):IFDV=8THENF$="0:"+F$
                                   :rem 157
840 T$=F$:ZK=PEEK(53)+256*PEEK(54)-LEN(T$
    ):POKE782,ZK/256               :rem 2
841 POKE781,ZK-PEEK(782)*256:POKE780,LEN(
    T$):SYS65469                  :rem 107
845 POKE780,1:POKE781,DV:POKE782,1:SYS654
    66                            :rem 70
850 POKE780,0:SYS65493             :rem 11
860 IF(PEEK(783)AND1)OR(ST AND191)THEN870
                                   :rem 111
865 PRINT"{DOWN}DONE.":GOTO310    :rem 96
870 PRINT"{DOWN}ERROR ON LOAD.{2 SPACES}T
    RY AGAIN.{DOWN}":IFDV=1THEN800
                                   :rem 172
880 OPEN15,8,15:INPUT#15,E1$,E2$:PRINTE1$
    ;E2$:CLOSE15:GOTO800          :rem 102
1000 REM BUZZER                   :rem 135
1001 POKE54296,15:POKE54277,45:POKE54278,
     165                          :rem 207
1002 POKE54276,33:POKE 54273,6:POKE54272,
     5                            :rem 42
1003 FORT=1TO200:NEXT:POKE54276,32:POKE54
     273,0:POKE54272,0:RETURN     :rem 202
2000 REM BELL SOUND                :rem 78
2001 POKE54296,15:POKE54277,0:POKE54278,2
     47                           :rem 152
2002 POKE 54276,17:POKE54273,40:POKE54272
     ,0                           :rem 86
2003 FORT=1TO100:NEXT:POKE54276,16:RETURN
                                   :rem 57
3000 PRINTC$;"{RVS}NOT ZERO PAGE OR ROM":
     GOTO1000                     :rem 89
```

## Program 2: MLX—VIC Version

```
100 PRINT"{CLR}{PUR}";CHR$(142);CHR$(8);
                                   :rem 181
101 POKE 788,194:REM DISABLE RUN/STOP
                                   :rem 174
110 PRINT"{RVS}{14 SPACES}"       :rem 117
120 PRINT"{RVS} {RIGHT}?{OFF}{*}£{RVS}
    {RIGHT} {RIGHT}{2 SPACES}{*}{OFF}
    {*}£{RVS}£{RVS} "             :rem 191
130 PRINT"{RVS} {RIGHT} {G}{RIGHT}
    {2 RIGHT} {OFF}£{RVS}£{*}{OFF}
    {*}{RVS} "                    :rem 232
140 PRINT"{RVS}{14 SPACES}"       :rem 120
200 PRINT"{2 DOWN}{PUR}{BLK}A FAILSAFE MA
    CHINE":PRINT"LANGUAGE EDITOR{5 DOWN}"
                                   :rem 141
210 PRINT"{BLK}{3 UP}STARTING ADDRESS":IN
    PUTS:F=1-F:C$=CHR$(31+119*F)  :rem 97
220 IFS<256ORS>32767THENGOSUB3000:GOTO210
                                     :rem 2
225 PRINT:PRINT:PRINT:PRINT       :rem 123
230 PRINT"{BLK}{3 UP}ENDING ADDRESS":INPU
    TE:F=1-F:C$=CHR$(31+119*F)    :rem 158
240 IFE<256ORE>32767THENGOSUB3000:GOTO230
                                   :rem 234
250 IFE<STHENPRINTC$;"{RVS}ENDING < START
    {2 SPACES}":GOSUB1000:GOTO 230
                                   :rem 176
260 PRINT:PRINT:PRINT             :rem 179
300 PRINT"{CLR}";CHR$(14):AD=S     :rem 56
```

```
310 PRINTRIGHT$("0000"+MID$(STR$(AD),2),5
    );":";:FORJ=1TO6                    :rem 234
320 GOSUB570:IFN=-1THENJ=J+N:GOTO320
                                        :rem 228
390 IFN=-211THEN 710                    :rem 62
400 IFN=-204THEN 790                    :rem 64
410 IFN=-206THENPRINT:INPUT"{DOWN}ENTER N
    EW ADDRESS";ZZ                      :rem 44
415 IFN=-206THENIFZZ<SORZZ>ETHENPRINT"
    {RVS}OUT OF RANGE":GOSUB1000:GOTO410
                                        :rem 225
417 IFN=-206THENAD=ZZ:PRINT:GOTO310
                                        :rem 238
420 IF N<>-196 THEN 480                 :rem 133
430 PRINT:INPUT"DISPLAY:FROM";F:PRINT,"TO
    ";:INPUTT                           :rem 234
440 IFF<SORF>EORT<SORT>ETHENPRINT"AT LEAS
    T";S;"{LEFT}, NOT MORE THAN";E:GOTO43
    0                                   :rem 159
450 FORI=FTOTSTEP6:PRINT:PRINTRIGHT$("000
    0"+MID$(STR$(I),2),5);":";          :rem 30
455 FORK=0TO5:N=PEEK(I+K):IFK=3THENPRINTS
    PC(10);                             :rem 34
457 PRINTRIGHT$("00"+MID$(STR$(N),2),3);"
    ,";                                 :rem 157
460 GETA$:IFA$>""THENPRINT:PRINT:GOTO310
                                        :rem 25
470 NEXTK:PRINTCHR$(20);:NEXTI:PRINT:PRIN
    T:GOTO310                           :rem 50
480 IFN<0 THEN PRINT:GOTO310            :rem 168
490 A(J)=N:NEXTJ                        :rem 199
500 CKSUM=AD-INT(AD/256)*256:FORI=1TO6:CK
    SUM=(CKSUM+A(I))AND255:NEXT         :rem 200
510 PRINTCHR$(18);:GOSUB570:PRINTCHR$(20)
                                        :rem 234
515 IFN=CKSUMTHEN530                    :rem 255
