

PROOF OF PURCHASE

Software Giveaway with Purchase of this Magazine

See back cover

HOME COMPUTERTM magazine

FOCUSING EXCLUSIVELY ON ● APPLE ● COMMODORE ● IBM ● TEXAS INSTRUMENTS

Vol. 4 No. 5

\$3.50 in USA
\$4.50 in Canada

A Ready-To-Use

Quiz Construction Set for:

- Vocabulary Building
- Trivia Gaming
- Foreign Language Training

Arcade Gaming Action with
Slithering Snakes and
Birds of Prey

Your Personal
Loan Calculator

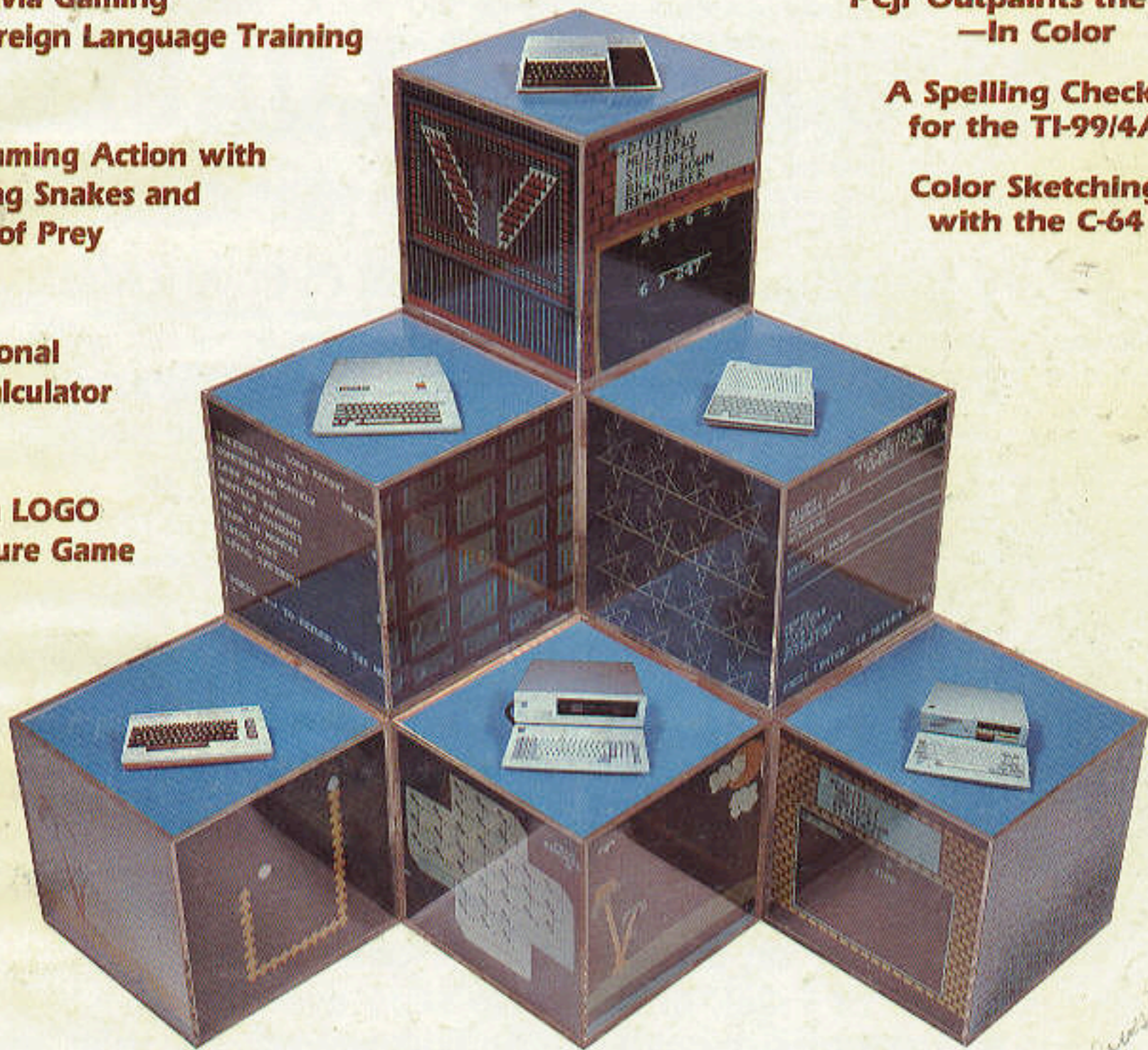
Building a LOGO
Adventure Game

Secrets of Programming
the Apple IIc

PCjr Outpays the Mac
—In Color

A Spelling Checker
for the TI-99/4A

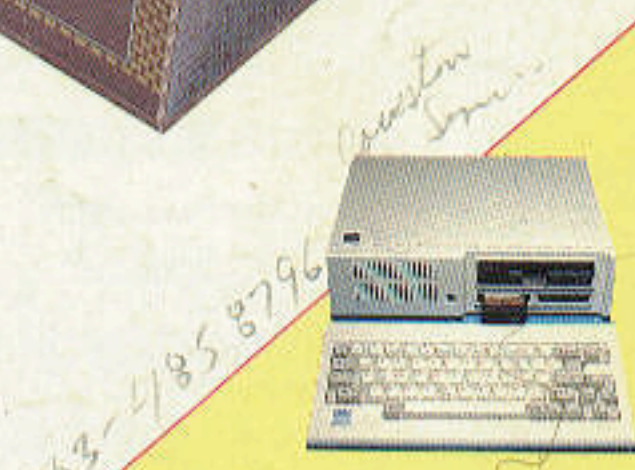
Color Sketching
with the C-64



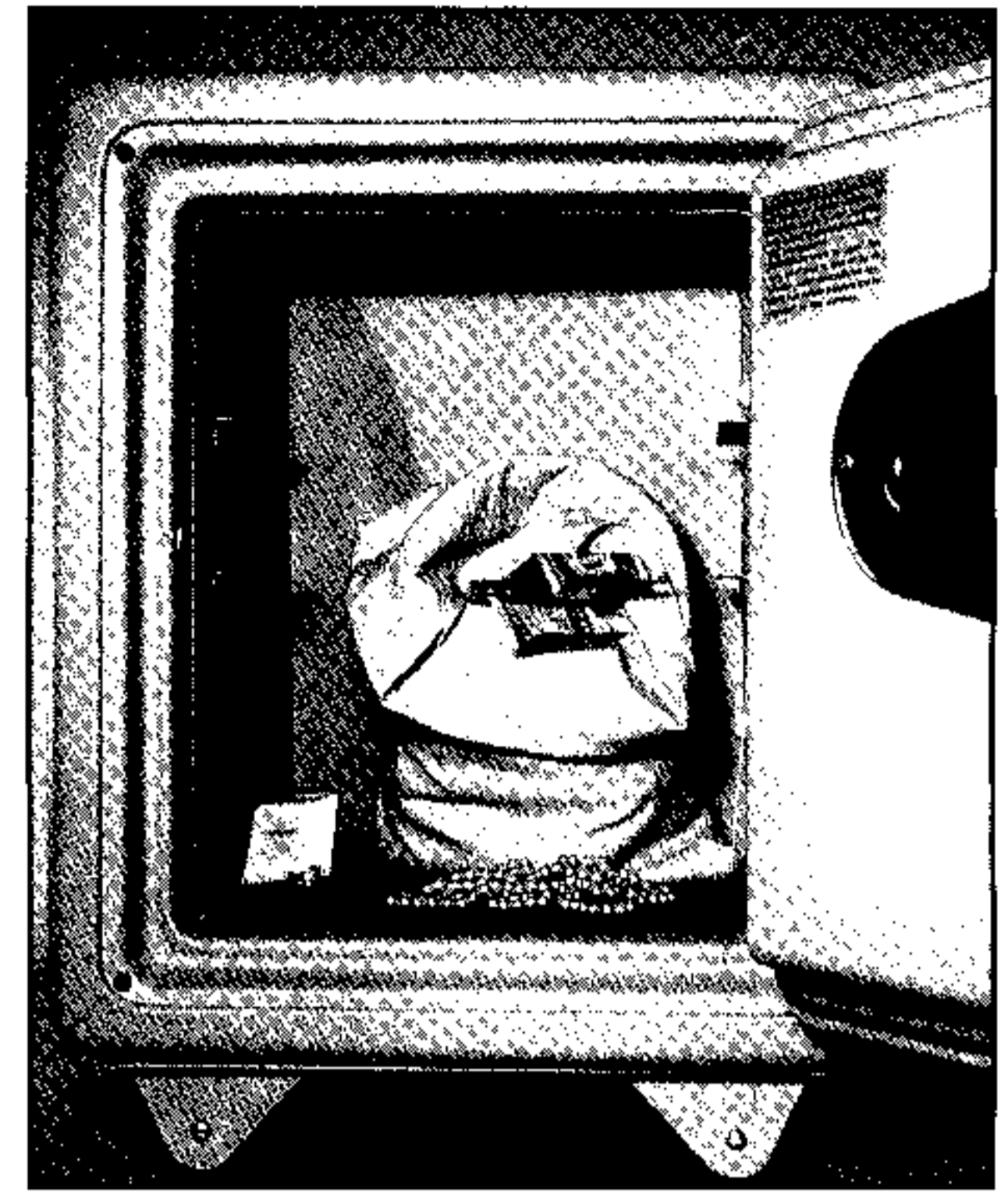
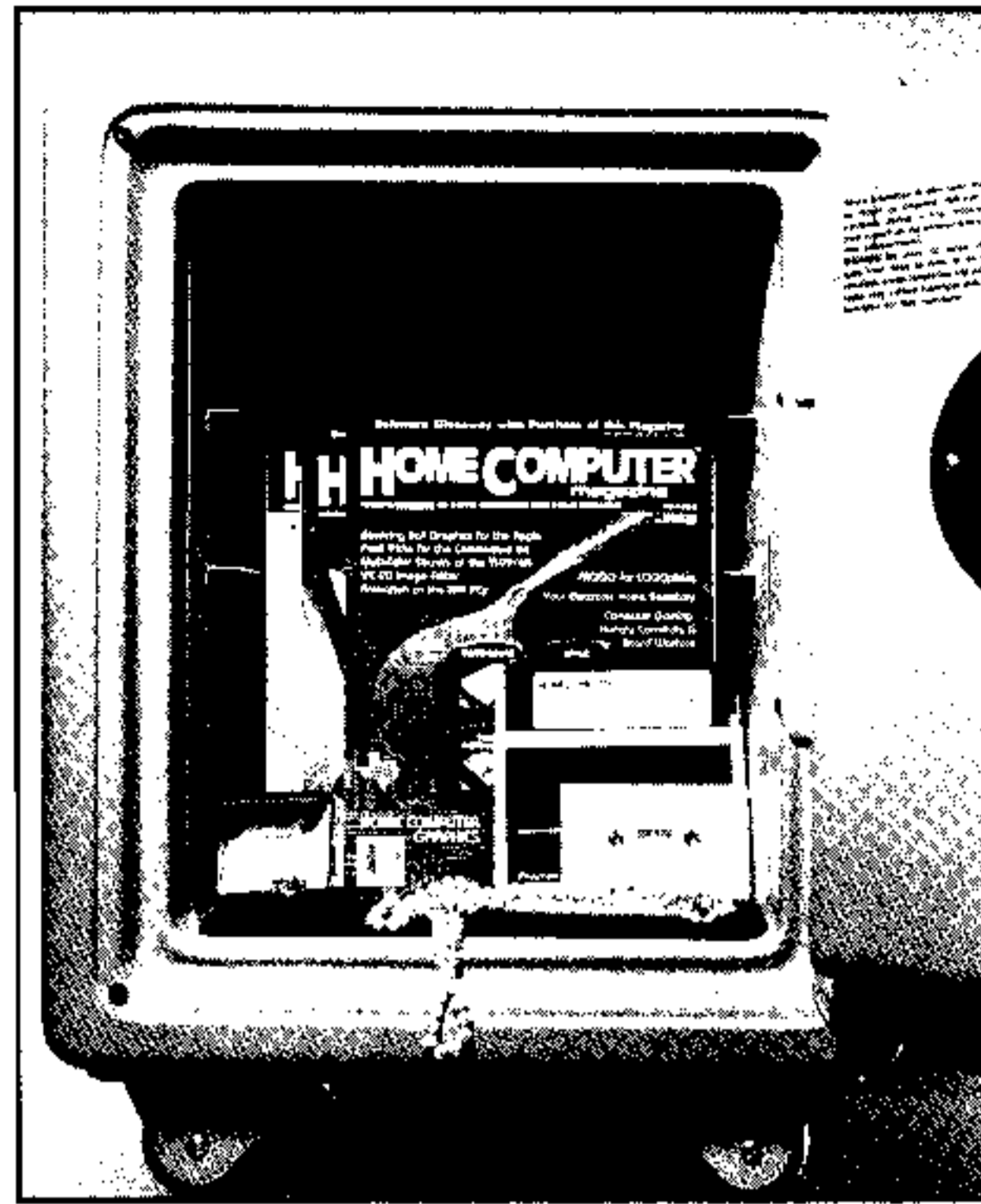
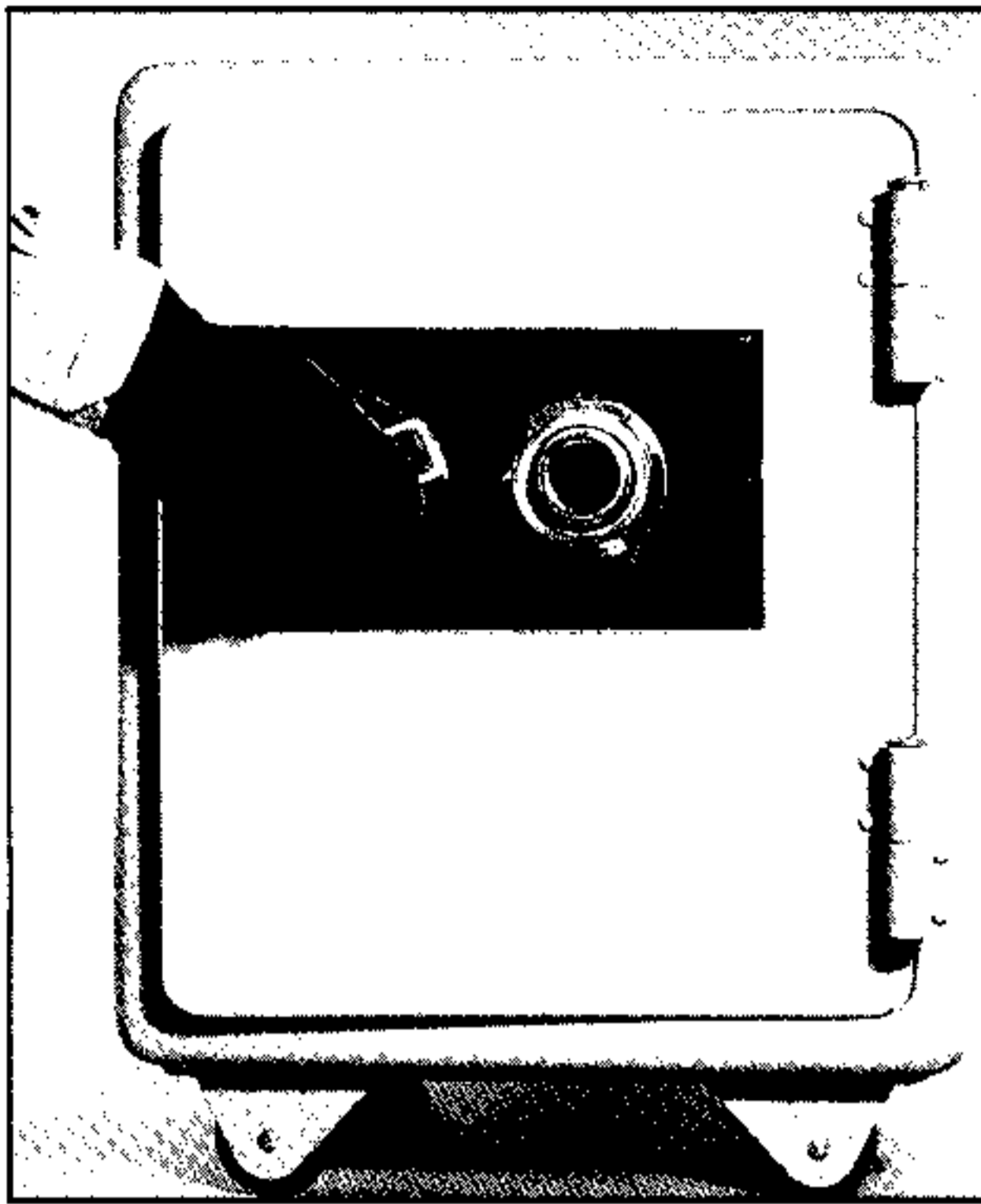
45806COUJRT 4 AUG85 H11135070
 JAMES COURTNEY
 RT 4 2744 W HUME RD
 CRIDERSVILLE OH 45806



**Building Up
Your Software Library**



Welcome New PCjr Owners:
We Do Junior Right!



MISSING ANY VALUABLES?

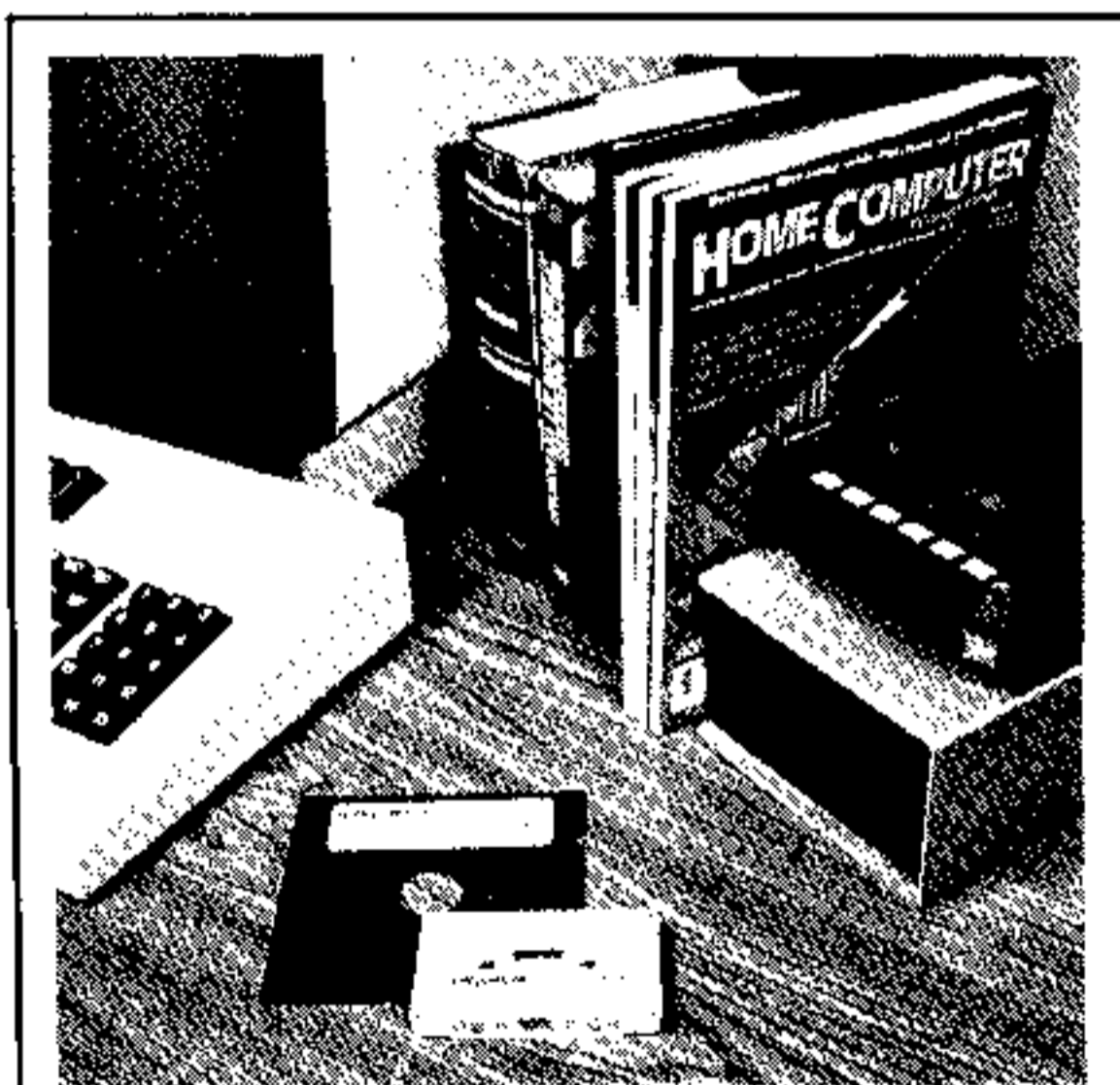
If you're missing any back issues of **HOME COMPUTER** magazine™ you're missing more than you'll ever know . . .

Having each issue of *Home Computer Magazine* readily at hand provides you with direct access to a valuable reference library of home computer knowledge—unequaled anywhere!

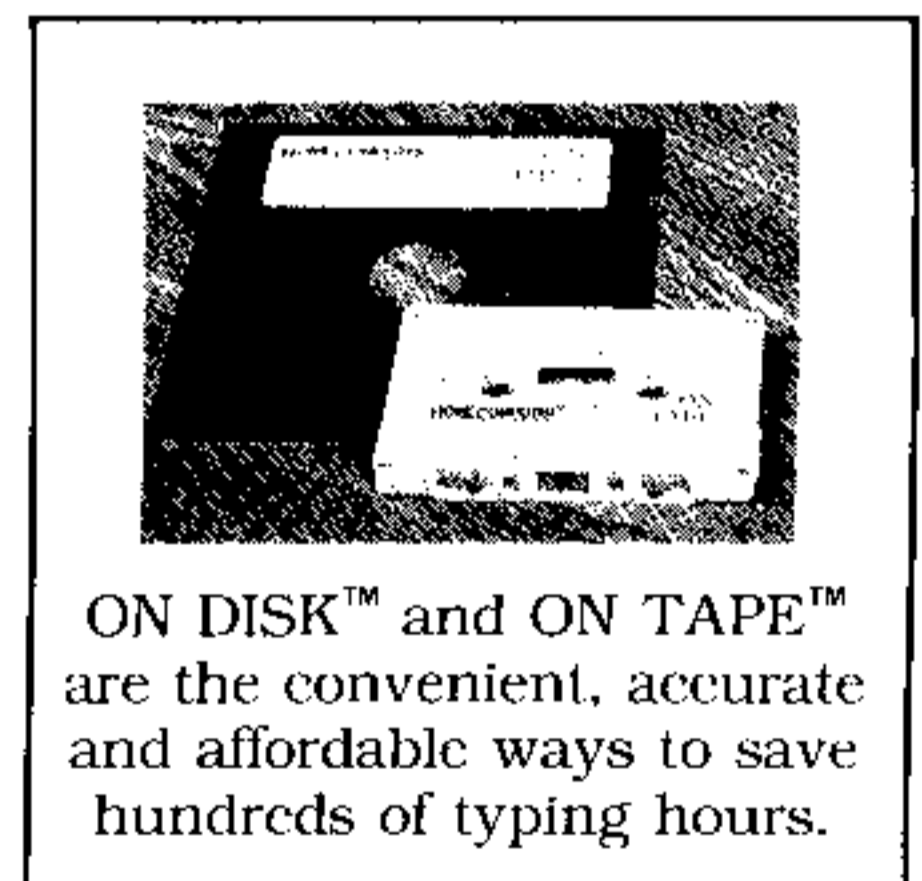
Back issues of HCM's program service—**ON DISK™** or **ON TAPE™** are also available.

Collect all the programs from each magazine issue on a ready-to-RUN quality floppy disk or cassette tape available in

separate versions for Apple, Commodore, IBM, and Texas Instruments home computers.



A valuable reference library of each *Home Computer Magazine* issue is the One Essential Peripheral™ for your home computer.



ON DISK™ and ON TAPE™ are the convenient, accurate and affordable ways to save hundreds of typing hours.

**“Safeguard” Your Home Computer Knowledge—
Order Valuable Back Issues Today!**

To Order, Use Bind-In Card at Center of Magazine.

FOR NEW READERS



The Plain & Simple Truth About **HOME COMPUTER**[™] magazine

Chock Full of Valuable Software & How-To Articles Without Filler

Every issue is a software "horn of plenty" with dozens of type-in-and-RUN programs printed in an easy-to-read listings format. Our programs are also available on inexpensive disks or cassettes for those who prefer the convenience of ready-to-RUN software. Step-by-step tutorials round out each issue, providing the solid facts you need without fluff or filler. Thus, each issue functions as an excellent reference work, as well as a valuable software source.



No Outside Advertising

Freed from the pressures of servicing *advertisers*, we concentrate on serving our *readers*. Each issue provides uninterrupted editorial flow and graphic layouts for better comprehension—plus unbiased product reviews which focus on true strengths and weaknesses, wherever the chips may fall . . . And we don't have to worry about losing advertisers because of publishing software in the magazine that is "too good." Consequently, we can provide the best free software available anywhere.

Focused on the 4 Hot Home Brands

We are 4 system-specific magazines under one wrapper—not a sprawling, "general interest" publication which attempts to cover too wide a field, only to spread itself too thin. The other side of the coin to this focused approach is the knowledge you gain from being exposed to the many tips, ideas, and techniques we provide for 3 of the 4 systems you may not even have. You'll learn more about your Apple, Commodore, IBM, or Texas Instruments home computer from this one magazine than from a host of more limited sources.

A Balanced Mix For a Perfect Recipe

In each issue we strive for a perfect balance of productivity, entertainment, education, utilities, and computer literacy—serving the needs of novice and pro alike. Every issue is a full-course meal, with a smorgasboard of tasty dishes for all palates. Whereas other computer magazines may dish out lumps of "editorial indigestion," we serve up a satisfying blend—one digestible byte at a time.

—Welcome to Our World of Home Computing

HOME COMPUTER

Home Computer Magazine (ISSN 0747-055X) is published monthly by Emerald Valley Publishing Co., P.O. Box 5537, Eugene, OR 97405. The editorial office is located at 1500 Valley River Drive, Suite 250, Eugene, OR 97401 (Tel. 503-485-8796). Subscription rates in U.S. and its possessions are \$25 for one year, \$45 for two years, and \$63 for three years. In Canada and Mexico add \$11 per year. Other foreign countries \$43 for one year surface mail. Inquire for air delivery. Single copy price in U.S. and its possessions is \$3.50, and \$4.50 in Canada and Mexico. Foreign subscription payment should be in United States funds drawn on a U.S. bank. Second-class postage paid at Eugene, OR 97401, and Columbia, MO 65201.

POSTMASTER: Send all address changes to **Home Computer Magazine**, P. O. Box 5537, Eugene, OR 97405. Subscribers should send all correspondence about subscriptions to above address.

Address all editorial correspondence to the Editor at **Home Computer Magazine**, 1500 Valley River Drive, Suite 250, Eugene, OR 97401. Unacceptable manuscripts will be returned if accompanied by sufficient first class postage and self-addressed envelope. Not responsible for lost manuscripts, photos, or program media. Opinions expressed by the authors are not necessarily those of **Home Computer Magazine**. All mail directed to the Editor or to the "Letters to the Editor" column will be treated as unconditionally assigned for publication, copyright purposes, and use in any other publication or brochure, and are subject to **Home Computer Magazine's** unrestricted right to edit and comment. **Home Computer Magazine** assumes no liability for errors in articles, programs, or advertisements. Mention of products by trade name in editorial material or advertisements contained herein in no way constitutes endorsement of the product or products by **Home Computer Magazine** or the publisher unless explicitly stated.

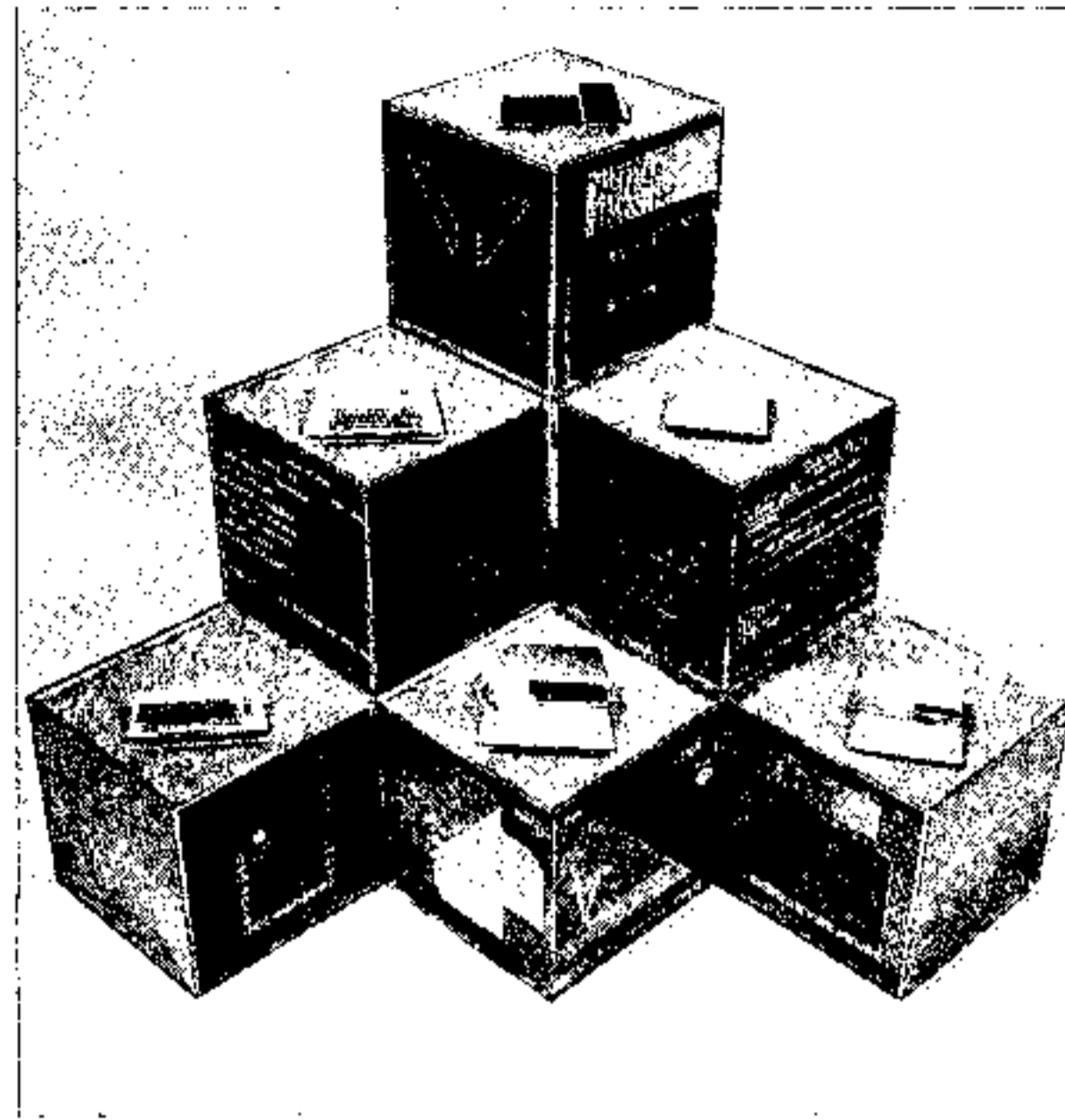
Each separate contribution to this December 1984 issue and the issue as a collective work is Copyright © 1984 by Emerald Valley Publishing Co. All rights reserved. Copying done for other than personal or internal reference use without the permission of Emerald Valley Publishing Co. is prohibited. Requests for special permission or bulk orders should be addressed to the publisher.

Limited License for use of programs in Home Computer Magazine. Emerald Valley Publishing Co. (EVP) is the owner of all rights to the computer programs and software published in this magazine. To allow for use of the software by the purchaser of the magazine, EVP grants to such purchaser only, the limited license to enter these programs into the purchaser's computer, and to place such programs on a diskette or cassette for the purchaser's personal use.

Any other use, distribution, sale, or copying of these computer programs without the written consent of EVP is expressly prohibited and in violation of this limited license and the copyright laws.

Home Computer Magazine, HCM, and Home Computer Digest are trademarks of Emerald Valley Publishing Co.

Publisher/Editor-in-Chief	Gary M. Kaplan
Executive Editor	David G. Brader
Managing Editor	Walter Hego
Associate Editor	Wayne Koberstein
Sr. Technical Editors	
	William K. Balthrop, Roger Wood
Technical Editors	
	D. Donaldson, Tom Green, G.R. Michaels, Steven P. Nelson, Patricia Swift
User Group Editor	Judy Campbell
Assistant Editor	Dana M. Campbell
Program Translators	
	Justin Bottero, Hendrik Broekhoff, Stephen A. Cordon, Ann Dahm, Jeff Fund, Jeff Jones, Robert Paschelke, Randy Thompson, Nancy Vendelin
Asst. to the Publisher	Rhea J. Grundy
Production Manager	Norman Winney, Jr.
Creative Director	Gei-Lei Gom
Typesetting	Curtis Byrd
Photography	
	Nelson Stevens, K.D. Wainsworth
Production Assistant	Rachel Knight
Customer Service	Tel. (503) 341-1029
	Sharon Hinshaw
Dealer Sales & Distribution	Tel. (503) 341-1036
	Paula Hoiland, Ken Reiling
Main Switchboard	Tel. (503) 485-8796



Outside HCM

Libraries have indeed changed. With your home computer, you now have the ability to house a vast library in a very compact space. But instead of shelves and books, the halls of *this* structure are lined with pixels and bytes. We picture on our cover the foundation of a great library—built not of piled concrete, but of original software piled high, and supporting the popular machines we cover. These are the actual programs available within the pages of this issue—a careful balance of entertainment, productivity, education, and utilities. So, as you start or continue to build your personal software library, we hope you'll turn to *this* Magazine for a readable supply of BASIC building blocks.

Inside HCM

Inherent in home computing is basic faith—a belief that the computer will make things better. Better work. Better play. Better education. And better creativity. At *Home Computer Magazine*, we have every reason to share in this faith because we've seen it *work*. We don't just *talk* about computers, we *use* them to make our magazine a better publication, and to help our readers become better served by their own machines.

According to psychologist Abraham Maslow, if the only tool you have is a hammer, you start seeing all of your problems as nails. Rather than provide just a "hammer," we instead open up an entire "toolbox" for your use. For instance, take a look at *Quiz Construction Set*, a brain-building software tool with enough built-in flexibility to suit teachers, students, or any amateur self-educator. With this program-kit, you can construct quizzes for yourself or others on virtually any subject. (Just to get you started, we include three complete quizzes—courtesy of our "quizzical" magazine staff.) As an additional self-teaching aid, we offer a computerized *Division Tutor*. This program will exercise young minds in the old-but-still-valuable discipline of long division.

Many other valuable key-in-and-RUN programs appear in this issue of *HCM*, including the *Personal Loan Calculator*—another power tool to help you calculate answers you need without becoming too "amortified." Anyone even *considering* taking out a loan will jump at the chance to get ahead of the borrowing game with this practical program. And after you've jumped into *that* game, there's no reason to avoid *Jumping Ahead with Game Programming*—a lively program/tutorial with a ready-made game to boot.

For the artistically inclined, *Sketch-64* provides a handy little graphics tool—at

the touch of a joystick—for Commodore 64 users. And if you're looking for pure enjoyment, try our two new game programs: scoop up schools of succulent fish with *Bird Brain*, or slide down a slippery path with *Slither*, the snake that snacks at every opportunity.

We have included our usual special features and how-to articles, like *Razzle Dazzle*, a small but dazzling bag of tricks for the TI-99/4A, and *Simon Sez*, a few good BASIC programming tips for the C-64. And for the benefit of all you new *llc* programmers, we provide some hot clues for *Putting the Puzzle All Together: Apple llc Programming Considerations*—an illuminating guide to maintaining program compatibility within the Apple orchard.

Orchards are fine, but you won't catch Junior standing under an Apple tree—especially when he's got some new keyboards to flaunt. There are, in fact, three new ones that we carefully scrutinize for you. Next, we doodle—in color—with a new, versatile artist's tool, *PCjr ColorPaint*. Finally, we cast off toward adventure... with a scintillating *Sailing* simulation and a colorfully animated *King's Quest*.

If your quest is to better understand new products before spending your hard-earned money this Holiday season, be sure to check out the rest of our reviews. You'll find such stocking stuffers as spelling checkers, speed reading courses, sketch pads, talking crickets...

And while the snow is drifting, you can be sifting—that is, sifting through a LOGO adventure which we start constructing in this gala issue.

So if you're seeking tools for improvement or bountiful activities for those cold nights ahead, you need look no farther than *Home Computer Magazine*.

Until next month, have fun reading, learning, and RUNing HCM

On Screen

By Gary M. Kaplan
Publisher & Editor-in-Chief

The cold winds of January will be nearly upon us by the time many of you read this message. I'd therefore like to take this opportunity to wish all of our readers a joyous and healthy holiday season and a successful new year ahead.

With the advent of 1985, *Home Computer Magazine* will be entering its fifth publication year—a major milestone for us here in the Emerald Valley, the terminus of the old Oregon Trail. Maybe it's the land . . . maybe it's the people . . . but we like to think that the pioneering spirit never died out here. Perhaps that's why we're so excited to be blazing new trails with our recently implemented no-advertising format. With real pleasure, we can now bring you uninterrupted editorial flow and graphic layouts that fully support our technical content—conveying it as clearly and concisely as possible.

As evidenced by the many thousands of enthusiastic letters and phone calls, our readers overwhelmingly support this change. We're absolutely thrilled by your favorable response. For the future then, our reader mandate is crystal clear: continue to provide plenty of useful software, understandable "how-to" activities, and candid exposures of product merits and flaws. In 1985, our software content will be quite comprehensive. Already on the drawing board are several exciting productivity, educational, and entertainment packages that you'll be able to immediately type in and RUN, or load from our inexpensive floppy disks and cassette tapes.

By the way, I don't use the word "exciting" loosely. These software packages really are exciting because they provide a BASIC taste of the new, bestselling, state-of-the-art software that could set you back as much as several hundred dollars in a computer store. Obviously, our programs must necessarily be scaled-down versions of these big-name products, but for some, our programs will be all that you'll need to do the job—if, in fact, the job really needs "doing" in the first place . . . For others, however, experience with our software may whet some appetites for the far-more-costly commercial versions. Coupled with our product reviews, this hands-on knowledge will make you a wiser, more discriminating purchaser when choosing from a confusing horde of similar products.



“ . . . our reader mandate is crystal clear: continue to provide plenty of useful software, understandable “how-to” activities, and candid exposures of product merits and flaws.”

In every facet of magazine and software publishing, we're striving for excellence—that is, to be the best at what we do. Being the *best*, however, takes ten times more effort than just being *good*. Consequently, it takes more time than a strict monthly schedule permits to prepare and debug each issue's software. Without cutting down on the sheer quantity of program coding in each magazine, we've

found it to be impractical to put out twelve top-notch issues each calendar year.

As a result, we've decided to solve this quality-vs-time dilemma by reducing our stated publication frequency to ten issues per year. Each issue will be identified by a volume and issue number on its cover and folio lines, rather than a stated month.

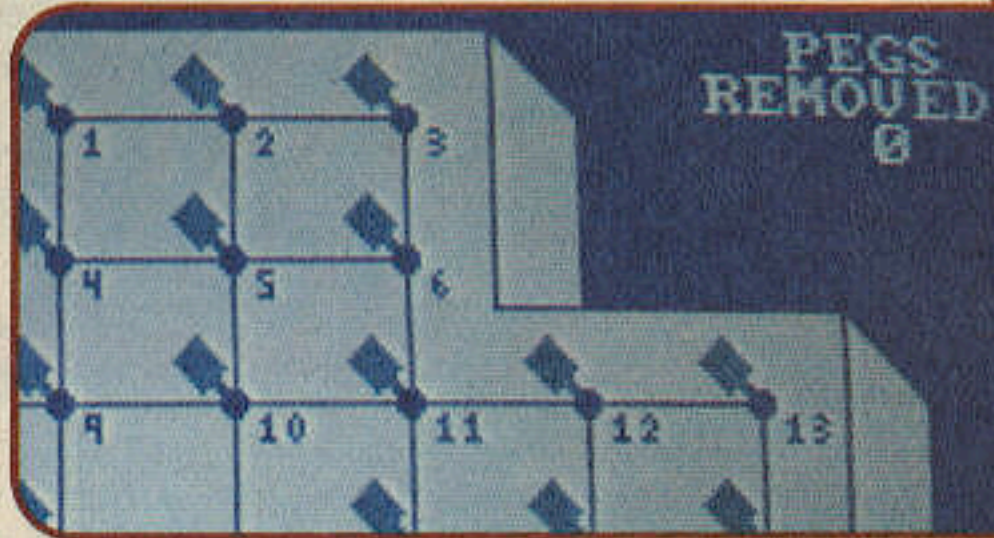
Present subscribers will still receive the correct number of issues they are entitled to based on their original subscription order. Renewals and new subscriptions entered prior to the cutoff date for publication of our next issue (Vol. 5, No. 1) will also be entitled to receive the magazine on this same 12-issue basis until expiration. All other subscriptions and renewals entered during 1985 will be based on ten issues per year, but new and renewing subscribers will still receive our free software media—ON TAPE or ON DISK—as premiums for joining (or rejoining) our “special family.”