520 PRINT:PRINT"LINE ENTERED WRONG":PRINT
    "RE-ENTER":PRINT:GOSUB1000:GOTO310
                                        :rem 129
530 GOSUB2000                           :rem 218
540 FORI=1TO6:POKEAD+I-1,A(I):NEXT:rem 80
550 AD=AD+6:IF AD<E THEN 310            :rem 212
560 GOTO 710                            :rem 108
570 N=0:Z=0                             :rem 88
580 PRINT"[+]";                         :rem 79
581 GETA$:IFA$=""THEN581                :rem 95
585 PRINTCHR$(20);:A=ASC(A$):IFA=13ORA=44
    ORA=32THEN670                       :rem 229
590 IFA>128THENN=-A:RETURN              :rem 137
600 IFA<>20 THEN 630                    :rem 10
610 GOSUB690:IFI=1ANDT=44THENN=-1:PRINT"
    {LEFT} {LEFT}";:GOTO690             :rem 172
620 GOTO570                             :rem 109
630 IFA<48ORA>57THEN580                 :rem 105
640 PRINTA$;:N=N*10+A-48                :rem 106
650 IFN>255 THEN A=20:GOSUB1000:GOTO600
                                        :rem 229
660 Z=Z+1:IFZ<3THEN580                  :rem 71
670 IFZ=0THENGOSUB1000:GOTO570          :rem 114
680 PRINT",";:RETURN                    :rem 240
690 S%=PEEK(209)+256*PEEK(210)+PEEK(211)
                                        :rem 149
692 FORI=1TO3:T=PEEK(S%-I)              :rem 68
695 IFT<>44ANDT<>58THENPOKES%-I,32:NEXT
                                        :rem 205
700 PRINTLEFT$("{3 LEFT}",I-1);:RETURN
                                        :rem 7
710 PRINT"{CLR}{RVS}*** SAVE ***{3 DOWN}"
                                        :rem 236
720 INPUT"{DOWN} FILENAME";F$           :rem 228
730 PRINT:PRINT"{2 DOWN}{RVS}T{OFF}APE OR
     {RVS}D{OFF}ISK: (T/D)"             :rem 228
740 GETA$:IFA$<>"T"ANDA$<>"D"THEN740
                                        :rem 36
750 DV=1-7*(A$="D"):IFDV=8THENF$="0:"+F$
                                        :rem 158
760 T$=F$:ZK=PEEK(53)+256*PEEK(54)-LEN(T$
    ):POKE782,ZK/256                    :rem 3
762 POKE781,ZK-PEEK(782)*256:POKE780,LEN(
    T$):SYS65469                        :rem 109
763 POKE780,1:POKE781,DV:POKE782,1:SYS654
    66                                  :rem 69
765 POKE254,S/256:POKE253,S-PEEK(254)*256
    :POKE780,253                        :rem 12
766 POKE782,E/256:POKE781,E-PEEK(782)*256
    :SYS65496                           :rem 124
770 IF(PEEK(783)AND1)OR(ST AND191)THEN780
                                        :rem 111
775 PRINT"{DOWN}DONE.":END              :rem 106
780 PRINT"{DOWN}ERROR ON SAVE.{2 SPACES}T
    RY AGAIN.":IFDV=1THEN720            :rem 171
781 OPEN15,8,15:INPUT#15,E1$,E2$:PRINTE1$
    ;E2$:CLOSE15:GOTO720                :rem 103
782 GOTO720                             :rem 115
790 PRINT"{CLR}{RVS}*** LOAD ***{2 DOWN}"
                                        :rem 212
800 INPUT"{2 DOWN} FILENAME";F$         :rem 244
810 PRINT:PRINT"{2 DOWN}{RVS}T{OFF}APE OR
     {RVS}D{OFF}ISK: (T/D)"             :rem 227
820 GETA$:IFA$<>"T"ANDA$<>"D"THEN820
                                        :rem 34
830 DV=1-7*(A$="D"):IFDV=8THENF$="0:"+F$
                                        :rem 157
840 T$=F$:ZK=PEEK(53)+256*PEEK(54)-LEN(T$
    ):POKE782,ZK/256                    :rem 2
841 POKE781,ZK-PEEK(782)*256:POKE780,LEN(
    T$):SYS65469                        :rem 107
845 POKE780,1:POKE781,DV:POKE782,1:SYS654
    66                                  :rem 70
850 POKE780,0:SYS65493                  :rem 11
860 IF(PEEK(783)AND1)OR(ST AND191)THEN870
                                        :rem 111
865 PRINT"{DOWN}DONE.":GOTO310          :rem 96
870 PRINT"{DOWN}ERROR ON LOAD.{2 SPACES}T
    RY AGAIN.{DOWN}":IFDV=1THEN800
                                        :rem 172
880 OPEN15,8,15:INPUT#15,E1$,E2$:PRINTE1$
    ;E2$:CLOSE15:GOTO800                :rem 102
1000 REM BUZZER                         :rem 135
1001 POKE36878,15:POKE36874,190         :rem 206
1002 FORW=1TO300:NEXTW                  :rem 117
1003 POKE36878,0:POKE36874,0:RETURN
                                        :rem 74
2000 REM BELL SOUND                     :rem 78
2001 FORW=15TO0STEP-1:POKE36878,W:POKE368
    76,240:NEXTW                        :rem 22
2002 POKE36876,0:RETURN                 :rem 119
3000 PRINTC$;"{RVS}NOT ZERO PAGE OR ROM":
    GOTO1000                            :rem 89
```