In closing, I'd like to thank all of you for your support and patience during this year of transition. Please keep your suggestions and submissions coming—it's the creative energy that fuels *your* magazine. And don't forget to introduce *Home Computer Magazine* to all of your friends, relatives, and associates who are interested in getting the most out of their present or future home computers. This kind of personal, word-of-mouth publicity is the best way to help us grow, while sharing with others the magazine you've come to value and trust.



















Thank you.






















HOME COMPUTER™

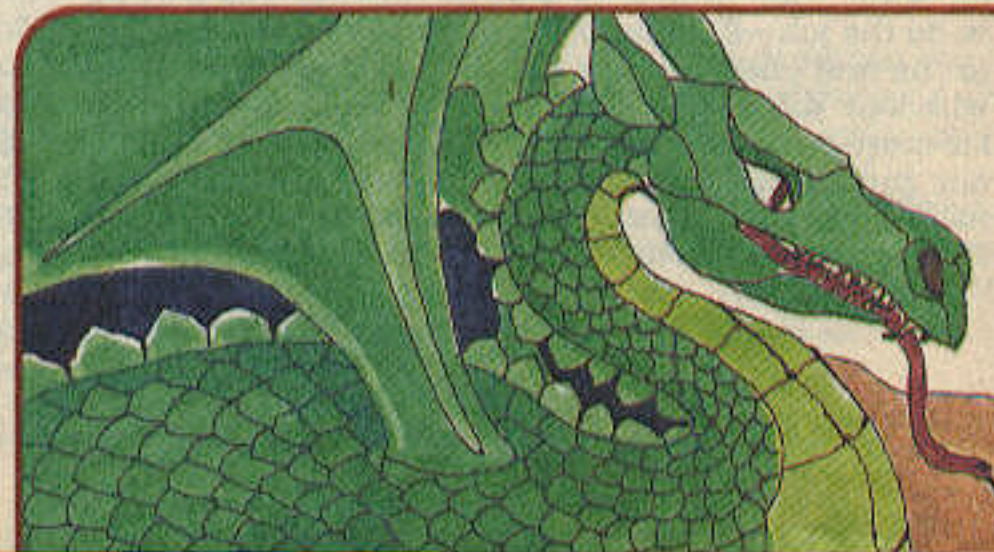
magazine



FEATURES

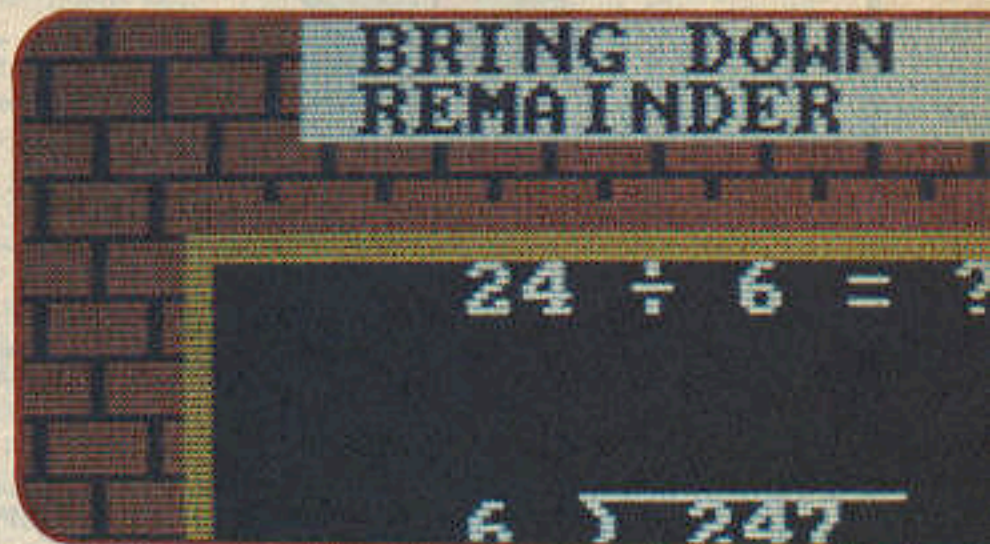
- 14 Quiz Construction Set**     
We provide the tools, you make it and take it. *by William K. Baithrop*
- 23 Personal Loan Calculator**     
Find out where your interest lies. *by H.W. Button and the HCM Staff*
- 26 Jumping Ahead With Game Programming**     
Learn the techniques from the inside out. *by Bob Stoffers and the HCM Staff*
- 31 Sketch-64** 
Let the joystick become your Commodore graphics link. *by James P. Chasse and the HCM Staff*
- 39 Simon Sez** 
New string-related commands from our learned friend. *by William K. Baithrop*
- 46 Razzle Dazzle** 
See how character manipulation enlivens TI graphics. *by William K. Baithrop*

- 58 Division Tutor**     
Long division meets the home computer. *by Steven Lisonbee and the HCM Staff*
- 61 Putting The Puzzle All Together: Apple IIc Programming Considerations** 
Secrets of programming the newest Apple. *by Peter Baum*
- 66 Bird Brain**     
Fly fishing takes on a new meaning. *by Craig Blazakis and the HCM Staff*
- 69 Slither**     
This slimy snake silently slithers and sneaks. *by Aaron Chew and the HCM Staff*
- 80 LOGO Clones: TI Graphics in a Turtle-Shell** 
Duplicate turtles do their thing. *by Sidney D. Nolte*
- 81 Build a LOGO Adventure**    
Part 1 of a series on creating interactive fiction. *by Andrew Keith and the HCM Staff*


















CONTENTS

VOLUME 4 NUMBER 5



PRODUCT REVIEWS

- 36 Race Across the Page: A Review of the Evelyn Wood Dynamic Reader**    
Speed reading taught in a new medium. *A Review*
- 38 99/4A Auto Spell Check** 
Proofread your spelling the easy way. *A Review*
- 40 PCjr ColorPaint** 
A mouse of many colors. *A Review*
- 42 Graphics Magic At Your Fingertips: A Review of Super Sketch**  
Sketch what you will on this new tablet. *A Review*
- 44 New Keys For Junior: A Review of Three Keyboards for the PCjr**  
No more Chicklets with this Good 'n' Plenty assortment. *A Review*

- 48 The Cricket!—A Sound Synthesizer And More For The Apple** 
More than just chirp—it sings! *A Review*
- 52 Sailing**  
Seafaring adventures in the Bermuda Triangle. *A Review*
- 53 Midnite Mason** 
Ghostly doings in a haunted building. *A Review*
- 54 King's Quest** 
Help Sir Grahame find the treasure. *A Review*

DEPARTMENTS

- | | | | |
|----|-----------------------|---------------------------|--------------------------|
| 3 | Welcome to HCM | 84 | Program Typing Guide |
| 4 | Inside/Outside HCM | 85 | Program Listing Contents |
| 5 | On Screen | 128 | DeBugs on Display |
| 9 | Letters to the Editor | Home Computer Tech Notes: | |
| 34 | HCM One Liners | 76 | Apple |
| 35 | HCM Review Criteria | 77 | Commodore |
| 50 | Group Grapevine | 78 | IBM |
| 56 | Industry Watch | 79 | TI |
| 71 | HCM Product News | | |





MAKE SOMEONE HAPPY!

GIVE A GIFT SUBSCRIPTION TO

HOME COMPUTERTM magazine

As a bonus gift we'll send YOU

or the GIFT RECIPIENT

2 issues of ON DISKTM or ON TAPETM

ABSOLUTELY FREE!

The Perfect Gift For Any Occasion!

Comments From Our Happy Readers

"You broke new ground when you introduced your unique typesetting style for your program listings; well, you have done it again by separating all of the listings in the magazine into one section in the center of the magazine. I call that genuine brilliance!
Warren Agee, Livonia, MI

"Well it happened again. Your magazine arrived in the mail, and I'm completely delighted with it."
Chris L. Chaffin, Omaha, NE

"I have subscribed to your magazine since its inception. I must say it has been most informative and has provided me with answers to many of my questions. Your feel for what the public wants is uncanny!"
Larry A. Hamel, Millington, TN

"I just received your August issue. I ordered a 3 year subscription exactly 1 year ago, and I have seen it grow in size and quality. This latest issue, with the separate section of program listings, reaffirms my wise subscription investment."
Mike Oliver, Clarendon Hills, IL

"When I saw the new version of your magazine I was elated! Naturally I subscribed.
Doug Barker, Exeter, CA

"I was a former subscriber to the 99'er Home Computer Magazine and I thought it was great. Then when I got the first issue of the Home Computer Magazine, I was twice as happy. It was a lot of information and great articles. Keep up the good work!"
Jenny Bures, Thousand Oaks, CA

"You have done a superb job of reaching other types of Home Computer enthusiasts and expanding your clientele while not depriving us 99'ers or leaving us by the wayside. The quality of the magazine is unsurpassed by any other, and I have looked at several different magazines! Hats off to you folks for your originality and continued endeavor to reach perfection."
John R. Stewart, Tucson, AZ

"...I am extremely pleased that a magazine such as Home Computer Magazine is around. I find the magazine extremely well written and of invaluable aid. . .keep up the good work with the magazine."
James L. Grigsby, Richmond, KY

Thanks to your thoughtfulness . . . your friends, family, and associates can enjoy a gift that keeps on giving all year through! They'll enjoy

12 ISSUES* of HCM

delivered right to their door. . .

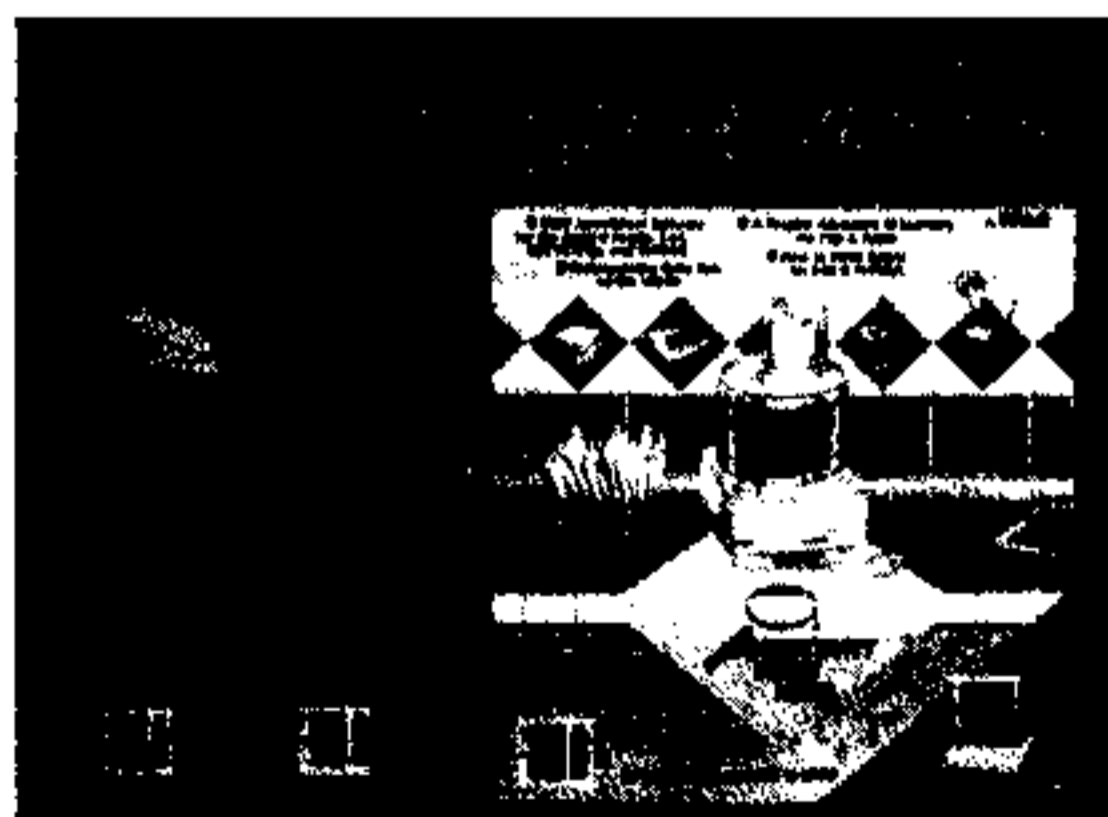
And you or the gift recipient will receive

2 Issues

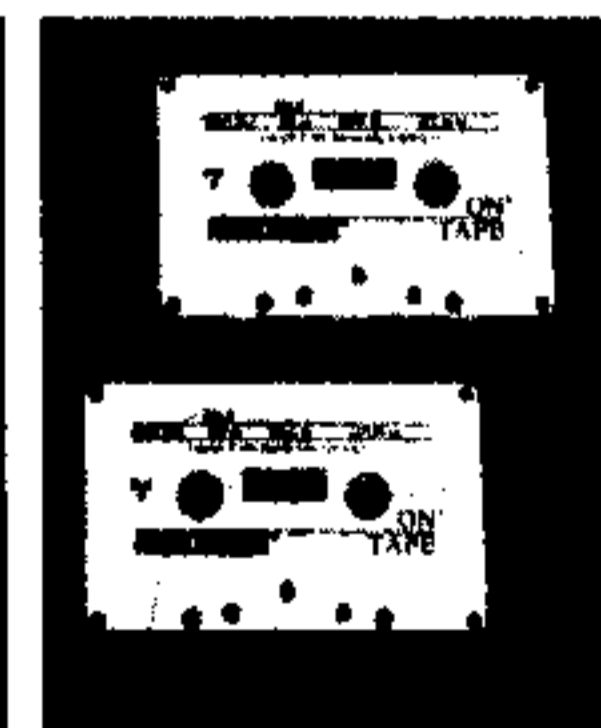
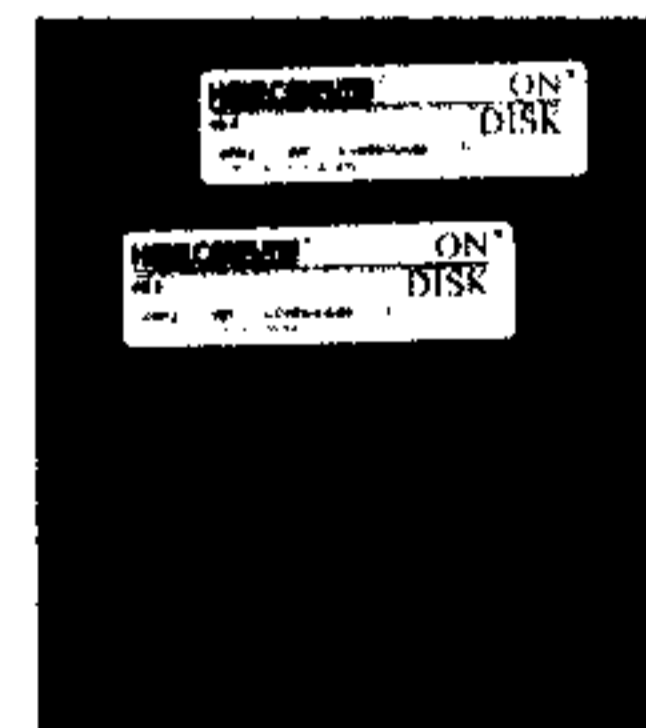
of our magazine program service

ON TAPETM or ON DISKTM

ABSOLUTELY FREE!!



Dozens of top quality key-in-and-RUN programs for Apple, Commodore, IBM, and Texas Instruments home computers appear in each issue. All of this for only \$25*.



FREE software—ON TAPETM or ON DISKTM—the same high-quality programs published monthly in the magazine. This cassette tape or floppy disk program service—normally a \$3.95 per issue* extra cost—is the convenient, accurate and affordable way to save hundreds of typing hours.

To Order A Gift-Subscription, Use Bind-In Card in Center of Magazine.

* Offer & Prices Subject To Change Without Notice.

Snap-Calc Fan Mail

[Note: If you are a Snap-Calc user, please consult the "Debugs on Display" section in this issue for updates to all versions of Snap-Calc. Although the program runs fine, we've corrected some minor bugs and have added a few enhancements (including negative number inputs and results).

If you have a companion subscription to the ON DISK media service, the Vol. 4, No. 5 issue will contain these Snap-Calc updates in a "mergeable" file. For details, see the "DeBugs on Display" section on page 128.—Ed.]

Wants Larger Fields

Dear Sir:

I have enjoyed the last 4 issues of *Home Computer Magazine* tremendously. I have owned a TI-99/4A for the past 18 months and have found that your publication leads the field in 99/4A information. Congratulations on a job well done. Keep up the good work.

The program Snap-Calc in your August issue is a very well-written piece of work. I have spent hours "crunching" numbers with the program. I have also spent much time (to no avail) trying to change the program to delete the decimal point so that larger numbers may be input and processed. Could you suggest a change to the program that would allow operation with six place numbers? It would be greatly appreciated.

Mark Georges
Niceville, FL 32578

Mark, Snap-Calc is a large program and one that lends itself to programming errors when modifications are attempted. Changing the field size for the numeric entries would have a ripple effect throughout the entire program. We will look into this as a possibility, and if we can provide this change as a simple upgrade of the software, we will include the enhancements on a forthcoming ON DISK issue.

Grand Designs with Snap-Calc

Dear Sir:

As soon as I receive the diskette with your Snap-Calc program for my IBM PC, I intend to use it to produce accountings for the various trusts that I manage. Since I don't need as many columns as your program supplies, but do need to have the capacity to display a million-dollar transaction, I am wondering if the following changes would produce a row or column display of \$1,234,567.

Line 210 . . . US \$ = "#,###,###." . . .

Line 220 [delete last two "+US\$+"]

William Hovey
Boston, MA 02108

As stated above, William, the answer to increasing the size of the fields in the Snap-Calc program is not as simple as just changing the screen format—there are many considerations to be dealt with. But due to the interest expressed by many of the readers, our programming staff is looking into the viability of increasing the field size without a major redesign of the software.

Sample Now ON DISK

Dear Sir:

I typed in Snap-Calc only to learn that I couldn't understand what or how to use it. The documentation is confusing when you read the general description, then the specific machine version, etc. Perhaps you'd consider printing a sample Snap-Calc program which would give us dummies a chance (1) to see it typed in without homemade bugs, and (2) to get a rough idea of how it might be used.

I've enclosed my renewal for your always-improving magazine and hope to see it again delivered to my door, so I needn't waste computer time going to the BX for my copy.

Joseph E. Bowker
Bergstrom AFB, TX 78719

Joseph, anyone purchasing the Vol. 4, No. 5 issue of our Magazine's programs ON DISK will find two additional files: TRIPCOST and SEATTLE. The first file, TRIPCOST, contains the spreadsheet template or logic (to be loaded using option 2 of the spreadsheet load screen). The second file, SEATTLE, contains the data for the TRIPCOST template (to be loaded using option 1 of the load option screen). We hope that the availability of this sample template and data file will encourage many less-experienced users to explore the fantastic possibilities that our Snap-Calc program offers.

You will be interested to know, Roswald, that we also "use both machines daily." All of our editors have a Macintosh as part of their technological arsenal. However, there are two factors that we must weigh when considering coverage of a new machine or language in the magazine: (1) Is there enough reader support for the machine/language to warrant adding the coverage? (2) Can we provide the added coverage without detracting from the machines/languages we presently include. In regard to coverage of the Macintosh, the time is not yet right.

Macintosh Coverage Wanted

Dear Sir:

I congratulate you on the best magazine for the home computerist. My family and I started out with the TI-99/4A computer, and now have an Apple Macintosh with 512K memory, Imagewriter printer, Microsoft BASIC, etc. We use both machines daily. I hope you plan to include articles and programs for the Macintosh in your future publications.

Roswald J. Allen
Fitzgerald, GA 31750

Great Screen Photos

Dear Sir:

I read with great interest your sprite tutorial, "Double Your Color—Double Your Fun" in *Home Computer Magazine* Vol. 4, No. 2. I noticed that the illustrations used were actual photographs taken from a CRT screen and was wondering if you could tell me what camera and settings were used.

I was thinking of using photographs to record various graphic displays and any information you can provide on doing this would be appreciated.

Victor Meeldijk
Bronx, NY 10462

Thank you for the compliment on the photography, Victor. Those particular CRT screen shots were taken with a Mamiya RB67 Pro with regular Kodak color film. The display was a Panasonic composite color monitor, and the shots were taken in a darkened room with a light meter reading from the brightest portion of the screen. Tip: If the light meter flickers, we use a setting between the high and low readings. Also, never use a shutter-speed faster than 1/30th of a second. Good luck in your photographic efforts.

Bugs in Junior's Second Drive?

Dear Sir:

I have had a TI-99/4A system for three years and have two slimline drives installed in the peripheral expansion box. After working with my new IBM PCjr for awhile, I was ready to find a way to add a second drive.

The *HCM* Vol. 4, No. 4 article about adding a second disk drive to the IBM PCjr is just what I was looking for, but the MODBOOT.BAT file doesn't seem to work. I keep getting errors on the first and fourth lines. Is there a bug in the program?

The hardware modification went just fine. I found the chips called for at a local electronics store and bought a Panasonic double-sided slimline disk drive at another local store. I built the cable by carefully removing the connectors from an old TI disk cable and reusing them with a new length of ribbon cable. I sure hope you can help with the software problem.

Mark Guseila
Canton, MI 48188

Mark, you found two bugs in the typesetting of the article. On page 86, at the top of the second column, the MODBOOT.BAT file is described and then listed in bold type. The first line of the listing reads: AO O:980, but should read: A 0:9080.

The second error occurs in the fourth line of the same listing, which reads:

OR BY (410),40

In fact, it should have brackets, not parentheses, like the following:

OR BY [410],40

We hope you enjoy your dual-drive system. Here at HCM we have several PCjrs configured this way.

Continued next page

C-64 Memory Trick or Treat?

Dear Sir:

I have found that if I enter POKE 644,255:SYS 58260 on my Commodore 64 and press [RETURN], the start-up message changes from 38,911 to 63,231 bytes free. That's 24K of extra memory! It seems too good to be true, but PRINT FRE(0) returns a value of -2307, which shows that the extra memory is really there. Even typing NEW cannot get rid of it. Does this mean that I can now write BASIC programs that are up to 62K long? What am I POKEing and SYSing to, anyway? Could it be the hidden RAM behind the ROM I hear so much about?

James Redd
Camden, OH 45311

You're right, James, it is too good to be true! What you have changed is a pointer used by the system, but you have not added any memory to the system. The BASIC interpreter will get confused and cause the computer to "crash."

Apple IIc with TI Printer

Dear Sir:

I am considering buying the new Apple IIc and have some questions you might be able to answer. Can I use my TI impact printer (Epson MX-80 with serial card) with the Apple IIc? If so, are any modifications required? Can I use the TI disk drive (Shugart 400L) from my TI expansion box as an external drive for the Apple IIc? Are any modifications required? Can I leave the drive in the expansion box, disconnect it from the TI controller card and connect it to the Apple IIc external drive port, then power up the expansion box to run the drive?

Any assistance you can provide will be greatly appreciated.

Jack A. Sharp
APO San Francisco 96301

The answer to your first question, Jack, is yes—you can use your TI printer (with the serial card installed) connected to the Apple IIc. We have done that here with no problem at all. Adding a TI disk drive (Shugart 400L), however, is not possible. The Apple II drive signals are not compatible with the disk drives used by the TI disk system.

To learn more about the IIc, Jack, we recommend that you read the article in the previous issue of HCM entitled "IIc: The Core of a New Machine" as well as its follow-up article in this issue "Putting the Puzzle All Together: Apple IIc Programming Considerations."

Needs Software for C-64

Dear Sir:

I am having difficulty finding "in-depth" literature for programming my Commodore 64. Retail store salesmen cannot help, and the one computer store selling Commodore 64's wanted to sell me hardware and software before finding out what I was really trying to program.

Briefly, I am an engineer engaged in estimating for a manufacturing and machine tool company. Armed with charts and a desk-top calculator, I estimate machining times, horsepower, and thrust on special machines. The "number crunching" ability of the computer would enhance my job enormously. The "hard copy" via a printer is invaluable to engineers, once we are awarded a contract for a machine(s).

My specific need is for a clear way to program an array and be able to pick out the information (by row and column), desired for subsequent use in my program. Printing the array on the screen or a printer is a waste—I need to pull only one piece of data out of memory and use it. Searching charts is what I am doing now.

I would appreciate your help because very little business programming is advertised for this computer.

John R. Johan
St. Clair Shores, MI 48081

John, we're not sure that we understand exactly what you're asking for. It sounds like you're looking for a data base manager for the Commodore 64 that will pass information into one of your own programs that you're using for doing calculations. This is not a "trivial pursuit." We do not have an immediate answer for you. Perhaps some of our HCM readers who have done something similar can help by making a recommendation for the appropriate software.

A User-Happy Letter

Dear Sir:

I do not wish to belabor the obvious. This is the most user-friendly magazine that I presently subscribe to that covers computers.

To show my enthusiasm, I have bought the Best of 99'er book and tape package and all of the back issues available.

I am glad that your efforts to make this magazine the best of its kind are succeeding.

Alexander Jaffe
Highland Park, NJ 08904

Thank you for the kind words, Alexander, and we appreciate your enthusiasm. For you and other readers with a TI-99/4A, we now have all of the software from the 99'er HCM back issues available on magnetic media—both ON DISK and ON TAPE. Besides being a real great value and a terrific way to expand your software library, magazine and media back-issue sales also go a long way in supporting your favorite magazine—a publication that is (by choice) devoid of outside advertising.

Wanted: C-64 Slave for 99/4A

Dear Sir:

I own both TI-99/4A and Commodore 64 computers. After many hours of programming with both machines, I have come to the conclusion that the TI-99/4A is a much superior machine to work with.

The TI has a Terminal Emulator and the Commodore has the Super Expander. While I concede that the Commodore has more memory and runs faster, programming with it is a chore.

If at all possible, I would like to utilize the Commodore as a "slave" unit to the TI-99/4A. Is this feasible, or should I put the Commodore on the shelf and forget the whole thing?

I would like to be able to have the TI machine access the Commodore's memory capacity. Is it as simple as connecting an RS232 cable between the two machines?

Robert W. Folsom, Jr.
Peabody, MA 01960

You pose a very interesting question, Robert. At this time we don't know of anyone who has used the Commodore 64 as a slave to the Texas Instruments machine. We agree with you, it would not be as simple as hooking up the RS232 cable between the two machines. The RS232 port is actually under the control of the processor in each computer, and in order for one processor to use the other computer's memory, the second computer's processor would have to be disabled. It should be possible to build an adapter to allow Direct Memory Access of the C-64's memory from the TI-99/4A main bus connector (on the right side of the console). This would require special logic to be designed and fabricated along with a special cable. We are not aware of the existence of such a device. But, this might be an interesting project for consideration—any adventurous spirits out there?

Tech Literature for the IIc

Dear Sir:

Fine magazine—keep up the good work. I hope you can help me. I have an Apple IIc and am looking for an Apple IIc Reference Manual. Do you know if any exist and where I might obtain one? Also, do you know of any books on assembly language programming of the 65C02, which is used in the Apple IIc?

Karen M. Lee
Westbrook, ME 0409

Karen, I think that the answers to your questions are probably contained in this issue's article on the IIc entitled, "Putting the Puzzle All Together: Apple IIc Programming Considerations" (a follow-up to last issue's article "IIc: The Core of a New Machine").

Words From South Africa

Dear Sir:

I found the back issues of HCM to be most valuable in getting MULTIPLAN and TI-Writer together. I have a FACIT 4510 printer on-line with the TI-99/4A. The FACIT 4510 is a quality matrix printer currently selling at \$1070 from some shops. The magic password (device name) which was discovered by a genius I found in Benoni, is

"RS232.BA = 9600.TW = 1.CH = N"

Incidentally, I work for Gillette South Africa, Ltd. as manager of their Affirmative Action Plan. We are a Sullivan signatory company. Once a year, a lengthy report is sent to the States to the Reverend Leon Sullivan in order to negative the disinvestment campaign being waged against American companies in the Republic. The Sullivan Report has to be audited for expenditures on our social responsibility activities. In May of this year I started using MULTIPLAN to record a year's expenditures. The resulting spreadsheet, all nicely laid out and pasted after printing, resulted in saving the auditors four days of work when compared with the time taken to audit the 1983 report. How is that for the TI-99/4A? This use of MULTIPLAN is a first in the Republic. You may wish to include this information in *HCM*.

G. E. Bagley
Republic of South Africa

Thank you very much for the information. We're sure that readers who have the Facet 4510 printer will be interested in learning of your "magic password."

C-64 to TI Printer—The Missing Link

Dear Sir:

In your Vol. 4, No. 4 issue of *HCM*, you gave Edmond Reynolds only half the answer on the method of hooking up a C-64 to his TI-99/4A printer. You are so right that the Cardco interface is necessary, but as I found out through much pain and suffering, that's not all folks.

Only after hitting every software store in my area and making "dozens" of frustrating long distance telephone calls to TI and Epson (after all, the TI printer is an Epson MX80 inside) did I finally get the answer from a TI technician.

In order to use the printer as a Centronics parallel, you must remove the serial board (the one containing the serial port) from your printer. To do this, follow the instructions in the TI printer manual for getting to the lower dip switches in the machine. Just remove the four screws holding the board down and disconnect the one wire hooked to the serial board. Gently remove the board (it's plugged into the lower board) and leave it out with the one loose wire hanging.

What you now have is a Centronics parallel Epson MX80 for use with your Cardco interface and the C-64 computer.

It's real easy to do, but what a job getting the information.

Allen Palazzo
Staten Island, NY 10304

You're right, Allen. We forgot to mention that in order to complete the hook-up of the Epson printer for parallel operation you need to pull out the serial board.

Apple Cassette Support On Wane

Dear Sir:

I am thinking of buying an Apple-compatible clone (I have been offered a new one at a good price). However, initially I won't be able to afford a disk drive.

I foresee my uses of the computer as being educational (for my five-year old son), entertainment (games, etc.—especially adventure games), probably some home management software, and a little experimental programming.

My main concern at the moment is whether or not Apple software is available on tape? If I type in programs from magazines, can I save them on tape? Are there any restrictions on doing this? (Do some programs look for a disk? If so, how can I tell?)

Thank you for any help you can give me.

H. King
Cornwall, Ontario, Canada

Apple has been phasing out its cassette software support for the Apple II Family. In fact, the Apple IIc doesn't have the software in its ROM to support a cassette port at all. Cassette-based software for the Apple II Family is very rare. When selecting an Apple program to key-in from *Home Computer Magazine*, read its accompanying article to see whether it requires a disk drive. The following articles in this issue include programs that can be used with a cassette, without any modifications: *Personal Loan Calculator*, *Jumping Ahead With Game Programming*, *Division Tutor*, *Bird Brain*, and *Slither*. Demand for Apple software on cassette is too low to warrant ON TAPE production.

Finding a TI p-Code Card

Dear Sir:

After spending \$30 in phone calls and about \$6 in postage, I need your help. Having subscribed to *HCM* (and *99'er*) for 2 years, your loyal readers are my last resort. I've nowhere else to turn.

I'm looking for the Texas Instruments p-Code card (UCSD Pascal). I've written to more than three-dozen supply houses and listed my name on dozens of bulletin board systems across the country, but *no one* knows where I can purchase the card.

If anyone has a p-Code card and/or the supporting cartridges or software available, I'd like to hear from them. This TI owner would like to complete his system, but even Texas Instruments, Inc. won't tell me *who* bought the remaining stock of Pascal software or the p-Code cards.

Don Graff
Byhalia, MS 38611

We have learned that the p-Code card and all the UCSD Pascal software packages are available, but in short supply, from Tex-Comp (P.O. Box 33084, Granada Hills, CA. 91344). The complete package price is \$119.95 (including software), and the firm requests that you order this particular package by mail only.

Junior Monitor Questions

Dear Sir:

The Vol. 4, No. 2 article on the PCjr mentions that to obtain screen mode six (high resolution), you must have a compatible direct-drive

RGBI monitor. I do not understand the difference between this monitor and composite color monitors, and I was wondering if the *HCM* staff could give me any help.

I would like to comment on your wonderful magazine. I love the reviews and the program listings. You have the best magazine I have seen in a long time.

Mark Guebert
Arnold, MO 63010

Mark, the three picture-tube color guns in the RGBI (red, green, blue, intensity) monitor are driven directly from computer signals. This allows detailed graphics to be displayed. A reasonably good RGBI monitor can be used to display 80 columns of text along with very good color graphics. In addition, the RGBI colors are "pure" in appearance.

A composite color monitor takes a single input of serially-coded (usually referred to as the "NTSC standard") information. This information contains the intensity signals for each of the three colors and is fed to the monitor in a serial fashion (red information followed by blue, followed by green, followed by red, etc.). All this must be decoded by the composite monitor and then used to activate the color guns in the picture tube. This method of transmitting information to the monitor makes it difficult to do precise positioning of color dots on the screen, causing "smearing" of the colors at points of color change.

TI Program Recovery

Dear Sir:

I have discovered some useful information for TI owners. Have you ever been typing in a long program and then accidentally pressed QUIT before you had a chance to save what you were typing? Well, if you have Extended BASIC and the 32K memory expansion, you can easily retrieve your lost program.

Here is an example of how to do this: First type a short program in Extended BASIC with the memory expansion on. Then type CALL PEEK(-31952,A,B,C,D) :: PRINT A;B;C;D The first two values shown point to the start of the line number table. The second pair of values point to the end of this table. Write down these numbers. Now press QUIT and re-enter Extended BASIC. Next, type CALL INIT Now you have to reload the values that you wrote down. Do this by typing CALL LOAD(-31952,W,X,Y,Z) replacing the W with the first number you wrote, the X with the second value, etc. Type LIST and presto! You have found your program.

There is only one drawback. If you add any lines to your program after you have found the four numbers, you must re-PEEK that address and get the four new values. If you do not do this, and then try to place the old values into memory, your computer will most likely lock up. I hope you find this information useful.

Mark Finkelstein
East Windsor, NJ 08520

Continued

Say, that's truly an interesting technique, Mark. It is one that can come in very handy when trying to debug a large Extended BASIC program. Thank you for the information.

Taking the C-64 for a Texas Drive?

Dear Sir:

I am writing to you to ask if there is any possible way to hook up our Commodore 64 computer to our Texas Instruments disk drive? If so, how and where can we purchase the items needed and at what cost? We would appreciate an answer as soon as possible. We subscribe to your magazine and would like to say keep up the good work.

Susan Jaszowski
Columbia, SC 29209

Sad to say, Susan, the Commodore 64 disk drive contains its own built-in disk drive controller and therefore is not compatible with the 99/4A disk drive which requires a separate external disk drive controller. If any reader knows of a disk drive controller for the C-64 that will work with SA400-style drives, please write to us.

HCM Wins Award

Dear Sir:

Congratulations! After nearly a month of comparing HCM to all the other computing magazines, I hereby proclaim you the winner of the first annual D. J. Branham Computer Magazine Contest!

As the grand prize winner, you will receive my subscription for two, not one, but two full years. In addition, I am also awarding you my subscription to ON DISK for 12 months.

This contest was conducted using my Apple IIc. This decision is final, as long as you continue to put out a truly superb magazine.

Seriously though, your magazine is far better than other magazines, even those that specialize in only one brand of computer. In future issues I would like to see you review and list entertainment and home management software which utilize the Apple mouse.

Thanks for a superior home computing magazine. Keep it up!

Danny J. Branham
Lawton, OK 73501

Gee thanks, Danny! We must be doing some things right—we are receiving lots of similar awards from all over the globe. We will watch for mouse-related software to bring to you, both in reviews and as an option in future HCM programs.

European Color TV Problems

Dear Sir:

I am a subscriber to your magazine and also an owner of a TI-99/4A computer. Your magazine is a great source of information for the home computer field.

I would like to ask you for a solution to one problem I have with my computer. I can't get a color picture on the screen. I have the U.S.

version TI-99/4A which produces a composite NTSC video signal with 3.58 MHz color carrier. I bought a color TV set (SONY KV 1423ME3) which is able to pick up PAL video signals with 4.43 MHz color carrier. I found in the circuit block diagram of this machine that the video signal is generated by the TMS9918A video processor chip. This chip is driven by a 10.7 MHz quartz resonator, and inside this chip is the new signal with a frequency of 3.58 MHz, which is probably the color carrier 3.58 MHz.

My question is: Can a replacement of a 10.7 MHz quartz resonator be made for a resonator with higher frequency—which would produce the color carrier of 4.43 MHz—or must the TMS9918A chip be replaced by a TMS9928 chip, which is used in the European version of the TI-99/4A? If yes, which components must be replaced?

I am considering buying an IBM PCjr computer—also U.S. version. I should like to get information on any simple arrangement for operation of this machine in NTSC 4.43 that can be made.

I think that my problem described above may be interesting to many owners of U.S. versions of personal computers outside the USA.

Thank you in advance for your answer.

Pavel Strihavka
Prague, Czechoslovakia

The TMS9928 used in the European model of the TI-99/4A interfaces to the video through encoder circuitry which includes its own 4.43 MHz crystal for compatibility with the PAL standard. The TMS9918A VDP found in US models of the TI-99/4A has the circuitry on-board the chip for producing the NTSC composite video with the 3.58 MHz color carrier. Both of these TI-99/4A Video Display Processors use a 10.7 MHz crystal for timing and should not be changed. We fear that your NTSC-compatible TI-99/4A is unable to interface with the Sony TV that you describe without major modification. If you did replace the TMS9918A with a TMS9928, you would still need to add the PAL composite video encoder circuitry as well. The situation with the IBM PCjr is about the same—we recommend that you purchase one designed for Europe.

He Finally Found Us

Dear Sir:

Several years ago when I first purchased my TI-99/4A, someone told me what a great magazine the 99'er was. After months of diligent searching, I finally gave up hope of ever finding a copy.

Imagine my amazement as I stood in a major department store trying to buy a microwave oven and there on the counter in front of me is this *Home Computer Magazine*, and in this little bitty Texas map it says "Continuing 99'er Magazine's coverage of the TI-99/4A." I snatched up both issues that were lying there and have been thoroughly engrossed in them ever since. My wife is still waiting for me to set up her microwave!

You have a marvelous magazine and a new dedicated reader.

J. David Schronce
Chicago, Ill. 60611

We're very glad you finally found us, David, and hope you enjoy your subscription. We also hope that word-of-mouth from you and other readers will get the message out that we are still "cooking." Now, why not be a good spouse and set up that microwave!

Getting a Jump on Junior

Dear Sir:

I'm writing to you regarding the article in Vol. 4, No. 4 of *Home Computer Magazine* about adding a second disk drive to the PCjr. I went through the procedure twice and both times ended up with the same result. I'm hoping you can help me.

I have a new IBM PCjr and a Percom 5-1/4" external disk drive (I believe that it is a Tandon 100-2). The literature that came with the drive doesn't show the jumper position for configuring it as a drive B:—this may be part of my problem. When I connect the new flat cable and turn on the power, the disk spins no matter which way I turn the cable (the red light comes on only momentarily). However, when I turn on the computer, it stops spinning. Also, the computer will not read drive A: (on power up, it reports ERROR H). If I do get it to read drive A: (by pressing ENTER after ERROR H), it doesn't function properly. It starts reading a file and then reports a sector-not-found error.

Mark Beifuss
Bryan, TX 77801

Mark, from your description it sounds like you should check two items, both of which are in the Tandon 100-2 disk drive: (1) Make sure that the load resistor IC pack has been removed from the drive (this can cause ERROR H problems). This load resistor pack is located near the ribbon-cable connector and is usually the only IC that is inserted into an IC socket. (2) When accessing drive A:, notice whether the external drive's red light also comes on. If so, you still have not found the right address selection as described on page 85 of the article. (If both drives are accessed at the same time, a "sector-not-found" error will most likely occur.)

Port-to-Port on the C-64

Dear Sir:

Is it possible to change the C-64 device number of the printer port with a POKE, or something from #4 and #5 to #2, so that an RS232 serial printer could be used with some of the canned software that do not offer this option?

Can the Commodore 64 computer be run and programmed from a remote terminal through the RS232 port? Do you know of a POKE statement or simple program that would allow this?

Scott Schultz
Athens, GA 30606

No, Scott, the hardware design of the C-64 will not allow the redirection of the "canned" printer output functions to the RS232 port. The answer to your second question is essentially no, but for clarification read the answer to the related question from Mr. Folsom (see previous letter, "Wanted: C-64 Slave for 99/4A").

Never Too Late for Software

Dear Sir:

Several comments I would like to make:

No advertising—Wow! Did not think this could be done in a computer magazine. I think the letter from Sandy Foote [Vol. 4, No. 4] proves your point well.

Most of your assembly language programs seem to involve the Mini Memory module rather than the Editor/Assembler. How about more parallel listings?

My only real complaint is that I did not discover your magazine sooner, particularly when it was strictly for the TI-99/4A.