### BEFORE TYPING...

Before typing in programs, please refer to "How To Type COMPUTE!'s Gazette Programs," "A Beginner's Guide To Typing In Programs," and "The Automatic Proofreader" that appear before the Program Listings.

TREASURE $2.97 tape for VIC 20™

# RAIDERS


uses joystick, no expansions


## TAXI DRIVER

READING DEVELOPMENT

$12.97 disk for Commodore 64™

uses paddles

VIC 20 and Commodore 64 are trademarks of Commodore Electronics Ltd.

**ARK INNOVATIONS, INC.**
18133 School St.
Box 155
Amador City, CA 95601

# NEW FROM HCP! EXPANSION PRODUCTS FOR YOUR VIC - 20™

*For You Who Care Enough To Use The Best*

### 24K RAM

- Single board for all RAM expansion uses only one slot
- Low power consumption provides cool, reliable operation and extended product life
- Fully VIC-20 compatible
- Available in 8K, 16K, 24K configuration

**$199.24**

### RS-232 CARD

- True RS-232 signal levels provide maximum compatibility with peripheral devices
- Duel output connectors make installation a snap
- Fully compatible with VIC-20 hardware and software

**$49.32**

### MOTHER BOARD

- Adds 4 slots to the memory expansion port
- Includes 3K of RAM
- A socket for a 2764 EPROM Modular power supply to reduce the load on your VIC

**$69.64**

ALL PRICES ARE FOB SANTA CLARA

**HALLMARK COMPUTER PRODUCTS, INC.**
**2565 Scott Blvd., Santa Clara, CA 95050**
**Phone (408) 748-9208**

Gold-plated contact fingers for long-lasting solid connection.

ALL PRODUCTS ARE SUPPORTED BY A 2 YEAR WARRANTY!
DEALER DISTRIBUTOR INQUIRIES INVITED

VIC-20 is a trademark of Commodore Business Machines, Inc.

# COMMODORE 64 SOFTWARE

| | |
|---|---|
| **Avalon Hill** | |
| Telengard | (C)16.75 (D)20.50 |
| B-1 Bomber | (C)11.66 (D)15.30 |
| Midway Campaign | (C)11.66 |
| Nukewar | (C)11.66 |
| Tac | (D)28.00 |
| Computer Football | (C)11.66 (D)15.30 |
| Flying Ace | (C)19.00 |
| Stocks and Bonds | (C) 14.60 |
| T.C.I.F. | (C)14.60 (D) 18.25 |
| Q-Bert | 39.95 |
| Beach-Head | (D) 26.95 |
| Frantic Freddy | (D) 26.95 |
| **Broderbund** | |
| Choplifter | (CT) 27.00 |
| Sea Fox | (CT) 27.00 |
| Serpentine | (CT) 27.00 |
| David's Midnight Magic | (D) 27.00 |
| A.E. | 27.00 |
| **Commodore** | |
| Assembler 64 | (D) 16.90 |
| Logo | (D) 47.90 |
| Pilot | (D)47.90 |
| Pet Emulator | (D) 16.90 |
| Screen Editor | (D) 16.90 |
| Bonus Pack | (D,C) 16.90 |
| CP/M 2.2 Operating System | (CT) 69.00 |
| Super Expander VSP | (CT)19.50 |
| Easy Finance (1-5) | (D)19.35 |
| Easy Calc 64 | (D)73.75 |
| The Manager | (D)43.00 |
| Easy Script | (D)43.00 |
| Easy Mail 64 | (D)16.90 |
| Easy Spell 64 | (D)16.90 |
| Word/Name Machine | (D)16.90 |
| Intro to Basic 1 | (C)19.35 |
| Gortek & the Microchips | (C)24.80 |
| Easy Lesson/Easy Quiz | (D)16.90 |
| Music Machine | (CT)16.90 |
| Codewriter | (D)47.90 |

| | |
|---|---|
| Zork 1,2,3 | (D)29.50 |
| Inventory Mgmt | (D)43.00 |
| Payroll/Checkwriting | (D)43.00 |
| Accts. Payable | (D)43.00 |
| Accts. Receivable | (D)43.00 |
| General Ledger | (D)43.00 |
| Nevada Cobol | (D)47.90 |
| Simons Basic | (D)19.50 |
| Super Expander | (CT)16.90 |
| Wizard of Wor | (CT)21.50 |
| Gorf | (CT)21.50 |
| Suspended | (D)29.50 |
| Starcross | (D)29.50 |
| Deadline | 29.50 |
| Magic Desk | (CT)74.95 |
| Dragon's Den | (CT)24.95 |
| Star Ranger | (CT)24.95 |
| **Continental** | |
| The Home Accountant | 50.75 |
| FCM (Form Letter) | 76.50 |
| **Creative Software** | |
| Decision Maker | (C)10.15 (D)13.50 |
| Household Finance | (C)13.50 (D)16.90 |
| Home Inventory | (C)10.15 (D)13.50 |

| | |
|---|---|
| Loan Analyzer | (C)10.15 (D)13.50 |
| Moon Dust | (CT)23.65 |
| Astrobirtz | (CT)23.65 |
| Trashman | (CT)23.65 |
| Save New York | (CT)23.65 |