Donald L. Mahler
Newton, MA 02159

We're glad you approve of our new format, Donald. Editor/Assembler source code listings for all those old Mini Memory programs are now available ON DISK for past issues. These disks also have the Mini Memory object code files. By the way, you don't have to feel sorry that you didn't discover us sooner because "The Best of 99'er—Vol. 1" and later back issues are still available—while supplies last. See the center bound-in order card and additional information on the inside rear cover.

Commodore Ribbon in the Black

Dear Sir:

Reading about the trouble your readers are having in locating printer ribbons for the Commodore 801 printer prompted me to write about how I solved the problem.

I bought an inkpad inker from my local office supply store. Whenever necessary, I re-ink the felt pad of the printing mechanism. If you ink the felt pad too much, the first couple of lines will be too dark. This darkness will soon disappear.

Hugh A. Valliant
Toronto, Ontario, Canada

Thanks, Hugh, for the tip on the Commodore 801 printer ribbons. It sounds a little messy, but it should extend the life of a few ribbons.

Is Junior's a Single or Double?

Dear Sir:

I have a question that no one thus far has been able to answer for me. I have the double-sided 360K disk drive on my PCjr with which I use both single- and double-sided disks. Sometimes when I either DIR or format a single-sided disk, I'll receive a message indicating that I have the byte capacity for a

double-sided disk! However, with some brands of single-sided disks, I'll get the correct capacity of 180K.

How does the double-sided disk drive "know" or distinguish between a single- and a double-sided floppy disk?

My best wishes to you on your new magazine format. It sounds like quite a challenge to omit all outside advertising. I sure hope you succeed. We've all seen a lot of computer magazines go under this year. Yours is the only one left that gives fair space to the PCjr. Keep up the good work.

Brad L. Barnes
Redwood City, CA 94063

You may purchase a diskette that is "rated" as a single-sided diskette, Brad, but if you format it yourself without specifying that it be formatted as a single-sided diskette, the PC-DOS FORMAT command will automatically default to a double-sided format. Most diskettes—whether they're marketed as single-sided or double-sided—will work as double-sided diskettes.

The PCjr PC-DOS 2.1 records what is called a "boot track" in the very first part of track 0 on each diskette as it is formatted. Information recorded in this area includes the format of the entire disk (single-sided, double-sided), the number of sectors it has per track, etc. This is the information accessed by the DIR command.

Tech Note Request Granted

Dear Sir:

In Vol. 4, No. 2 you had a most desirable and useful page in the "Tech Notes" section—the page on the TI-99/4A that was written by William K. Balthrop. Is there any way you can get some more of these unusually valuable ideas from him?

Specifically, I might mention the desirability of a basic routine that would take the place of the Extended BASIC command +LINPUT+.

Hope to see some more of his ideas.

Joel Martin
Fort Lauderdale, FL 33314

Ask and ye shall receive, Joel. See this month's TI-related "Home Computer Tech Note," which describes how to generate a LINPUT statement in TI BASIC.

Reading Format Enjoyable

Dear Sir:

I just finished reading the latest issue of your magazine in its new format. You have taken a bold and courageous step to break out of the traditional mold, and I for one applaud you for it. For the first time ever, I was able to read a computer magazine all the way through and found myself enjoying it because of the lack of clutter and the ease of continuity—much like seeing a good TV program without the annoyance of commercials.

You have a great magazine and you have just made it better. Keep up the good work, keep serving the TI user as long as possible, and good luck in a tough environment.

John F. Banocy
Englewood, FL 33533

Thanks for the vote of confidence, John. We especially like the TV analogy—it made our publisher feel like Alistair Cook on PBS's Masterpiece Theatre.

Apple User Likes It "Jam-Packed"

Dear Sir:

I am an Apple user who is very pleased with your magazine. You have many programs in an easy-to-follow format and with no outside advertising, this publication is superb!

I am interested in starting a software exchange program for Apple. I would appreciate it immensely if you could help me out by either writing to me or by printing this letter in your magazine.

One last word. Your magazine is jam-packed full of useful information and is one of the best in the field.

Thank you.

Mike Huston
Mendon, MI 49072

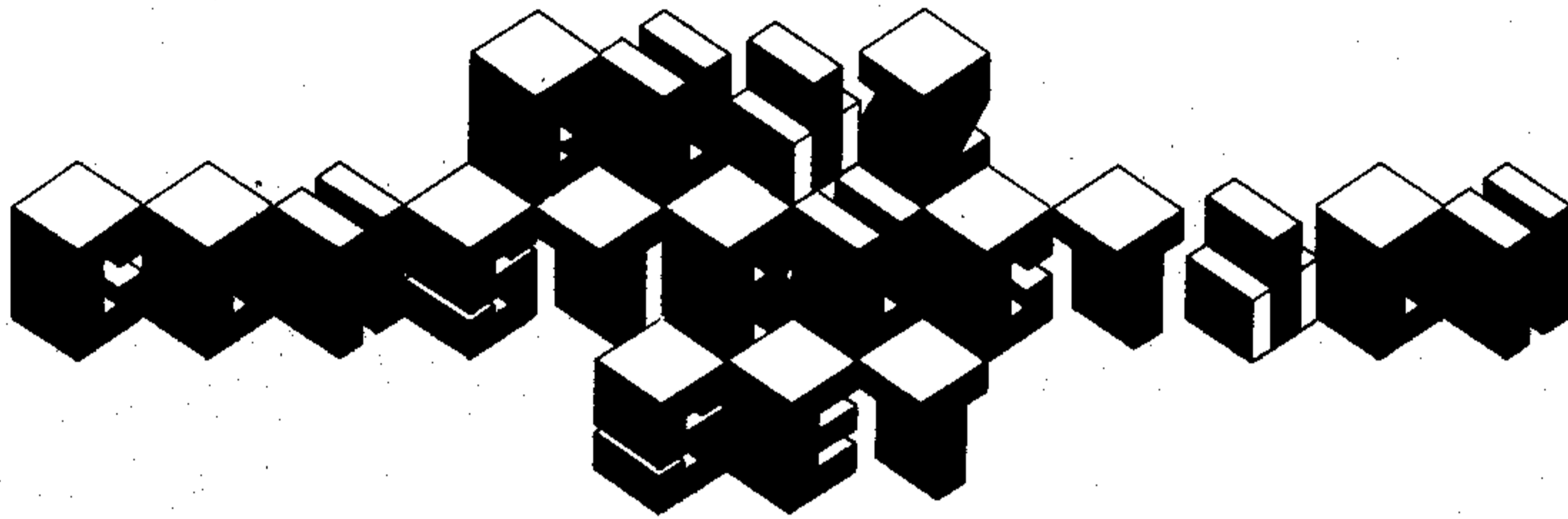
We appreciate your support and praise of Home Computer Magazine, Mike, and will do our best to continue to fulfill your expectations. One caution to you and others who start software exchanges: Beware of copyright infringements on the software appearing in Home Computer Magazine or in other magazines. Our very existence depends on reader support for software sales of ON DISK and ON TAPE. The software provided in Home Computer Magazine is protected by copyright and is not public domain software. Be very careful that any software put into your exchange is in the public domain. If you'll send us details on how the exchange will operate, we'll see whether we can help you get started. Good luck in your endeavors.

HCM

Special Announcement:

Home Computer Magazine is looking for "One-Liners".

If you have written a 1-line program in any language that is available on the computers we cover, send it in addressed to *Letters to the Editor*. It may win a prize! Now turn to page 34 to enjoy the best of this issue's One-Liners.



by **William K. Balthrop**
HCM Staff

**Calling all teachers, students, trivia and non-trivia buffs—
All who seek self-improvement and greater knowledge . . .
Create your own questions and find your own answers
with this do-it-all quiz machine!**

In the beginning, there was the question. Then came an answer—and the first quiz was born.

Many questions—and answers—have resulted from mankind's sometimes trivial, sometimes not so trivial pursuit of information and knowledge. Besides asking the eternal questions common to every generation, people are devoting more and more of their time to educating themselves in every area of human interest. And, at work or at play, the basic question and answer quiz is still a favored learning and teaching aid.

A quiz may be spoken or written—or nowadays, designed, stored, and taken on a home computer. Just a few years ago, the "teaching machine" was pretty much a joke—a complicated electro-mechanical device stuck in some school lab, and probably bolted to the floor. Now, teaching is but one natural function of a much smaller, multi-purpose device. With your computer and the program published here, you can do everything you could have done with that bulky old machine—and much more.

Quiz Construction Set is just what it says: a program that provides all the pieces you need to build, store, and retrieve your own direct-response quiz. It is perfect for school or home learning situations—and can provide a good deal of entertainment as well. You may enter questions of virtually any type, on any subject, with accompanying answers. Use them to exercise your own memory or someone else's. You also may select one of two types of clues to accompany a question, and determine how many chances will be given to get the right answer. As you take the quiz, the program keeps a running score of both right and wrong answers—and also checks your answers for correct spelling, tabulating a score for that as well. (For more on spell-checking, see "Taking The Quiz," page 17.)

Two In One

This software is actually two programs in one package: *Quiz-Make* and *Quiz-Take*. Here again, the names tell you what to expect. You may use the first routine to construct and store your quizzes, and the second to retrieve and take them. This structure serves several purposes: First, it frees up memory to hold larger quizzes. Second, it offers a form of security: If you don't want the quiz takers to be able to modify the quiz, you can give them only *Quiz-Take*, which has no provisions for making alterations in either questions or answers.

Quiz-Make

Use this program to create and modify a quiz before you use *Quiz-Take*. The type of quiz you create is limited only by the total number of questions, your system's memory capacity, and the size of the question and answer fields. The size of these fields is limited to two screen lines for a question, and one screen line for an answer. The maximum screen width is 40 characters on the IBM, Apple and Commodore computers, and 28 characters on the TI-99/4A.

Let's go through the process of setting up a simple quiz. After loading and running the program, you will be shown a title screen. To progress to the Main Menu, press either (ENTER) or (RETURN). You will see six choices:

- 1) EDIT
- 2) LOAD
- 3) SAVE
- 4) PRINT
- 5) CHANGE PARAMETERS
- 6) EXIT



To start, press 1. If you were modifying an existing quiz, you would simply begin entering your new questions and answers. Because this is a new quiz, you will be taken to the parameter setup screen and asked to design your quiz:

QUIZ TITLE — Enter the title of the quiz. This title will be displayed on the top of the screen during the Edit mode, and while a person is taking the quiz.

AUTHOR'S NAME — Enter your name here if you are the quiz's creator.

QUESTIONS HEADER — Enter the prompt you would like to see above all of the questions. This could be the name of a category, or simply the word "Question."

ANSWERS HEADER — Enter the prompt you would like to display above the answer field.

QUIZ TYPE: 1. SEQUENTIAL
2. RANDOM

If you press 1, the Sequential option, the quiz will be given in the same order that you enter the questions. Option number 2, Random, means that the questions will be selected at random from the entire quiz file.

PERCENTAGE OF LETTER CLUES (0 - 80) — This option determines how many letter clues will be given for a missed answer. When the Letter Clues option is selected in the *Quiz-Take* program, the student is shown a few of the letters from the answer. The number of letters given is calculated by multiplying the letter-clue percentage times the total number of letters in the correct answer. The spaces where letters are not displayed are filled with asterisks. A 50% letter clue might look like this:

INTERPOSITION (correct answer)
TE*P ITI** (letter clues)

It's possible that fewer clues than the percentage you selected will be displayed. This will happen if the program chooses the same letter twice. In the example above, if the program had twice picked the first T in the word INTERPOSITION there would then be one less letter clue displayed. You should think of this option as a *maximum* letter-clue percentage.

"You may enter questions of virtually any type, on any subject, with accompanying answers. Use them to exercise your own memory or someone else's"

TIME (IN SECONDS) RESPONSE DISPLAY (0 TO 99) — This prompt is asking you to enter the length of time the correct answer will be displayed when a person enters a wrong answer. It will not affect the length of time one has to enter an answer. There is no time limit for a response.

After entering this information, the program will re-display your entries and ask you if they are all

correct. If they are correct, then press Y. If you wish to change anything, press N. The program will then repeat the setup routine, asking you to re-enter all of the values.

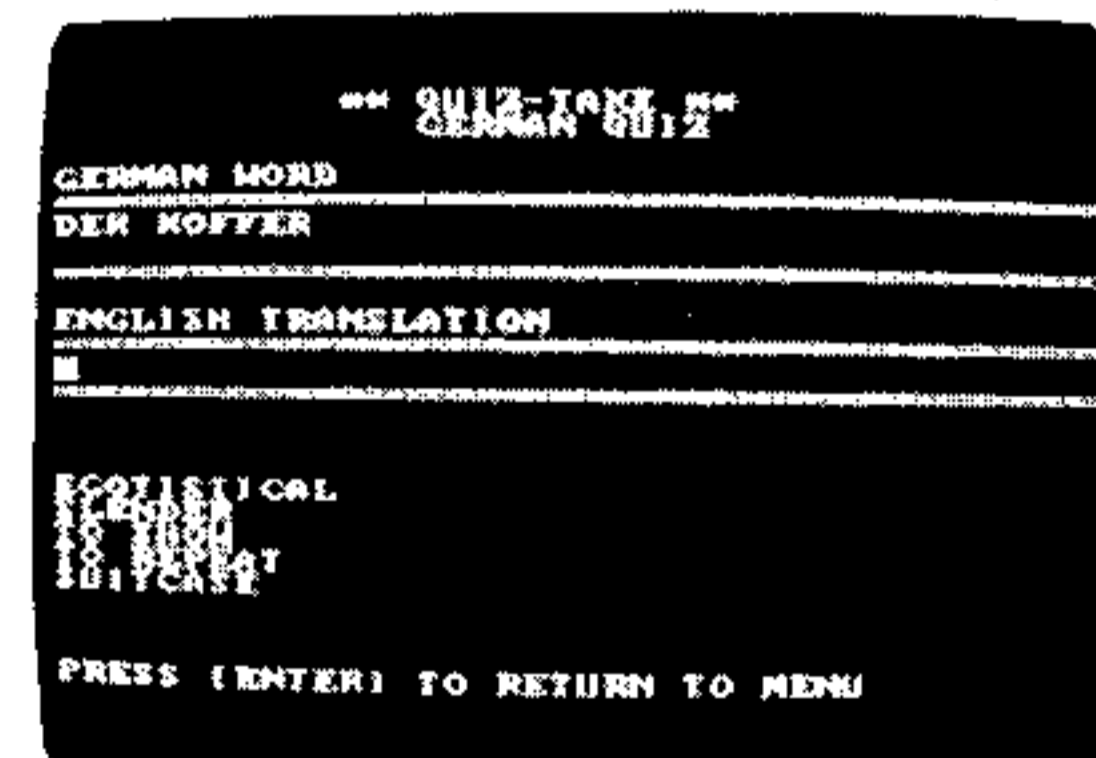
Editing A Quiz

When you conclude the setup, you are taken to the Editor screen. The top of the screen displays the title and the current record number. The record number should be #1, the first record. Below this are two entry fields for the questions and answers. Above each field is the field prompt that you created on the setup screen. A sample question header for a quiz which teaches German might read GERMAN WORD, with the answer header ENGLISH WORD.

The cursor should be flashing inside the questions field. To enter a question, simply type it in. Completely filling the question field automatically transfers you to the answer field.

Enter GESUNDHEIT into the question field and, because it does not fill the question field, press (RETURN) or (ENTER) to terminate the input. After you enter the question, the cursor will move down to the answer field. The answer field is only one screen-line in size. Enter the answer GOOD HEALTH and press (RETURN) or (ENTER). The question and answer fields will clear, and the cursor will reappear in the question field for the next question. The record number at the top of the screen should now read #2.

This photo shows the IBM version administering a German language quiz. The Word Clues option displays five words at the bottom of the screen, one of which is correct.



Go ahead and experiment with your own questions and answers. When you are ready to save your quiz, press (RETURN) or (ENTER) when either the question or answer field is empty, and you will return to the Main Menu screen. If you accidentally do this before you've finished entering questions and answers, simply select 1) EDIT again to continue editing. If you exit from a blank answer field after entering a question, the question you entered for that record will be lost and you will need to re-enter it.

Searching And Changing Records

Back in the Edit mode, you can remodel the quiz you have built so far by searching for a string of characters in either the question field or the answer field. The keys used to select the search vary from

system to system, so check the Control Capsule for your machine. You can select either a question search, or an answer search. Once selected, the words **SEARCHFOR** will appear above the field. Enter the string that you want to search for in this field.

For example, if you want to locate the first question you entered, **GESUNDHEIT**, select the Question Search option, then enter **GES**. The program then searches for questions with the letter combination **GES** in them. You could have entered **HEIT**, or **GESUND**, and the search would have located the first record. When a record is found, its contents are displayed in the question and answer fields. If there is more than one record which matches the search characters or words you enter, then you can select the Next option from the choices listed below the question and answer fields:

PRESS C-CHANGE N-NEXT E-EXIT

If you press **C**, both the question and the answer fields will be cleared, and you will be able to re-enter them. Each time you press **N**, the program will continue to search for the next occurrence of the search string you entered. You can keep searching—every record if necessary—until you find the record you want, or reach the end of the file. If you reach the end of the file and there are no other matches, the program will return to normal Edit mode, and the first blank record. This is also true when entering **EDIT** from the Main Menu.

If you wish to discontinue the search, press the key associated with **EXIT**. The actual key used to exit varies from system to system, so you will need to read your screen display or refer to the Control Capsule for your machine.

Save The Quiz File

To save your data, return to the Main Menu mode by pressing **(ENTER)** or **(RETURN)** in either field without entering anything else. Press **3** to select **SAVE** from the menu. The screen will clear, and then you will be asked to enter the following information:

QUIZ FILE NAME:
TODAY'S DATE:
YOUR NAME:

The **QUIZ FILE NAME** should be the file name you wish your quiz to have. On some systems you may be asked to also enter a device name or type of device, e.g., disk or tape. For **TODAY'S DATE** and **YOUR NAME**, you can enter anything you want to keep a history of the file. This information is displayed when the quiz is printed to the screen or a printer.

Once the save is complete, the program will report how many records were saved. To return to the Main Menu after saving, press **(ENTER)** or **(RETURN)**.

"This software is actually two programs in one package . . . the first to construct and store your quizzes, and the second to retrieve and take them."

Load The Quiz File

Once you have created and saved a quiz to tape or disk, you may want to work on it again to expand it or make modifications. You can load the quiz by selecting option **2** from the Main Menu. You will be asked for the quiz file name. On some systems you may be asked to enter the device name. Enter the

same file name used when you saved the quiz. The program will display information about the file as it's loaded:

```
title
LAST MODIFIED ON date
BY author's name
QUESTIONS: question header
ANSWERS: answer header
THERE ARE XX RECORDS
READING RECORD # XX
XX RECORDS LOADED
PRESS ANY KEY TO CONTINUE
```

Printing The Quiz

To list the quiz file contents for review, select option **4** from the Main Menu. You can list the quiz either to the screen, or to another device.

The information listed consists of the quiz parameters entered for the quiz on the parameter setup screen, followed by each question and answer in the quiz file.

Next issue we will present a third program, *Quiz-Print*, which will allow teachers to prepare hardcopy quizzes on a printer. With this program you will be able to format printed quizzes with a large number of options, as well as produce an answer sheet for grading purposes.

Change Parameters

If you have already created a quiz but would like to change its original parameters, select option **5**. This will take you to the setup screen, and will ask you to re-enter all of the parameters. After entering them, you will be asked whether they are correct. If so, press **Y** and you will be returned to the Main Menu.

Exit The Program

If you have a quiz in memory and have made changes to it, then you will be notified before leaving the program, and be given an opportunity to return to the Main Menu. From the Main Menu, you can save the changed quiz, and then exit the program.

You can exit the program *without* saving the changes simply by indicating this when the exit routine warns you. If there have been no changes to the quiz, the exit routine will not stop you when you use option **6**.

Quiz-Take

This program is used to take or study a quiz. You cannot alter the quiz from this program. If you wish to create or change a quiz, you must first use *Quiz-Make* to build or alter a quiz file.

After loading and running this program, you will be presented with a title screen. Press **(ENTER)** or **(RETURN)** to go to the Main Menu:

- 1) TAKE QUIZ
- 2) LOAD
- 3) STUDY QUIZ
- 4) EXIT

To select an option, simply press the number beside it. You do not need to press **(RETURN)** or **(ENTER)**.

1) **TAKE QUIZ** — Before you can use this option to take the quiz, you will first need to use option **2** to **LOAD** a quiz.

2) **LOAD QUIZ** — This option must be used to **LOAD** a quiz file previously created with *Quiz-Make*. If you haven't yet created a file with *Quiz-Make*, then refer to the previous section on running that program. When you select this option you will be prompted to enter

a file name. On some systems you will be asked to also enter the device name—e.g., tape or disk, drive 1 or drive 2. The program will display the number of records read in from the file, and then wait for you to press (ENTER) or (RETURN) before continuing back to the Main Menu.