| **Datasoft** | |
| Moonshuttle | (D,C)20.30 |
| Pool Yan | (C,D)20.30 |
| Genesis | (D)27.00 |
| O Riley's Mine | (D)20.30 |
| Bruce Lee | 27.00 |
| **Epyx** | |
| Temple of Apshai | (D)27.00 |
| Upper Reaches of Apshai | (D)13.55 |
| Curse of Ra | (D)13.55 |
| Sword of Rargoal | (D)20.30 |
| Jumpman | (D,C)27.00 |
| Gateway to Apshai | (D)27.00 |
| Pitstop | (D,C)27.00 |
| Exidy Arcade Classics | 27.00 |
| Jumpman Jr. | (CT)27.00 |
| **Hes Software** | |
| Hes Mon 64 | (CT)27.00 |
| Turtle Graphics II | (CT)40.60 |

| | |
|---|---|
| Attack of the Mutant Camels | (CT)20.45 |
| Time Money Manager | (D)51.00 |
| Omni/Writer | 48.45 |
| Omni/Calc | (D)58.25 |
| Type N Writer | 24.20 |
| Synthesound 64 | (D)25.50 |
| Multiplan | 79.95 |
| Hes Writer 64 | (CT)30.45 |
| Gridrunner | (CT)20.25 |
| Benji Space Rescue | (D)30.45 |
| **Micro Prose** | |
| Floyd of the Jungle | (D)23.95 |
| Hellcat Ace | (D)23.95 |
| Solo Flight | 26.95 |
| Nato Commander | 26.95 |
| Wingman | 26.95 |
| Mig Alley Ace | 26.95 |
| **Sierra on Line** | |
| Frogger | (D,C)23.65 |
| New Jawbreaker | (D)20.30 |
| Sammy Lightfoot | (D)21.80 |
| Threshold | (D)21.80 |
| Lunar Leeper | (D)21.80 |
| B.C. Qst. for Tires | (D)25.50 |

| | |
|---|---|
| Oil's Well | (D)21.80 |
| Learning W/Leeper | (D)21.80 |
| **Sirus Software** | |
| Turmoil | (D)27.00 |
| Snake Byte | (D)27.00 |
| Squish 'Em | (D)23.65 |
| Repton | (D)27.00 |
| Blade/Blackpoole | (D)27.00 |
| Type Attack | (D)27.00 |
| **Spinnaker** | |
| Face Maker | (D)23.65 |
| Hey Diddle Diddle | (D)20.30 |
| Kindercomp | (D)20.30 |
| Snooper Troops 1 | (D)27.00 |
| Fraction Fever | (D)23.65 |
| Amazing Thin | (D)27.00 |
| Alphabet Zoo | 23.65 |
| Kids on Keys | 23.65 |
| Delta Drawing | 27.25 |
| Up for Grabs | 27.25 |
| Cosmic Life | 23.65 |
| Trains | 28.00 |
| **Synapse** | |
| Ft. Apocalypse | (D,C)23.70 |
| Shamus | (D,C)23.70 |
| Blue Max | (D,C)23.70 |
| Dimension X | (D,C)23.70 |
| **Strategic Simulations** | |
| Combat Leader | (D)29.00 |
| Knights of the Desert | (D)29.00 |
| Computer Baseball | (D)29.00 |
| Eagles | (D)29.00 |
| Ringside Seat | (D)29.00 |
| **Timeworks** | |
| Dungeons of Algebra Dragons | (D,C)19.10 |
| Robbers of the Lost Tomb | (D,C)19.10 |
| Wall Street | (D,C)19.10 |
| Money Manager | (D,C)19.10 |
| Electronic Checkbook | (D,C)19.10 |
| **Word Pro 3 +** | **65.00** |

**commodore**

| | |
|---|---|
| Commodore 64 | 219.96 |
| 1541 Disk Drive | 239.95 |
| 1530 datasette | 61.30 |
| MPS 801 Printer | 214.75 |
| 1650 Automodem | 89.95 |
| 1702 Color Monitor | 239.95 |
| 1600 Modem | 55.00 |
| RS 232 Interface | 57.80 |
| 3 Port Expander | 30.00 |
| 6 Port Expander | 75.00 |
| 64 Prog. Ref. Guide | 17.00 |

**Diskettes**

| | |
|---|---|
| BASF 5¼" SSDD2Pak | 5.50 |
| BASF Soft Box (10) | 23.80 |

**Joysticks**

| | |
|---|---|
| Pointmaster | 10.95 |
| Pointmaster Pro | 18.55 |
| Fire Control | 7.85 |
| Wico Command Ctrl. | 19.50 |
| Wico "Boss" | 13.50 |
| Wico Red Ball | 20.75 |

**COLECO**

| | |
|---|---|
| Adam Computer System | 595.00 |
| Adam Computer Module | 445.00 |

**Koala**

Koala Touchtablet (D)79.95

Terms and Ordering Information:
To order call 1-800-527-8698 and send certified checks, money orders or personal checks (allow 2 weeks to clear), or use your Visa, Master Card or American Express. Inside Texas call 1-800-442-8717. Include $2 for postage and handling (C.O.D. orders add $1.65) UPS Blue Label $3, Canada $6 call for shipping charges on Hardware. Other countries include 10% for P & H. All products factory sealed with manufacture's warranty. All returns require R.A. #. Prices subject to change without notice. Order desk hours: Mon-Fri 8-5, Sat 9-1 CST.

**ATARI, APPLE, VIC 20, T.I., SOFTWARE LISTS AVAILABLE**
**For further information and inquires call 214-753-0485**
**FOR ORDERS ONLY CALL TOLL FREE**
Outside Texas 1-800-527-8698
Inside Texas 1-800-442-8717

**SoftWare Warehouse outlet**

**P.O. Box 2511 Longview, Texas 75606**

# PRODUCT MART

## Video-RF remodulator for Commodore VIC-20 and 64

At last your TV set performs like a video monitor. Simply plugs into your video output.

**$69.95** Retail

Bear Technologies INC.
4321 Airwest S.E.
Grand Rapids, MI 49508
(616) 698-5000
VISA and Mastercard accepted
Dealer inquiries invited

## Numeric key pad for Commodore VIC-20 and 64

With full cursor control and special function keys. No software interaction.

**$59.95** Retail

Bear Technologies INC.
4321 Airwest S.E.
Grand Rapids, MI 49508
(616) 698-5000
VISA and Mastercard accepted

## DISK DUPLICATOR FOR COMMODORE SINGLE DISK DRIVES

(1540, 1541 and 2031 used with a VIC-20, commodore 64, CBM 4000 or CBM 8000 computer)

DISK DUPLICATOR provides you a fast and easy way to make back-up copies of your precious, irreplaceable diskettes. Enjoy the convenience of a dual disk drive without the expense. DISK DUPLICATOR is 100% MACHINE LANGUAGE, 100% FAST, and most importantly, 100% AFFORDABLE!

Don't let an accident or mistake catch you without back-up copies of all your diskettes. ORDER "DISK DUPLICATOR" TODAY at the special introductory price of only $14.95 postage paid (check or money order only please).

H&H ENTERPRISES DEPT. 123G
5056 NORTH 41st STREET
MILWAUKEE, WISCONSIN 53209

## CONVERSE WITH YOUR COMPUTER

**AT LAST! A FULL IMPLEMENTATION** of the original ELIZA program is now available to run on your Commodore 64!

Created at MIT in 1966, ELIZA has become the world's most celebrated artificial intelligence demonstration program. ELIZA is a non-directive psychotherapist who analyzes each statement as you type it in and then responds with her own comment or question – and her remarks are often amazingly appropriate!

Designed to run on a large mainframe, ELIZA has never before been available to personal computer users except in greatly stripped down versions lacking the sophistication which made the original program so fascinating.

Now, our new Commodore 64 version possessing the **FULL** power and range of expression of the original is being offered at the introductory price of only $25. And if you want to find out how she does it (or teach her to do more) we will include the complete **SOURCE PROGRAM** for only $20 additional.

Order your copy of ELIZA today and you'll never again wonder how to respond when you hear someone say, "Okay, let's see what this computer of yours can actually do!"

**ELIZA IS AVAILABLE IN THE FOLLOWING FORMATS:**
(Please specify Disk or Cassette)

1. Protected Version ..... $25
(Protected Version can be run but not listed or modified)
2. Un-protected Commodore 64 BASIC Source Version ..... $45
(Source Version can be listed and modified as well as run)
Both versions include a six page user manual.

Please add $2.00 shipping and handling to all orders
(California residents please add 6% sales tax)

**ARTIFICIAL INTELLIGENCE RESEARCH GROUP**
921 North La Jolla Avenue, Dept. G
Los Angeles, CA 90046
(213) 656-7368 (213) 654-2214
MC, VISA and checks accepted

## EXPAND YOUR VIC 20™ MEMORY!

Affordable Memory Expander lets you add 2K RAM Circuits as your needs increase. (Up to 35K.)