3) **STUDY QUIZ** — This option allows you to study a quiz. Four questions and answers will be displayed on the screen at a time. You can then scroll through the list of questions and answers by pressing the appropriate keys. (The keys used for each system are described in the Control Capsules included with this article.) You can exit this mode and return back to the Main Menu at any time by pressing the appropriate escape key, also described in the Control Capsules for each system.

4) **EXIT** — There are no restrictions in exiting this program as there are in *Quiz-Make*. You may exit the program at any time you like. You will never cause the loss of data by exiting the program because this program can not alter any files you have created. The only thing that may be lost by exiting the program is your score—and possibly your pride . . .

Quiz Level

After selecting the **TAKE QUIZ** option you will be shown another menu screen. This screen is used to select difficulty level of the quiz and the type of clues, if any.

1) WORD CLUES	2 TRIES
2) WORD CLUES	1 TRY
3) LETTER CLUES	3 TRIES
4) LETTER CLUES	2 TRIES
5) NO CLUES	1 TRY
6) SAME QUIZ	
7) EXIT	

1) & 2) **Word Clues** — If you select options 1 or 2, you will be given a list of five answers at the bottom of the screen for every question. One of those five answers will be the correct answer. The answers displayed are selected completely at random from all of the answers in the quiz file, so that each time the quiz is taken, or the same question is asked, there will be a different list of possible answers.

In option 1 you have two chances to answer a question correctly. If you miss the answer on the second try, a spelling check will be done to see whether you simply misspelled the word.

If you select option 2, you must answer each question on the first try. If an answer is incorrect, then the program will perform the spelling check.

3) & 4) **Letter Clues** — Options 3 and 4 will give you letter clues if you enter the wrong answer. The letter clues were explained in more detail in the *Quiz-Make* section "Percentage of Letter Clues."

In option 3, you are given three opportunities to answer a question. On the first try, no clues are given. If you miss the answer on the first try, then clues will be displayed in the answer field, with asterisks indicating character positions where a clue is not given. You can then type right over the clues and asterisks to enter your next answer.

If the second try is wrong, you will be given a new set of letter clues, and another chance at answering the question. If you miss the question on the third try, a spelling check will be performed.

If option 4 is selected, the quiz will act just as it did for option three, except that only *one* set of letter clues will be given. If you miss the answer on the second try, a spelling check will be performed.

5) **No Clues** — Option 5 will not give you any clues to the right answer, and will only allow *one* try to

answer the question. If the answer is wrong on the first try, a spelling check is done to see if the answer has been misspelled.

6) **Same Quiz** — At any time during a quiz, you may return to the Main Menu screen by pressing (ENTER) or (RETURN). You may resume the quiz where you left off by selecting option 1 from the Main Menu (**TAKE QUIZ**), and then selecting option 6 from the quiz level menu (**SAME QUIZ**). The same quiz will be resumed with the score you had at the time you exited the quiz. If you select a quiz level other than option 6 (**SAME QUIZ**), the score will reset to zero and the quiz will start over.

7) **Exit** — Selecting option 7 will return you to the Main Menu.

Taking The Quiz

After selecting the quiz level, you will be taken to the quiz screen. This screen looks just like the one used for editing the quiz in the *Quiz-Make* program.

If the quiz is set up for sequential operation, then all of the questions have a predetermined order—they will be given in the same order in which they were entered. A question will be displayed in the question field, and the cursor will start blinking in the answer field, waiting for an answer to be entered. If a Word Clues option has been selected, it will be displayed at this time. After you enter the answer, it is checked against the correct answer. If it is not 100% correct, letter for letter, the answer is considered wrong. If the entry was the last try, or the *only* try (as in options 2 and 5), the answer undergoes a spelling check to see if the word is misspelled.

The spelling check is not 100% foolproof, but it does manage to catch minor spelling errors. The check is done by comparing each letter in your answer with the correct answer. Character position is important here. If 70% or more of the characters match, the answer is considered to be correct but misspelled. Less than a 70% match, and the answer is counted as being wrong. The comparison may look like this:

GOOD HEALTH	(Correct answer)
GOOD HAELTH	(9 out of 11— 81% — misspelled)
GOOD FOOD	(5 out of 11— 45%—wrong)

"At work or at play, the basic question and answer quiz is still a favored learning and teaching aid."

Notice that in the third answer above, *five* characters—not *four*—matched out of eleven because *all* of the characters are checked, even spaces. The alignment of the characters is important as well. If a character is simply left out, such as:

GOOD HEALTH	(Correct answer)
GOOD HALTH	(7 out of 11— 64%—wrong)

the characters to the right of the E in the word **HEALTH** would not line up correctly with the characters in the correct answer, and would *all* be considered wrong. Thus, in this example only 64% of the characters match, making the answer incorrect—not just a misspelling.

Your score is displayed at the top of the screen during the quiz. The score is actually a percentage, and not a total. The percentages are for right answers, wrong answers, and misspellings. By putting the

misspellings in a separate category, placement of the score can be left up to the administrator of the quiz. It could be added to the right or the wrong answer score, or simply used as a separate method of evaluation. This "adding of the scores" must be done by the person giving the quiz—there are no provisions in the program to have it done automatically.

Administering Quizzes

To use these programs with a disk system, initialize two disks. Place *Quiz-Make* on one disk, and *Quiz-Take* on the other. Do all of your quiz development on the disk with *Quiz-Make*. When the quiz is complete, SAVE a copy of the quiz file to the disk with *Quiz-Take*. This will give you a back-up of the quiz file. In addition, the quiz taker will not be able to change the quiz file because the *Quiz-Take* disk does not contain the program *Quiz-Make*.

If you are using a cassette system, SAVE the two programs separately on two different tapes. Also, each quiz should be kept on its own tape. Label all tapes very clearly. This will prevent you from accidentally recording over one of the programs or the quiz file.

Make the following modifications to *Quiz-Make*:


```

810 INVERSE : PRINT "LOAD DATA NORMAL
      PRINT 15 : VT " 3: HT = 17: GOSUB 156
      " : ML IF BS = A THEN 240
820 Q$ = BS : ASC (LEFT$ (FS, 1)) : INV
      FS = > FILE OR A < 64 OR ASC
      1 TO 1000 : NEXT CHR$ (7) : PRINT FOR T =
825 FOR I = 1 TO LEN (FS) : VS = MID$ (
      FS, I, 1)
830 IF AND NOT ((VS = " ") OR (VS > "0"
      AND VS < "Z")) OR (VS > "a" AND
      VS < "z")) THEN PRINT CHR$ (7) :
      FOR PRINT = 1 TO 1000 : NEXT HOME : G
      OTO 810
835 PRINT : PRINT "INSERT DISK IN DRIVE
836 1 AND PRESS A KEY" : GOSUB 1750 : DS
      CHR$ (4)
837 PRINT DS : PREFIX, D1
980 PRINT : PRINT "ENTER FILE N
      UB : 1560 : VT IF BS = " THEN 240 : GOS
      FS = > FILE OR A < 64 OR ASC (LEFT$ (FS, 1)) : INV
      1 TO 1000 : NEXT CHR$ (7) : PRINT FOR T =
995 FOR I = 1 TO LEN (FS) : VS = MID$ (
      FS, I, 1)
1000 IF AND NOT ((VS = " ") OR (VS > "0"
      AND VS < "Z")) OR (VS > "a" AND
      VS < "z")) THEN PRINT CHR$ (7) :
      FOR PRINT = 1 TO 1000 : NEXT HOME : G
      OTO 980
1005 PRINT : PRINT "INSERT DISK IN DRIVE
1006 1 AND PRESS A KEY" : GOSUB 1750 : DS
      CHR$ (4)
1007 PRINT DS : PREFIX, D1
1010 PRINT DS : "OPEN" : FS
  
```

Make the following modifications to *Quiz-Take*:

```


800 PRINT : PRINT "ENTER FILE NAME : " : M
      L = 15 : VT " 3: HT = 17: GOSUB 1020 :
      IF BS = A THEN 230
810 Q$ = BS : ASC (Q$) < 64 OR ASC
      (Q$) > 95 THEN PRINT "INVALID FILE
      NAME" : CHR$ (7) : FOR T = 1 TO 1000
      : NEXT HOME : GOTO 790
815 FS = MID$ (FS, I, 1) : VS =
      VS
820 IF AND NOT ((VS = " ") OR (VS > "0"
      AND VS < "Z")) OR (VS > "a" AND
      VS < "z")) THEN PRINT CHR$ (7) :
      FOR PRINT = 1 TO 1000 : NEXT HOME : G
      OTO 790
825 NEXT I : PRINT "INSERT DISK IN DRIVE
826 1 AND PRESS A KEY" : GOSUB 1020 : DS
      CHR$ (4)
827 PRINT DS : PREFIX, D1
830 PRINT DS : "VERIFY" : FS
840 PRINT DS : "OPEN" : FS
  
```



CONTROL CAPSULE

Quiz-Make

Edit Mode KEY	FUNCTION
CTRL Q	Select question search mode.
CTRL A	Select answer search mode.
Left CRSR	Back space to erase.
RETURN	Return to menu.
Search Mode KEY	FUNCTION
C	Change record.
N	Next record.
E	Return to menu.
Take Quiz Mode KEY	FUNCTION
Left CRSR	Back Space to erase.
RETURN	Return to menu.
Study Quiz Mode KEY	FUNCTION
Left CRSR	Scroll up.
Right	Scroll down
Esc	Return to menu.



CONTROL CAPSULE

Quiz-Make

Edit Mode KEY	FUNCTION
SHFT Q	Select question search mode.
SHFT A	Select answer search mode.
DEL	Back space to erase.
RETURN	Return to menu.
Search Mode KEY	FUNCTION
C	Change record.
N	Next record.
E	Return to menu.
Take Quiz Mode KEY	FUNCTION
SHFT X	Return to menu.
DEL	Back space to erase.
Study Quiz Mode KEY	FUNCTION
Down CRSR	Scroll down.
Up CRSR	Scroll up.
SHFT X	Return to menu.

Special Enhancement for Apple ProDOS

Under the Apple II family's new operating system, ProDOS, the system must be informed whenever you wish to access a disk with a different volume name or PREFIX. The program as published in the listing section was written to run under DOS 3.3, which does not use PREFIXes. To use the *Quiz Construction Set* programs when running under ProDOS, make the changes indicated in the following listings. [See the Apple-related "Home Computer Tech Note" in this issue for more information on ProDOS PREFIXes.—Ed.]

CONTROL CAPSULE
Quiz-Make



Edit Mode
KEY FUNCTION
F1 Select question search mode.
F2 Select answer search mode.
Back Space Back space to erase.
ENTER Return to menu.

Search Mode
KEY FUNCTION
C Change record.
N Next record.
ENTER Return to Edit mode.

Quiz-Take

Take Quiz Mode
KEY FUNCTION
ENTER Return to menu.
Back Space Back space to erase.

Study Quiz Mode
KEY FUNCTION
F1 Scroll up.
F2 Scroll down.
Esc Return to menu.

CONTROL CAPSULE
Quiz-Make



Edit Mode
KEY FUNCTION
FCTN E Select questions search mode.
FCTN X Select answers search mode.
FCTN 3 Back space to erase.
ENTER Return to menu.

Search Mode
KEY FUNCTION
C Change a record.
N Next record.
E Return to Edit mode.

Quiz-Take

Take Quiz Mode
KEY FUNCTION
FCTN 9 Return to menu.
FCTN 3 Back space to erase.

Study Quiz Mode
KEY FUNCTION
FCTN 9 Return to menu.
FCTN E Scroll up.
FCTN X Scroll down.

SAMPLE DATA BASES

THREE QUIZZES READY-TO-RUN

Here are three quizzes that you can enter into the *Quiz Construction Set* right away. Each of the first two quizzes is divided into 2 parts: Part 1 of the English vocabulary quiz places words in each answer field, and their definitions in the question fields. Part 2 features words used only in a special context, such as trade jargon or slang words—with each word given as a question, and each definition as an answer. Part 1 of the German quiz lists common German words in the question fields, and their meanings in the answer fields. In part 2, each question is a common English phrase and each answer, the German equivalent. (We suggest you use a Word Clues option for part 2 of each quiz.) A short trivia quiz is the last sample data base.

The questions and answers given here can be used with all four versions of the program. You can also add to these quizzes, use only those questions and answers you want, or create a quiz comprised of questions from all of the examples shown here.

VOCABULARY QUIZ 1

Q. harsh or discordant sound
A. cacophony

Q. decaying organic matter found on the forest floor
A. duff

Q. the space between nerve cells
A. synapse

Q. term for a fertilized egg
A. zygote

Q. a riddle
A. conundrum

Q. act of giving birth
A. parturition

Q. calm, happy, and peaceful
A. halcyon

Q. fear of foreigners
A. xenophobia

Q. a sudden outburst
A. salvo

Q. an article of food
A. viand

Q. to talk informally; chat
A. confabulate

Q. tending to melt or dissolve
A. deliquescent

Q. resembling a tree in structure
A. dendriform

Q. irritable or peevishly sensitive
A. tetchy

Q. inappropriately jocular
A. facetious

Q. having no petals
A. apetaious

Q. to cheat out of what is due
A. bilk

Q. funnel-shaped clay smoking pipe
A. chillum

Q. extremely cold
A. gelid

Q. following in time or order
A. subsequent

Q. just and fair; impartial
A. equitable

Q. social courtesies; manners
A. amenities

Q. characterized by verbal abuse
A. vituperative

Q. liberating energy
A. exergonic

VOCABULARY QUIZ 2

Q. saute'
A. fry in small amount of fat

Q. schuss
A. ski downhill at high speed

Q. allegro
A. a fast tempo in music

Q. gaffer
A. movie lighting technician

Q. plumb
A. straight up and down

Q. tweek
A. to adjust (electronics)

Q. byte
A. eight bits of data

Q. bullish
A. optimistic of boom market

Q. overdub
A. to record sound on sound

Q. codex
A. a manuscript book

Q. build-down
A. keep only new weapons

Q. totem
A. emblem or revered symbol

Q. gable
A. end wall of a building

Q. chutzpah
A. extreme self-confidence

Q. frappe
A. a partly frozen drink

Q. piquant
A. engagingly provocative

Q. pixel
A. screen picture element

Q. bug
A. a program malfunction

Q. hyperbole
A. extravagant exaggeration

Q. sprent
A. sprinkled over

Q. yarder
A. a log-pulling machine

Q. vapid
A. lacking liveliness

Q. gaggle
A. flock of geese on ground

Q. parry
A. to ward off an attack

Q. maquette
A. small preliminary model

Q. perquisite
A. extra reward or gratuity

Q. farrier
A. one that shoes horses

Q. warp
A. lengthwise strings in loom

GERMAN QUIZ 1

Q. der Koffer
A. suitcase

Q. gutaussehend
A. good-looking

Q. die Reise
A. trip

Q. nuelich
A. recently

Q. zwischen
A. between

Q. augenblichlich
A. immediately

Q. die Brieftasche
A. wallet

Q. eingebildet
A. egotistical

Q. das Fließband
A. assembly line

Q. schlafen
A. to sleep

Q. die Innenstadt
A. downtown

Q. wiederholen
A. to repeat

Q. vergebens
A. in vain

Q. die Gemeinschaftschule
A. primary school

Q. die Schreibwaren
A. stationery

Q. zeigen
A. to show

Q. das Verfahren
A. procedure

Q. furchtbar
A. horrible

Q. wunderbar
A. wonderful

Q. die Armbanduhr
A. wristwatch

Q. ungezwungen
A. casual

Q. die Verwandten
A. relatives

Q. zugeben
A. to forgive

Q. das Rasiermesser
A. razor

Q. schlank
A. slender

Q. jawohl
A. indeed

Q. verstehen
A. to understand

GERMAN QUIZ 2

Q. in the meantime
A. in der Zwischenzeit

Q. what's the matter?
A. was ist los?

Q. take care!
A. mach's gut!

Q. I am sorry
A. es tut mir leid

Q. it's now or never
A. entweder jetzt oder nie

Q. it works wonders
A. wunder wirken

Q. do you have a light?
A. haben sie Feuer?

Q. help yourself
A. sich bedienen

Q. now and then
A. hin und wieder

Q. for example
A. zum Beispiel

Q. in that case
A. in diesem Fall

Q. take it easy
A. nimm's leicht

Q. without a doubt
A. ohne Zweifel

Q. be that as it may
A. wie dem auch sei

Q. hurry up
A. mach schnell

Q. in the morning
A. am Morgen

Q. how are you?
A. wie geht's?

Q. good day!
A. guten Tag!

Q. we have a lot in common
A. sie steht mir nahe

Q. you're welcome
A. bitte sehr

Q. show me
A. zeigen sie mir

Q. what does that come to?
A. wieviel macht das?

Q. that turns me on
A. das begeistert mich

Q. a big shot
A. ein hohes Tier

Q. time is up
A. die Zeit ist um

Q. see you later!
A. Aufwiedersehen!

Q. Name the first computer to use a mouse and icons.
A. Xerox Star

Q. Who shot James A. Garfield?
A. Charles Guiteau

Q. How many typographic points to the inch?
A. 72

Q. Which particle has both light and matter properties?
A. neutrino

Q. Which famous cowboy movie star carried a whip?
A. Lash La Rue

Q. What is the world's highest-flying jet aircraft?
A. SR-71 Blackbird

Q. Who was Fred Flintstone's best friend?
A. Barney Rubble

Q. Where was the first semiconductor produced?
A. Bell Labs

Q. Who is the father of the Pascal language?
A. Nicholas Wirth

Q. What is Ringo's other name on the Sgt. Pepper album?
A. Billy Shears

Q. Who said, "I have not yet begun to fight"?
A. John Paul Jones

Q. In which film did Chaplin satirize Adolf Hitler?
A. The Great Dictator

Q. Who was the "Man of a Thousand Faces"?
A. Lon Chaney Sr.

Q. What was the name of the dog in "Topper"?
A. Nell

Q. What is the name of Ricky's brother in "Ozzie and Harriet"?
A. David

Q. What is the largest self-supporting concrete roof?
A. The Seattle Kingdome

Q. Who was the founder of the Republic of China?
A. Sun Yat-sen

Q. Which computer magazine has no outside advertising?
A. Home Computer Magazine

Q. Who said, "I will fight no more forever"?
A. Chief Joseph

Q. What is the name of the spice in "Dune"?
A. melange

Q. Who hosted "You Are There" in the 1950's?
A. Walter Cronkite

Q. What were Romeo and Juliet's last names?
A. Montague and Capulet

Q. Which President was raised as a Quaker?
A. Richard Nixon

Q. What was Priam's prize for judging a beauty contest?
A. Helen

Q. Which early radio show restaged movie hits?
A. Lux Radio Theater

Q. What was the dry planet in "The Dispossessed"?
A. Anarres

Q. Who was the housekeeper in "My Three Sons"?
A. Bub

Q. What substance powers the Starship Enterprise?
A. dilithium crystals

TRIVIA QUIZ

Q. Who was the fifth Marx brother?
A. Gummo

Q. What's the flip side of the Beatles' single, "Rain"?
A. Paperback Writer

Q. To what religious sect do we owe the circular saw?
A. Shakers

Q. Which was the 1st major car with front wheel drive?
A. The Cord

Q. Who is the robot in "The Day the Earth Stood Still"?
A. Tobar

Q. Who are the people of "The Forbidden Planet"?
A. The Krell

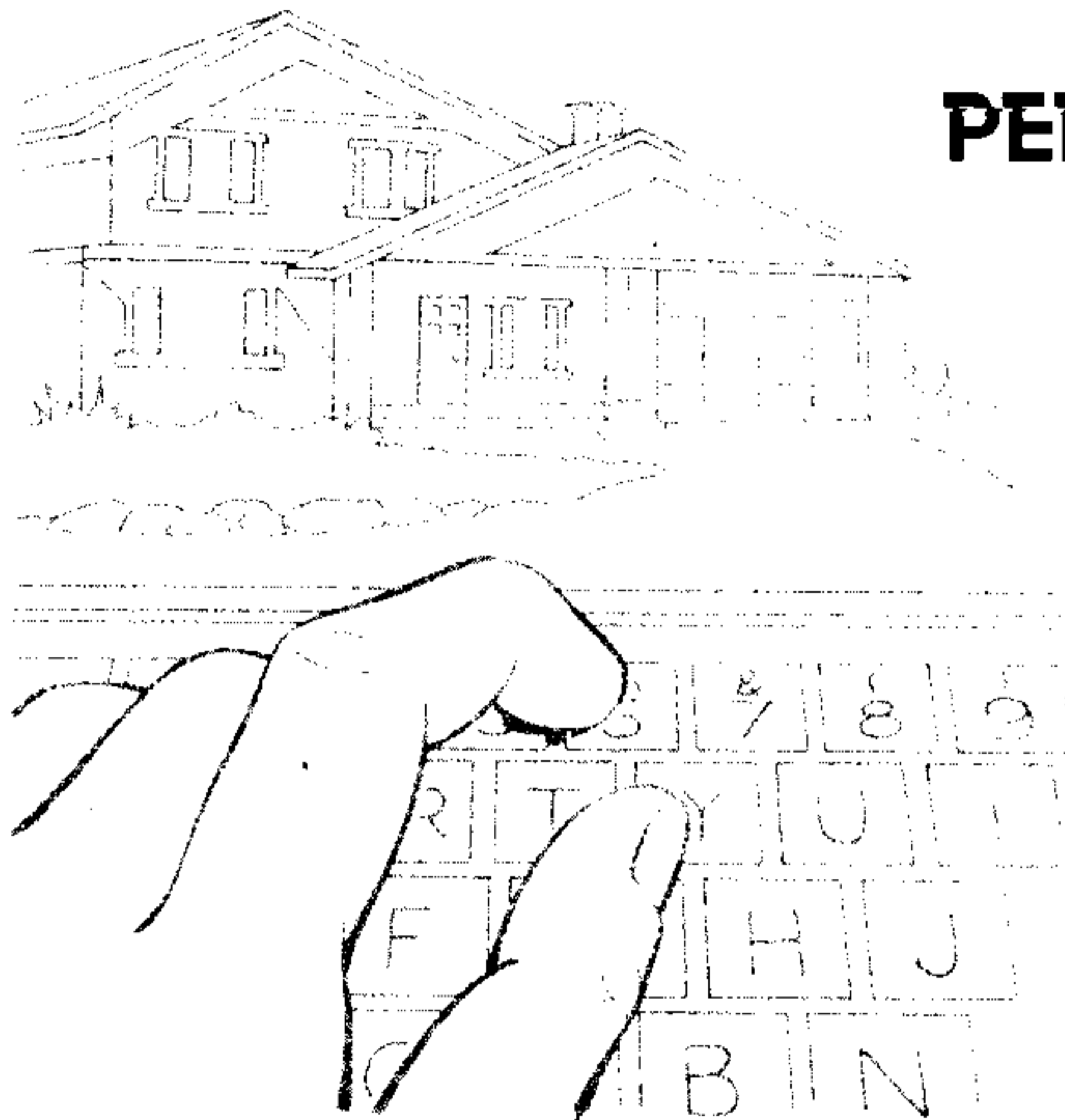
Q. What TV show featured Cochise?
A. Broken Arrow