| | |
|---|---|
| Mother Board with Instructions | $39.95 |
| Mother Board & Sockets with Instructions | $54.95 |
| Complete Kit with Cabinet (35K) | $159.95 |
| Assembled & Tested Expander (35K) | $189.95 |
| Each Add'l 2K Chip (Up to 280K paged RAM) [1 Shipping] | $8.95 |

Send Check or Money Order to

**PERIPHERAL DEVELOPMENT**
P. O. Box 28247
St. Louis, MO 63132
(Add $5 S/H Missouri residence add 5¼% tax)
Prices subject to change without notice

Vic 20 is a trademark of Commodore Electronics, Ltd.

## HOMEWORK HELPER 3-6 GRADE ARITHMETIC

Child enters problems from his text book and is guided through the entire process. Every step is displayed on the screen including carry digits. Help feature and color screens make math fun. Addition, subtraction, multiplication and division.

VIC 20 or C64 TAPE

***Send check or money order for $11.00.***

**DEB HOMEWARES**
4044 Westlake Dr.
Cortland, Ohio 44410

## LAIR of the LIZARDMEN

In an abandoned corner of desert lie ancient, unsolved mysteries. Danger, death and forgotten treasure await you in

LAIR OF THE LIZARDMEN
text adventure for C-64 or VIC-20
(VIC requires 8k expansion)
Specify disk or tape, VIC or 64

$15.00 disk or tape. Add $1 shipping. California residents add 6½% tax.

WEREWOLF SOFTWARE
109 Minna Street
Suite 353
San Francisco, CA 94105

## VIC & 64


AT LAST! The information you need, without always going back to the manual. These durable plastic coated overlays contain program starting locations, function key labeling, commands and additional aids in center cutout.

Please send me the following Leroy's Cheatsheet™ keyboard overlays

| 20 | 64 | | 20 | 64 | |
|---|---|---|---|---|---|
| ☐ | ☐ | Programmer's Aid[1] | ☐ | ☐ | Graphic printer (1515 & 1525)[1] |
| ☐ | ☐ | Vicmon[1] | ☐ | ☐ | Paper Clip |
| ☐ | ☐ | Super Expander[1] | ☐ | ☐ | Script 64 |
| ☐ | ☐ | Hesmon | ☐ | ☐ | HES Writer[2] |
| ☐ | ☐ | Calc Result (advanced) | ☐ | ☐ | Wordpro 3 plus |
| ☐ | ☐ | Term 64[1] | ☐ | ☐ | Easy Script[1] |
| ☐ | ☐ | Quick Brown Fox | ☐ | ☐ | Basic |

CG384

Send check or money order plus $1.00 (postage and handling)
PA residents add 6% sales tax.

Name _____
Address _____
City _____ State _____ Zip _____

(1) Products of Commodore Business Machines, Inc. (2) Product of United Microware Industries, Inc.
(3) Products of Human Engineered Software. VIC-20 is a Trademark of Commodore Business Machines, Inc.

CHEATSHEET PRODUCTS™
P.O. Box 8299 Pittsburgh PA. 15218 (412) 456-7420

VISA **OPEN SALE!** MasterCard

finally it's here: a genuine discount wholesale computer supply company. Quality guaranteed sales & service, for the lowest prices anywhere.

# DATA-BYTE

**314-423-3469**

p.o. box 8467, st. louis, missouri 63132

Special Script 64 Dictionary Disk $15.95

WE SELL EVERYTHING FOR LESS

FREE: Programs on Tape or Disk with orders of $75.00 or more.

---

## Want A Greener Green Thumb? Shake Hands with PLANTIN' PAL

**If you enjoy Gardening, you'll love PLANTIN' PAL.**

With PLANTIN' PAL you can:

- Take your custom-designed layout to your garden
- Plant the right amount of each vegetable for your needs
- Identify and beat bugs & diseases
- Create your individual planting & fertilizing schedule
- And so much more

**Get friendly with Plantin' Pal today.**
On disk for Commodore 64.
Printer optional but preferred.

VISA **$29.95** MasterCard

To order call (612)925-2591 or write:
**Home & Hobby Software**
4936 Morgan Ave. S., Minneapolis, MN 55409

---

## BASIC-PLUS

FOR COMMODORE - 64

****14 POWERFUL SYSTEM COMMANDS****
***Make Basic Programming Faster & Easier***

**Line Numbering:**
*AUTO (Incr)
*RENUM (Old, New, Incr)

**Formatting:**
*CRUNCH
*UNCRUNCH

**Program Structure:**
*APPEND 'Program', d
*DELETE m-n
*OLD

**Search/List:**
*ANALYSE
*FIND/String/
*REF m-n

**Function-Key Definition/Control:**
*FSET Fmn='Commands' FSAVE'Table', d
*FLIST m-n FLOAD 'Table', d

**5k Program, Written in Machine Language:
*Loads in 15 Seconds (From Diskette)
*Can Co-Exist With DOS Wedge
**Ultra-Fast (Completely Renumbers 500-Line Program in 5 Seconds!!!)
**FIND & REF Commands Practicly Eliminate Manual Program Scanning For Usage Info
**64 Frequently Used Command Sequences Can Be Defined & Accessed Via **TWO-TOUCH** Function-Key Selection (A Useful Function Table Is Supplied)
**CRUNCH Can Improve Pgm Load-Time And Space Requirements As Much As 15%!!!
**Package Also Includes A Utility Pgm To Copy Machine Language Pgm Files

**$21.95** — Send Check Or Money Order To:
D. Burnett, 4817 Clipping Ct., Louisville, Ky., 40222
(502-228-0341). Please Allow 2-3 Weeks For Delivery.

---

## VIC 20 COMMODORE 64

**UMI / VIC 20**

| | |
|---|---|
| Spiders of Mars [C] | $29.95 |
| Meteor Run [C] | $29.95 |
| Amok [T] | $16.95 |
| Sat & Met [C] | $29.95 |

**UMI / New for C-64**

| | |
|---|---|
| Pennant Drive<br>2 player baseball strategy | $29.95 |
| Motor Mania<br>hi-performance racing game | $29.95 |
| Fuego<br>fight fire-throwing drones to save the space crew | $29.95 |

**TOTL SOFTWARE**

| | |
|---|---|
| TOTL Text 2.6 [D] | $34.00 |
| TOTL Label 2.6 [D] | $20.00 |

Send cash, check or money order to:
ARIES MARKETING CO.
P.O. Box 4196
4200 Shannon Drive
Baltimore, Maryland 21205
Md. residents add 5% state sales tax

---

## COMMODORE ATARI/T.I.

**PROTECTIVE COVERS $5.99**

Atari CX 2600
Commodore Vic 20/64
Texas Instruments 99/4
also
Atari Backpack $7.99

**C-64 SOFTWARE (cartridges)**

| | |
|---|---|
| Robbers of the Lost Tomb (Timeworks) | $18.95 |
| Moondust (Creative) | $23.95 |
| Save New York (Creative) | $23.95 |
| C-64 Audio/Video Cable (Audio input/output & video output) | $ 7.95 |

**SLM COMPUTER PRODUCTS**
**1472 Lou Dillon Ln. #2**
**Santa Barbara, Ca. 93103**

Add $1.50 Shipping/Handling—Ca. res 6% tax

---

VIC 20™

**KIDBIT SOFTWARE**
PRESENTS:

## THE PLAYSCHOOL TAPES

*Pre-school learning programs, on cassettes for the unexpanded VIC*

- ☐ WORMSICAL COUNT
- ☐ SAME/NOT SAME GAME
- ☐ SMALL WIZARD/CAPITAL WIZARD
- ☐ ALPHA-BEE SEQUENCE

☐ **$7.95 ea.** ☐ **$12.95 for 2**
☐ **$24.95 all 4** ☐ **FREE BROCHURE**

CA residents add 6½% sales tax.
Postage & Handling Add $1.00
Personal check or money order accepted.

**Kidbit Software**
**7001 Sunkist Drive**
**Oakland, CA 94605**
**(415) 638-1243**

---

## EPROM PROGRAMMER

**FOR**
**PET · COMMODORE-64 · VIC-20**

**DELUXE-PROGRAMS**

**2716 thru 27128**
**2516 thru 2564**
**MENU driven DISK sftwr $99.50***
**incl. MACH. LANG. MONITOR and FILE MANAGEMENT. COMMANDS include: read, verify, block select, program, load, save, etc +MONITOR COMMANDS**
**L.I.F. SOCKETS for easy operation. device connects to CPU's USER PORT. software to operate ALL CPU's incl.**

**ECONOMY-2716-64**
**read, pgm, & ver. ONLY**
**basic programmer-when editing & file storage ARE NOT needed-LIF incl. $59.50***

(215)256-6933 **DATCO** (215)256-6933

VISA **Box 267 Lederach, Pa. 19450** MasterCard
***+$2. ship. & Pa. res. add 6%**

---

SAN FRANCISCO AREA
COMMODORE USERS GROUP

# SPHINX

PET-VIC 20-C64

MEETINGS CLASSES

NEWSLETTER

BULLETIN BOARD SYSTEM

PUBLIC DOMAIN SOFTWARE

SEND $2 for CATALOG

JOIN ● $24

ANNUAL MEMBERSHIP FEE

(415) 527-9286

267 ARLINGTON AVENUE
KENSINGTON, CA 94707

---

## 'PUBLIC DOMAIN'™ — SOFTWARE —

**Supporting all COMMODORE computers**
**Written by users, for users**
**★ GAMES ★ UTILITIES ★ EDUCATIONAL ★**

**VIC 20™**
collection #1 – collection #2 – collection #3
collection #4 – collection #5 – collection #6
70+ programs per collection – Tape/Disk – $10.00

**COMMODORE 64™**
64 collection #1 – 64 collection #2 – 64 collection #3
64 collection #4 – 64 collection #5
25+ programs per collection – Tape/Disk – $10.00

**PET® / CBM®**
5 Utility – Tapes/Disks – $10.00 each
11 Game – Tapes/Disks – $10.00 each
6 Educational – Tapes/Disks – $10.00 each

**DINSET™: Reset Switch**
Works on Vic 20 or Commodore 64 – $5.00

**All prices include shipping and handling.**
**CHECK, MONEY ORDERS,**
**VISA and MASTERCARD accepted.**

*For A Free Catalog Write:*

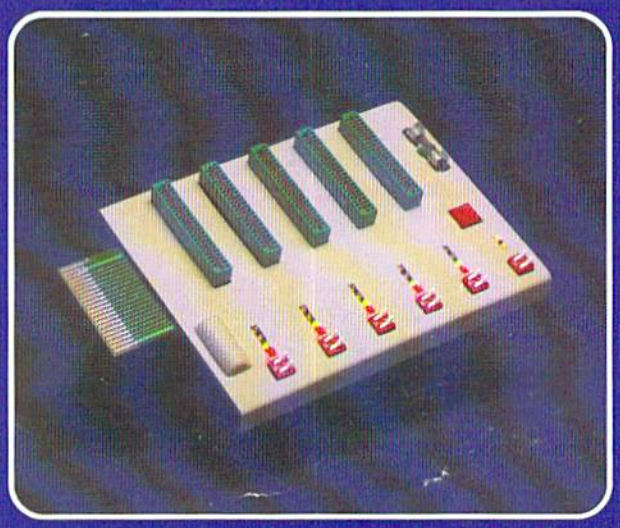
**Public Domain, Inc.**
5025 S. Rangeline Rd., W. Milton, OH 45383
10:00 a.m. - 5:00 p.m. EST – Mon. thru Fri.