Quiz-Make (Apple II Family) Explanation of the Program

Line Nos.	
100-180	Program header.
190	Define error-trapping routines location.
200-230	Title screen.
240-290	Main menu.
300-800	Edit quiz and search records.
810-940	Load quiz file.
950-1100	Save quiz file.
1110-1320	Print quiz file.
1330-1470	Input-parameters routine.
1480-1540	Exit-program routine.
1550	Illegal entry message.
1560-1740	Main-input routine.
1750-1760	Single-key-input routine.
1770-1810	Error-trapping routine.
1820-1830	Program data.

Quiz-Make (C-64) Explanation of the Program

Line Nos.	
100-170	Program header.
180-250	Display title screen.
260-370	Main menu.
380-750	Edit the quiz.
760-1130	Search mode.
1140-1450	Load the quiz file.
1460-1780	Save the quiz file.
1790-2200	Print the quiz file.
2210-2510	Change-parameters routine.
2520-2630	Input routine.
2640-2720	Illegal entry messages.
2730-2870	Input-a-question routine.
2880-3020	Input-an-answer routine.
3030-3040	Clear parts of the edit screen.
3050-3060	Routine to locate the cursor.
3070-3170	Exit-program routine.



PERSONAL LOAN CALCULATOR

by H. W. Button

and the HCM Staff

Borrowing money?

Here's a program to give you all the information you need to make that loan pay off.

Personal Loan Calculator is a handy companion to the *Savings* program published in Vol. 4, No. 1 for the Apple, Commodore and IBM computers, and in Vol. 2, No. 6 for the TI-99/4A. The two programs together form a comprehensive software package for everyday personal financial decisions.

Borrowing money can be terrifying. A long-term loan, such as a home mortgage, may be considered a brave act—and certainly sometimes a confusing one.

Where Your Interest Lies . . .

Many inexperienced borrowers are surprised to find that for the first few months, or even years, of their payment schedule, they will be paying primarily the *interest* on their loan. Only after most of this interest is covered will they begin to pay off a significant portion of the *principal*—that is, the actual amount of money borrowed. With each payment, the proportion of interest to principal changes: the percentage of the payment going to interest *decreases*, and the percentage going to the principal *increases*.

Many questions arise when someone considers either an existing or a prospective loan: How much will I end up paying over the entire term of the loan? How many payments will I have to make? How big will the payments be? How big a loan can I afford? At any given time, how much will I be paying to interest, and to principal? And how much principal will I have left to pay? The *Loan Calculator* is designed to answer these questions with the use of your home computer.

Displayed on the menu of *Loan Calculator* are five options:

1. Payment amount
2. Number of payments
3. Loan amount
(How much you can afford to borrow.)
4. Amortization Schedule
5. Exit the program

You may select any one of these options, in any order. In options 1, 2, or 3, you are first asked two questions: The first inquires whether your payments will be made monthly or annually. The second asks whether the length of the loan period should be expressed in months or years. This input will affect how often the interest is compounded. Compounding the interest will alter the amount of interest due for each payment,

according to the amount of principal still owed on the loan. For example, an annual interest rate of 12% will compound monthly at 1%. If you had \$10,000.00 in principal left to repay, the interest for one month would be 1% of \$10,000, or \$100.

In Option 1, you also will be asked the following:

- Interest rate?
- Months (years) of loan?
- Amount of loan?

In Option 2 you will be asked:

- Interest rate?
- Monthly (annual) payment?
- Amount of loan?

If the payment amount you enter in Option 2 is too low to cover the interest generated during each payment period, you will receive an error message asking you to enter new data (either raise your payment or lower the interest rate or loan amount).

In Option 3 you will be asked:

- Interest rate?
- Monthly (annual) payment?
- Months (years) of loan?

“Many inexperienced borrowers are surprised to find that for the first few months, or even years, of their payment schedule, they will be paying primarily the interest on their loan.”

Using any of these first three options will generate a report which consists of the following new information, plus the data you have already entered:

Interest rate	Compounded (monthly or annually)
Loan amount	Payment amount
Number of payments	Term of the loan
Total cost (principal + interest)	Total interest

In this report (see photo), the program will supply the data you have *not* entered, but want to know.

Finally, in Option 4 (Amortization Schedule), you will be asked the following:

- Loan amount?
- Number of monthly payments?
- Interest rate?

After you enter this data, the following information will be printed:

- Monthly payment
- Final payment

You will then be prompted for the following:

- Show schedule from payment #?
- To payment #?

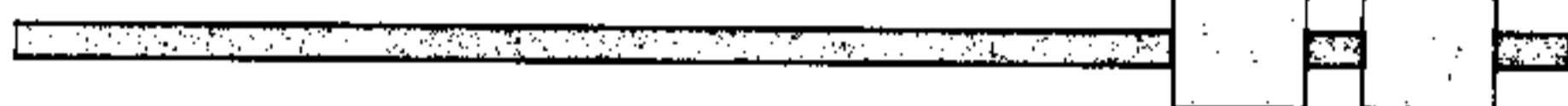
Here you enter the starting and ending payments that you want included in the report. The report for all of these months will then be displayed, one month at a time. This report will include:

- Payment #
- Interest for this payment
- Principal for this payment
- Loan balance for this payment

You may scroll through the payment schedule month by month, and return to the Main Menu after viewing the last monthly report.

This *Loan Calculator* program is a very handy, flexible tool for quickly generating information vital to anyone who is either considering or already paying off a loan. If you are in one of these positions, this program is "just what the banker ordered."

For your key-in listing see HCM PROGRAM LISTINGS Contents on page 85.



[NOTE: When using *Loan Calculator* on the IBM PC, make sure to start the BASIC language by typing in BASICA/D to enable Double Precision mode—Ed.]

The IBM PC and PCjr have a very powerful function built into their BASIC PRINT statement which allows display formatting. The USING option of the PRINT statement allows a programmer to specify certain formatting parameters for the display. The most common type of formatting is done with numeric values. Three format types are used in this program, as follows:

```
F$      $$###,###,###,###.##
IST$    #####.##### %
F2$     #####.#####
```

The first format, F\$, is used to display all monetary values. The double dollar sign (\$\$) indicates that a floating dollar sign will be displayed to the left of a number, and each pound sign represents a digit. If there aren't enough digits to fill all of the pound signs, they will be replaced with spaces. The commas will be printed only if there are enough digits in the number to fill the pound signs to the left of the comma. The two pound signs to the right of the decimal point indicate that the value will be rounded off to two decimal places. If the value is an integer, or has only one decimal place, the pound signs to the right of the decimal point will be replaced with zeros (0). The number 2483.1 would be displayed as: \$2,483.10.

The second format shown above is used to display the interest rate. The third format is used for other miscellaneous numeric displays.

As shown in this IBM version of the report screen, the program will display the information you entered, plus the missing pieces.

LOAN REPORT	
INTEREST RATE IS	28.000000 %
COMPOUNDED MONTHLY	
LOAN AMOUNT	\$3,500.00
MONTHLY PAYMENT	\$324.22
NO. OF PAYMENTS	12.0000
TERM IN MONTHS	12.0000
TOTAL COST	\$3,890.65
TOTAL INTEREST	\$390.65
PRESS ← TO RETURN TO THE MENU	



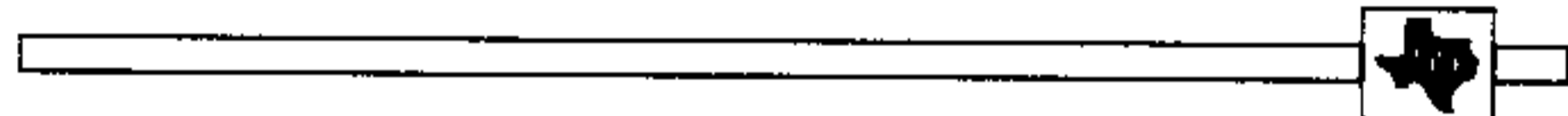
The Commodore 64 computer has several commands that can be used to read the keyboard. For instance, you can read an entire line of input with the INPUT statement, or read single characters with the GET statement.

In *Loan Calculator*, a user is required to enter a variety of numeric values which will be used in complicated mathematical formulas. If we use the INPUT statement for this purpose, it would be possible for the user to enter a number up to 1.70141183E + 38 (the largest number that can be handled by the C-64 BASIC system, and much too large for our calculations). And, if a user enters alphabetic characters instead of numbers, a lot of limit checking would be required after an entry is made. Any errors would mean that the entire entry would have to be done over again.

However, there is a solution—the GET statement can read one character at a time. If we include a limit check on each character as it is typed, we can eliminate any illegal characters before they can be added to the user's input. This is done in a subroutine (lines 1700-2030) that is called every time an input is required. Thus, the program can pass several variables to the routine to prevent illegal entries. Below is a list of the variables that are passed to the routine. This routine may come in handy in some of your own programming ventures.

Variable Function

- LN Screen line where input will appear
- HV High value limit check
- LV Low value limit check



[Special Note: Due to space limitations, we are publishing only an Extended BASIC version of *Loan Calculator* in this issue of HCM. Extended BASIC provides for a more elegant solution to many problems, due to its superior ability to handle error-checking, and its advanced input and display functions. Recognizing, however, that *Loan Calculator* is a highly useful program, and that many TI-99/4A owners do not have Extended BASIC, we have also written a TI BASIC version that can be found along with the Extended BASIC version on this issue's (Vol. 4, No. 5) TI edition of ON TAPE or ON DISK. See back cover and order card for information.]

This Extended BASIC version of *Loan Calculator* takes advantage of the **USING** option of the **DISPLAY AT** statement to format the values displayed on the report screen. To indicate the format to be used with the **USING** option, you must specify the line number that contains the **IMAGE** statement. The **IMAGE** statement must be the only statement on that line; it contains a string of characters which indicate the output format. The three formats used in this program are on line numbers 1840, 1850, and 1860. To use the format in line 1840, place **USING 1840** in the **DISPLAY AT** statement as is done in line 1440.

The first two statements in the program will trap any errors that may occur in the program. There are two levels to the error-trapping. If the error is serious enough, it will be trapped by the **ON ERROR** statement, and the program will branch to an error-handling routine which starts in line 1870. The error routine will flash a message on the screen telling the user that there was an error in the calculation, meaning that the program can't finish it. The program will then return to the Main Menu screen. Minor errors will simply be passed over because of the **ON WARNING NEXT** statement.



Because the Apple computer has no formal way to format numeric data displayed on the screen, it is necessary to do it manually within the program by employing a rounding algorithm.

For example, when you display dollars and cents, you obviously don't want the value to contain more than two digits to the right of the decimal point—i.e., \$123.456 isn't correct. The value \$123.46 is what you are seeking. It is possible to achieve this result with a simple rounding algorithm. If the value you want to round off is contained in the variable **A**, then the following line will round it to the nearest cent:

$$A = \text{INT}(100 * A + .5) / 100$$

This works by first multiplying **A** by 100: 123.456 would become 12345.6 in value. The program would then add .5 to that value, giving us 12346.1 as an intermediate result. The **INT** function would then chop off the .1, returning only the integer part of the number 12346. Finally, this result is divided by 100, giving us 123.46 as the final displayed result. If you desire a result that is not rounded, but is still limited to only two digits past the decimal point, then omit the **+ .5** from the equation.

Loan Calculator (Apple II Family) Explanation of the Program

Line Nos.	
100-170	Program header.
180-230	Main menu.
240	Exit program.
250-290	Enter frequency of payments, and the expression for the length of the loan period.
300-490	Option 1—Payment amount.
500-660	Option 3—Amount of loan.
670-810	Option 2—Number of payments.
820-930	Final report for options 1, 2, and 3.
940-1160	Option 4—Amortization Schedule.
1170-1220	Routine to format the display output.

Loan Calculator (TI-99/4A) Explanation of the Program

Line Nos.	
100-170	Program header.
180-200	Title screen.
210-280	Main Menu screen.
290-370	Enter values for frequency of payments, and the expression used for the length of the loan period.
380-690	Option 1—Amount of payment.
700-1020	Option 2—Total loan amount.
1030-1380	Option 3—Number of payments.
1390-1530	Display the final report for options 1, 2, and 3.
1540	End the program.
1550-1830	Option 4—calculate and display the Amortization Schedule.
1840-1860	IMAGE formats for the USING option.
1870-1880	Error routine.

Loan Calculator (IBM PC, PCjr) Explanation of the Program

Line Nos.	
100-190	Program header.
200-230	Initialize program.
240-290	Main Menu screen.
300-350	Enter frequency of payments, and the expression of the term of the loan.
360-440	Subroutines for all inputs.
450-540	Option 1—Amount of payments.
550-640	Option 2—Number of payments.
650-740	Option 3—Amount of loan.
750-870	Final report for options 1, 2, and 3.
880-1090	Option 4—Amortization Schedule.
1100-1210	Utility subroutines.
1220-1230	Error-handling routine.
1240-1250	(Esc)-key routine.
1260-1270	Exit-program routine.

Loan Calculator (C-64) Explanation of the Program

Line Nos.	
100-180	Program header.
190-300	Main Menu.
310-420	Get inputs for length of loan period and term expression.
430-630	Option 1—Routine for amount of payment.
640-860	Option 2—Routine for total loan amount.
870-1110	Option 3—Routine for number of payments.
1120-1360	Display final report for options 1, 2, and 3.
1370-1850	Option 4—Routine to calculate and display the Amortization Schedule.
1860	Exit program.
1870-1890	Error-in-calculations routine.
1900-2030	Input routine. Checks that all entries are numeric, and within the specified value ranges.

HCM

PEGS
LEFT
32

PEGS
REMOVED
0

Jumping Ahead

With Game Programming

FROM: _____

TO: _____

Program
by **Bob Stoffers**
and the HCM Staff

With Tutorial Text
by **William K. Balthrop**
HCM Staff

*There's more to home computing than just playing games—
You can create the games as well. Learn how by following along
as we analyze this video version of the classic peg-jumping puzzle.*

[**Note:** In Vol. 4, No. 3 of *HCM*, an article entitled, "Programming: The Name of the Game" presented some helpful tips on game programming, looking at specific coding and trouble-shooting techniques. In the article below, we look at the structure of one particular game, and how it reflects important principles of game design.]

Have you ever wanted to produce your own video games? Perhaps—even though you may be so-inspired—you have shied away from the actual task of programming an original game, and you've been content to play whatever comes along. Still, some of the most fun you can have with your home computer is not just *playing* games, but *creating* them.

We would like to encourage you to try your own hand in game programming by illustrating how a game is first designed and then transformed to program code. In the process—and as an example—we will introduce *Peg Jump*, a board game that will challenge your puzzle-solving skills for many hours.

Getting Started

Sometimes, the hardest part of writing a game program is coming up with an idea for it. If this is a problem for you, try sitting back in a comfortable chair with your eyes closed, and daydream a little. Think about what you would like your computer to be able to do. At this point, don't worry about how such

a program might work, think only about what it would do. Thinking about details at this time will probably cloud the imaginative process.

One important point, however—don't try to simply copy someone else's program or idea. This exhibits the lowest level of imagination, and, if carried too far, may infringe on copyright laws. Other programmers' ideas can show you what is *possible*, inspiring you to create your *own* original (and practical) ideas.

In *Peg Jump* the idea was to make a game which simulates a puzzle that many of you may have played as children. The puzzle is known by several names, and consists of a playing board in the shape of a cross, with 33 holes in it. The game starts with a peg in every hole except the center. The object is to jump one peg at a time over another peg, removing each peg jumped as you go. To win, you must finish the game with just one peg left on the board, in the center hole.

Most computer games fall into one of the following categories: Board game, Puzzle, Arcade, Adventure, or Education.

Any game may actually cross over into two or more of these categories. *Peg Jump* is a combination board game and puzzle. It's a board game because the action takes place on a playing surface which could just as easily be a game board on your kitchen table. For example, Chess and Backgammon—two computer favorites—are both board games. *Peg Jump* is also a puzzle because it challenges the player to solve a problem through a series of moves—in a sense, playing *against* the board.



Preliminary Design

At this stage you should already know what you want your program to do. Now you need to sit down and create a general outline of the events that will take place in the program. This will help you organize your thoughts, and result in a more complete design on the first pass.

Your general outline may take the shape of this example used for *Peg Jump* :

1. Initialize the game.
2. Draw a board on which to play the game.
3. Let the player enter a move.
4. Check for legal moves.
5. Move the peg from one hole to the next; delete the peg that was jumped.
6. Check for an end-of-game situation.
7. Have the puzzle solve itself upon request.

If you work from a list like this, and tackle one problem at a time, you will usually end up with a much cleaner program.

Another preliminary step—creating a flow chart—is optional, but recommended. Often, programmers create a flow chart of their program before they write its code. A flow chart is a series of boxes connected by lines; each box contains a function or decision that the program performs. The lines connecting the boxes show the “flow” of the program. By creating such a chart you can figure out what kind of routines the program will need.

Other programmers find a flow chart too restrictive and prefer to work from general outlines (like the one above), structured charts, or even scraps of paper. The important thing is that you, the programmer, have a clear idea of all aspects of the programming task, and a carefully-laid plan of attack to accomplish it before you start coding.

Starting to Write the Code

Now you're ready to sit down at your computer and start banging away at the keyboard. Don't forget to make a large pot of hot coffee, and say goodbye to your family and friends. Then GOTO it!

Writing the code isn't as scary as it may seem at first. A good place to start is with a title screen. You can even get fancy if you want, making large letters or adding graphics to spruce it up. Keep one point in mind here—do not leave players hanging without letting them know what to do. An example of this would be a program which displays a title screen, and then sits there waiting for the player to press a key to continue—without any message on the screen. This may seem trivial, but you would be surprised at how many programs don't give adequate screen instructions. On the other hand, this shouldn't mean 40 pages of screen text either. Keep all screen messages as short and informative as possible.

Initialize The Program

After you have designed the title screen, you should implement the initialization phase of the program. This is the part of the routine that sets up everything for the rest of the game. It also should ensure that the program is going to run in the proper environment (memory size, graphics options, etc.).

Ask yourself a few questions before starting this section. Will you need to use an array, and if so, how big does it need to be? Do you need to create any

graphics patterns, or assign values to key variables? You may need to come back and add to this section from time to time as the program develops, and you get a better feel for what is needed.