(513) 698-5638 or (513) 339-1725

# ADVERTISERS INDEX

# Five Slot Expansion Interface for the C-64™

The CARDBOARD/5 (CB/5) is an enclosed five slot, fully switch selectable, expansion interface for the Commodore 64™. This quality product allows the user to switch select any cartridge slot or combination of cartridge slots. Twenty-two color coded light emitting diodes give status indication. Each slot has four LEDs and two toggle switches for indication and control. Two master toggle switches allow the user to manually override any situation.

All Cardco products are individually tested to insure quality and reliability.



**Some of the features of the CARDBOARD/5 are:**

- high quality glass/epoxy circuit board
- gold plated contacts
- logic lines are switched by solid state IC switches
- full LED status indication
- convenient toggle switches
- full support under the board to prevent flexing
- full plastic enclosure to insure safety
- fused to protect your computer
- convenient reset button
- CARDCO, Inc.'s exclusive Lifetime Guarantee

Manufacturer's Suggested Price: $79.95


See a complete line of American made Cardco Products at a computer store near you, today.

313 Mathewson • Wichita, Kansas 67214 • (316) 267-6525

Commodore ™ is a registered trademark of Commodore Business Systems, Inc.

Cardco, Inc.

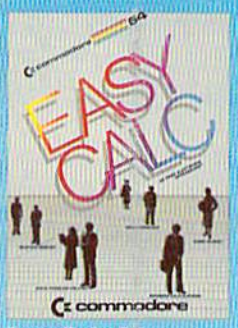
commodore 64

# INDISPENSABLE SOFTWARE

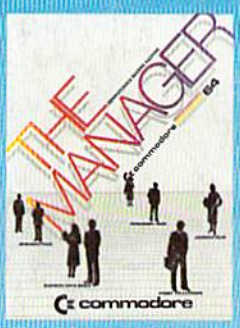
## For Your Most Important Computing Needs


**EasyScript 64**
Displays 764 lines × 240 characters. Prints to 130 columns. Works with EasySpell 64.

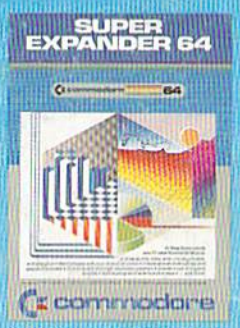

**EasySpell 64**
20,000 word Master Dictionary and automatic spelling checker. Works with EasyScript 64.




**EasyCalc 64**
Multiple electronic spreadsheet with color bar graph feature. 63 columns × 254 rows.




**The Manager**
Sophisticated database system with 4 built-in applications, or design your own. Text, formulas, graphics.




**SuperExpander 64**
21 special commands. Combine text with high resolution graphics. Music and game sounds.


**Easy Finance I—Loan Analysis**
12 loan functions. Bar graph forecasting as well as calculation.


**Easy Finance II—Basic Investment Analysis**
16 stock investment functions. Investment bar graph.


**Easy Finance III—Advanced Investment Analysis**
16 capital investment functions. Bar graphs.


**Easy Finance IV—Business Management**
21 business management features. Bar graphs.


**Easy Finance V—Statistics and Forecasting**
Assess present/future sales trends with 9 statistics and forecasting functions.


**Accounts Payable/ Checkwriting**
11 functions. Automatic billing. 50 vendors/disk.


**Accounts Receivable/Billing**
11 billing functions. Printed statements.


**General Ledger**
8 general ledger options. Custom income statement, trial balances, reports.


**Inventory Management**
1000 inventory items. Full reports.


**Payroll**
24 different payroll functions. Integrated with G/L system.

commodore COMPUTERS

**First In Quality Software**

www.commodore.ca