If the program must go through a long initialization with nothing happening on screen, you should make users aware of the delay; otherwise they may think the program or computer isn't working properly. This is true on the Texas Instruments, Commodore, and Apple versions of *Peg Jump*—they all go through a fairly lengthy initialization, and so display the message PLEASE STAND BY.

“Ideally, a game should be so easy to operate that a player can begin without even reading the instructions—no matter how complex the game actually may be.”

Draw a Game Board

This part of the program sets up the playing screen for the rest of the game. Board games are generally the easiest type of screens to set up because most of them are static (remain the same). Of primary importance to the enjoyment of your program is appearance. An attractive and exciting screen will keep people interested in the game longer than if the screen were dull and boring. This is especially true for board games and puzzles, where the action is usually very slow.

Before you start placing graphics randomly on the screen, you should first sketch them out on a piece of graph paper with a pencil and a good eraser. You may save yourself hours of programming headaches by first proving that a screen design works on paper. You can then use the graph-paper grids to help calculate screen coordinates. This is really an invaluable step in the design process, without which you may spend countless hours designing and re-designing your display, only to finally end up with a lackluster screen.

In *Peg Jump*, we need to place an imaginary board in the shape of a cross on the screen. We must somehow illustrate the 33 holes in the board, and indicate which holes have pegs in them. The actual method used varies from system to system, so we won't go into great detail on the written code. It is easy to draw the board on a piece of paper, and then locate where the holes should line up. From this, it is possible to calculate where on the screen each hole is located. Once this is known, the coordinates can be stored in an array. To find the location of any given hole, you simply need to index into the array, and locate the hole's screen coordinates.

But how will the player know which holes are which? Remember our discussion about giving the player adequate information? If a player had to sit there and count all the holes to find the desired hole number in order to make a move, the game would soon become more of a headache than a pleasure. The best way to handle the situation is to number each hole with its index value in the array that contains the hole's location. This will also simplify the program code. We use small numbers next to each hole so that we don't clutter the board with large, awkward figures.

Knowing the Score

The final touch is a display of the player's score. In this case the display indicates how many pegs are left on the board, and how many pegs have been removed. The score display should be placed on the screen where the player can glance at it without losing concentration; it should be available, but not a distraction. Depending on the situation, you may choose not to display the score until the end of the game, although this technique is not used very often. Most people, when playing a game, are inspired by trying to increase their score. If they can't see their score during the game, they have no incentive to try to improve.

Player Interaction

This part of a program is the interface between the player and the computer. It is very critical, and should be given close attention. If a player must make awkward moves, if the commands are difficult to remember, or if the keys to control the game don't make any sense, the game loses a lot of appeal—and in arcade-type games, a lot of realism. Ideally, a game should be so easy to operate that a player can begin without even reading the instructions—no matter how complex the game actually may be. If necessary, provide brief screen instructions to indicate what type of action is required of the player. If the computer is waiting for the player to enter a number, as in *Peg Jump*, let the player know what you want. In *Peg Jump*, the message FROM: lets the player know that the computer is waiting for a number indicating which peg is to be moved. The message TO: will then let the player know that the computer is waiting for a number to indicate where to move the peg. Notice that these messages are very short, but that they get the point across. You don't need to make the player read a book of instructions when a very short message can accomplish the same task.

When the user enters responses at the keyboard, you need to make it clear what the entry is to be for, and then check that the entry is valid. (This is not yet part of the legal game move check.) If a user is to enter a number, what happens when a letter or (ENTER) is pressed? The program must catch this sort of thing, and either allow the player to re-enter, or give the player some warning. This kind of a check is called "exception testing." It is one of the most overlooked aspects in programming. Often, when testing the program, the programmer knows what the correct type of entry should be, and never thinks about what would happen if . . .

There are two special values in *Peg Jump* which you can enter at the FROM: prompt. If you enter 88, the program will start the Auto Solve mode and actually show you how to solve the puzzle. The second special input is 99, which will reset the board to the beginning of a game. Use this option when you realize that there is no hope of completing the puzzle.

Check for Legal Moves

This section extends beyond simply verifying that input is of the right type. The player should not be able to do anything that would violate the rules of the game. If you rely on a player's honesty to keep from making illegal moves, a player could still make an unrecoverable mistake. Before writing this section of

the code, you should devise the rules of the game. As an example, the rules for *Peg Jump* are listed here:

1. You must move from a hole containing a peg.
2. You must move to a hole which is empty.
3. All moves must jump one—and only one—hole.
4. The hole jumped must contain a peg.
5. The peg jumped must be removed after the jump.
6. Pegs can't jump diagonally.

Each of these cases should then be checked. If the check fails, the program returns to get a legal input. When designing your own games, you may wish to make the player lose a turn, or lose points for attempting to make an illegal move. Then the outcome of the game will depend to some extent on how well the player follows the rules.

Move the Peg

This section of the program takes a player's input, and changes the game situation accordingly. In this case, the computer must move a peg from one hole to another, and remove the peg jumped. It is difficult to pin this section down to DO's and DON'T's in general terms, because there are a large number of ways this situation could be handled.

If there is to be any action—or reaction—to a player's input, it should be handled in a way that is agreeable to the player. Taking *Peg Jump* as the example again: If all we did was erase the peg from one hole, draw it in another, and then erase the peg jumped, the real action of the program would be missed. The player has no concept of a peg jumping from one location to another. You should try to convey to the player what is taking place, not just display the results. Here, we were able to accomplish this with a little "low level" animation. By taking the peg to be moved, and actually showing it sliding across the board to its new location, the player gets a real feel for what is happening. In almost all games, the extra effort to produce realistic effects is worthwhile.

End-of-Game Check

All game programs should make a check at some point in the game to see if it is over. While the player may be aware that the game has come to a halt, the program must also be aware. If the game is over, or the end of one part in a multi-part game is reached, you must decide what to do. You should display the score, and possibly any other game statistics that may be interesting, such as the high score.

After the game is over, you should give the player a chance to play again. No program should just dump the player back to the system without asking permission—it's just not polite. If the player chooses to play the game again, there are a number of things you should keep in mind. First of all, you will probably need to re-initialize some of the variables, in particular the score. In doing this, try not to re-initialize more than is necessary. You shouldn't have to go all the way back to the first line of the program. In *Peg Jump* the location of each hole on the screen is stored in an array. This array doesn't need to be reloaded for each game, because it stays the same. This saves considerable time in the re-initialization phase of the program.

The second thing you should watch for is the re-entry of skill levels, or the like. There are no skill levels in *Peg Jump*, but many programs do contain them. If a player chooses to play again, you should decide whether he or she is allowed to choose a new level.

Auto-Solve Mode—A Bonus

One program feature usually found only in (but not limited to) puzzle-type games is an auto-solve mode. It enables the computer to solve and prove that an answer to the puzzle is possible. This type of feature is considered a bonus, and is not required to play the game. The level of intelligence you give the program depends solely on how much you want to dazzle the person using the game. In *Peg Jump*, we chose the simplest form of auto solve. No real intelligence is involved, so the computer doesn't need to actually find a solution—it has all of the moves required to solve the puzzle stored in DATA statements. Then it simply READS the next move, and uses the same routines that the player uses to move a peg.

**"No program should just dump
the player back to the system
without asking permission—
it's just not polite."**

The Final Phase

After every program is written it should go through rigorous *play-testing*. This part of a program's development can't be overemphasized.

Because it is often difficult to be completely objective with your own program, you should have several other people unfamiliar with the program take a whack at it. It's even better if these people are unfamiliar with computers. Have them document anything unusual they might encounter. Many times you will write a program, and because you already know the program's or system's limitations, you will only design the program to the point where it does everything right. This is not enough. The program must also watch for anything the user might do wrong. This may be as simple as not letting the user type a letter when a number is expected, or making sure illegal control keys don't upset the game.

Try everything you can think of to make the program *bomb* (stop running prematurely). If the program survives the experience, you're getting closer. Until every line of code has been run with every possible combination of values, the program may still contain hidden bugs.

Have Fun

You can take a certain amount of pride in writing your own game programs. It really pays off when you see people playing one of your games and they say, "Wow, this game is great. Where did you get it?" You can just flash them a broad smile and say "I wrote it myself."

For your Key-in listing see HCM PROGRAM LISTINGS Contents on page 85.

CONTROL CAPSULE *Peg Jump*

VALUE INPUT	FUNCTION
1 - 33	Move a peg from one hole to another.
88	Auto-Solve mode.
99	Reset the game.



The Apple II program uses shape tables to generate graphics shapes on the screen in BASIC. These shape tables were used to generate small numbers from 0 to 9 for the identification of holes on the board. The data for these shapes is contained in lines 1210 through 1410. Using these shapes in place of the standard characters offers several advantages: (1) They can be positioned anywhere on the screen. (2) They are not limited to the boundaries of the normal character. (3) They can be placed on the graphics screen. (4) They can be rotated or expanded using the ROT=, or the SCALE= functions.

The routine that uses these shapes resides in lines 1080 through 1160. It could be easily modified for use in any other application. Using these shapes would allow you to display up to 80 digits per line on a graphics screen.



Because of the Commodore's ability to print graphics characters within the quotes of a string, we decided to simply PRINT the board on the screen—rather than place it on the screen one character at a time. This is done in lines 480 through 700.

The title screen for this version was spruced up using the same method. Graphics characters were formatted within strings, and then PRINTed to produce enlarged text characters.

A FOR_NEXT loop is used to move a peg. This routine, in lines 980 through 1030, will move the peg across the board. Each time the peg advances one character position, the character below it is saved, so that after the peg leaves that spot, the original character can be returned. In doing this, the board's graphics are undisturbed by a peg moving across.



The IBM systems use quite a different method to create the playing board. If your PC system has the Color/Graphics Adapter and color monitor, or a compatible color system, you will be able to use this program, and the commands described below. If you have the PCjr, you will need only Cartridge BASIC.

The DRAW statement allows the programmer to draw lines and fill areas of the screen with color. This makes graphics creation extremely easy. For this reason we made the IBM board a little more elaborate by making it appear three-dimensional. It was a simple matter of using the DRAW statement to add sides to the board, giving it depth.

Animating the pegs posed a problem because the pegs interfered with the lines that connected each of the holes. Using the PUT statement to move a peg disrupted these lines if the peg started or stopped on a line. However, slanting the pegs got them away from the lines, and as an additional "bonus," remains consistent with the board's three-dimensional effect.

You may want to extract the small-numbers routine for your own use. These numbers are used to identify the holes. Each number is only four pixels wide. This would give you 80 digits across the screen on a 40-column screen, and 160 digits across on an

80-column screen. Each number is drawn with the DRAW statement. The commands to control the drawing are kept in a string array. There are ten elements, one for each number 0 through 9. This routine could be used in almost any program, with slight modifications. All you would need to do is pass the value to be drawn in the variable Z. The PSET command would then have to be modified to output the characters at the proper location on the screen. Currently, it uses an array of screen coordinates for the holes in the board.



The TI-99/4A version of *Peg Jump* makes use of character graphics to display the playing board. The program was written in BASIC.

To identify the holes, it was necessary to create small number labels to conserve space on the board, so that it will not look cluttered. These numbers are coded into ASCII characters 100 through 132. The FOR NEXT loop in lines 540 to 570 reads the shapes for the numbers and assigns them to the characters. The loop in lines 950 to 970 then places these numbers on the screen in their appropriate locations.

To create the animation of moving the peg from one hole to another, two short routines were used. Lines 1660 through 1790 contain the two routines that will move the peg in any of the legal directions. Any characters that the peg passes over will be replaced, leaving the board's graphics unchanged.

Peg Jump (TI-99/4A) (TI BASIC) **Explanation of the Program**

Line Nos.	
100-180	Program header.
190-330	Initialize variables.
340-360	Data for game messages.
370-440	Display the title screen.
450-670	Initialize playing screen graphics.
680-1130	Display the playing screen.
1140-1230	Display current peg status.
1240-1470	Input the move from and the move to. Check preliminary limits.
1480-1660	Check for illegal moves.
1670-1820	Move the peg.
1830-1850	Update score.
1860-2090	Display "getting close" messages.
2100-2360	Display illegal move message.
2370-2380	Read next move when in Auto Solve mode.
2390-2400	Computer's victory message, after completion of the Auto Solve mode.
2410-2500	Music routine.
2510-2540	Routine to display text without scrolling the screen.
2550-2680	Input routine.
2690-2710	Data for peg locations.
2720-2820	Data for graphics characters.
2830-2840	Data for graphics positioning.
2850	Data for peg status display.
2860-2870	Data for Auto Solve mode.

Peg Jump (C-64) **Explanation of the Program**

Line Nos.	
100-180	Program header.
190-370	Display the title screen.
380-410	Store screen messages.
420-450	Initialize the game variables.
460-730	Display the playing screen.
740-920	Get move, and check to see that it's legal.
930-1090	Move pegs.
1100-1270	Routine to display messages.
1280-1380	Illegal move routine.
1390-1470	Clear message areas.
1480-1700	Music routine.
1710-1810	Input routine.
1820-1830	Cursor placement routine.
1840-2030	Data for the music routine.
2040-2090	Data for messages.
2100-2150	Data for peg locations.
2160-2190	Data for Auto Solve mode.

Peg Jump (IBM PC, PCjr) **Explanation of the Program**

Line Nos.	
100-200	Program header.
210-220	Display the title screen.
230-270	Initialize the program.
280-360	Draw the playing board.
370-440	Input moves and check for illegal moves.
450-480	Jump the peg vertically.
490-520	Jump the peg horizontally.
530	Display the peg status.
540-550	Subroutine to draw small numbers.
560-590	Key input routine.
600-650	Display the "getting close" messages.
660-680	Read next move for Auto Solve mode.
690-760	Display illegal jump messages.
770-790	Data for peg positions on the screen.
800-810	Data for the DRAW statement to create the small numbers.
820-830	Data for the "getting close" messages.
840-850	Data for the Auto Solve mode.

Peg Jump (Apple II Family) **Explanation of the Program**

Line Nos.	
100-180	Program header.
190-380	Initialize the program variables and arrays.
390-400	Branch to POKE in the music routine and the shape tables.
410-420	Branch to set up the playing screen.
430-580	Input the FROM peg and the TO peg. Do preliminary limit checks.
590-810	Check for illegal moves.
820-930	Move the pegs.
940-1170	Auto Solve mode.
1180-1440	POKE shape tables into memory.
1450-1480	Call sound routine.
1490	POKE in sound routine.
1500-1550	Display "getting close" messages.
1560	Sound effects routine.

HCM

In the "Letters to the Editor" section of the previous issue, we put out the call for one-line programs written in any language on any of the computers that are covered in *Home Computer Magazine*. The response was so great that we decided to devote an entire page to one-liners. Although many interesting programs were submitted, we have selected what we felt were the four best (one for each brand of computer) of those that arrived prior to this issue's press date. If you have not submitted your masterpiece, it is not too late! As long as we keep getting great one-liners, we'll keep filling this page for you. Our prize winner this issue is G. A. Hamilton, who won \$50 for sharing a unique one-line "arcade game" with our readers.

One-Line Arcade Game

[TI Extended BASIC on the TI-99/4A]
Dear Sir:

ALPHABET AT-TACK

You (the @ symbol) are being attacked by alpha characters. Points are scored against you for hits on your @ ship. Low score wins. The longer you take, the faster the alphas go. The program includes 24 various colored alphabet sprites that move in different speeds and directions. One sprite (@) is controlled by the No. 1 joystick. The score is continuously presented on the

screen. Type until the computer beeps and press (ENTER). Type 1, then (FCTN) E, then (ENTER), then (FCTN) 8 and finish typing the line.

G. A. Hamilton
Nepean, Ontario, Canada

```
1 N=28:FOR X=4 TO N:GOTO 2
2 ALL SPRITE(#X,60+X,X/2,
3 LN,X,M):FOR Y,#4,N+M,C
4 CALL COINC(#Y,#4,N+M,C
5 ):M=M-C:DISP(1,EF,2
6 ):CALL MOTION(#4,-*F,2
7 *E):NEXT Y:NEXT X
```



Apple Geometry Tutor

[Applesoft BASIC on the Apple II]
Dear Sir:

This one-line program draws and names the regular polygons of 3 to 9 sides. It must be entered with no spaces (except those between quotes) and using ? instead of PRINT. Note the use of RUN instead of the usual RESTORE: GOTO 1 sequence to save space.

Thomas Bavis
Macedon, NY 14502

```
1 C=69:VTAB 24:HCOLOR=3:
2 ??:P=3:SIDES:HPLOT+C*
3 SIN(P),C+C*COS(P):FOR
4 1+C*SIN(Q),P+2*I+C*CO
5 ADA$:NEXT?:?A$:RUN:DA
6 A??:TRIANGLE,SQUARE,
7 NTAGON,HEXAGON,HEPTA
8 OCTAGON,NONAGON
```



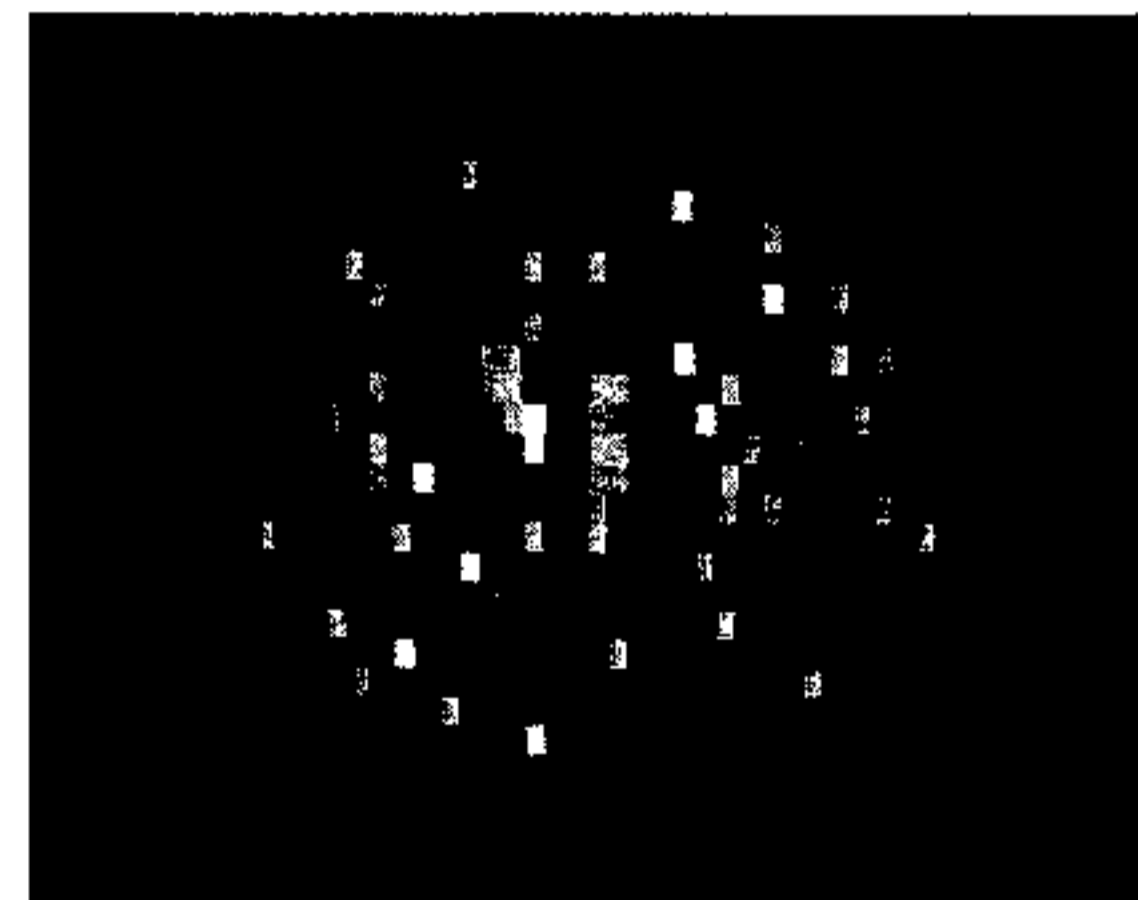
Graphics Spectacular

[Commodore BASIC on the C-64]
Dear Sir:

My submission creates a spiral of multi-colored squares—resulting in what appears to be an explosion of color. I used the abbreviated form of the BASIC keywords FOR [F (SHIFT) O] and POKE [P (SHIFT) O] in order to fit the entire program within one Commodore BASIC line.

Paul Kelley
Vancouver, BC, Canada

```
1 F=5:SA=8:FTO=C=1:TO96:STEP
2 (C)+C/8)*40:PSHIFT
3 (524+A,160:P)SHIF
4 96+A,C:NEXT
```



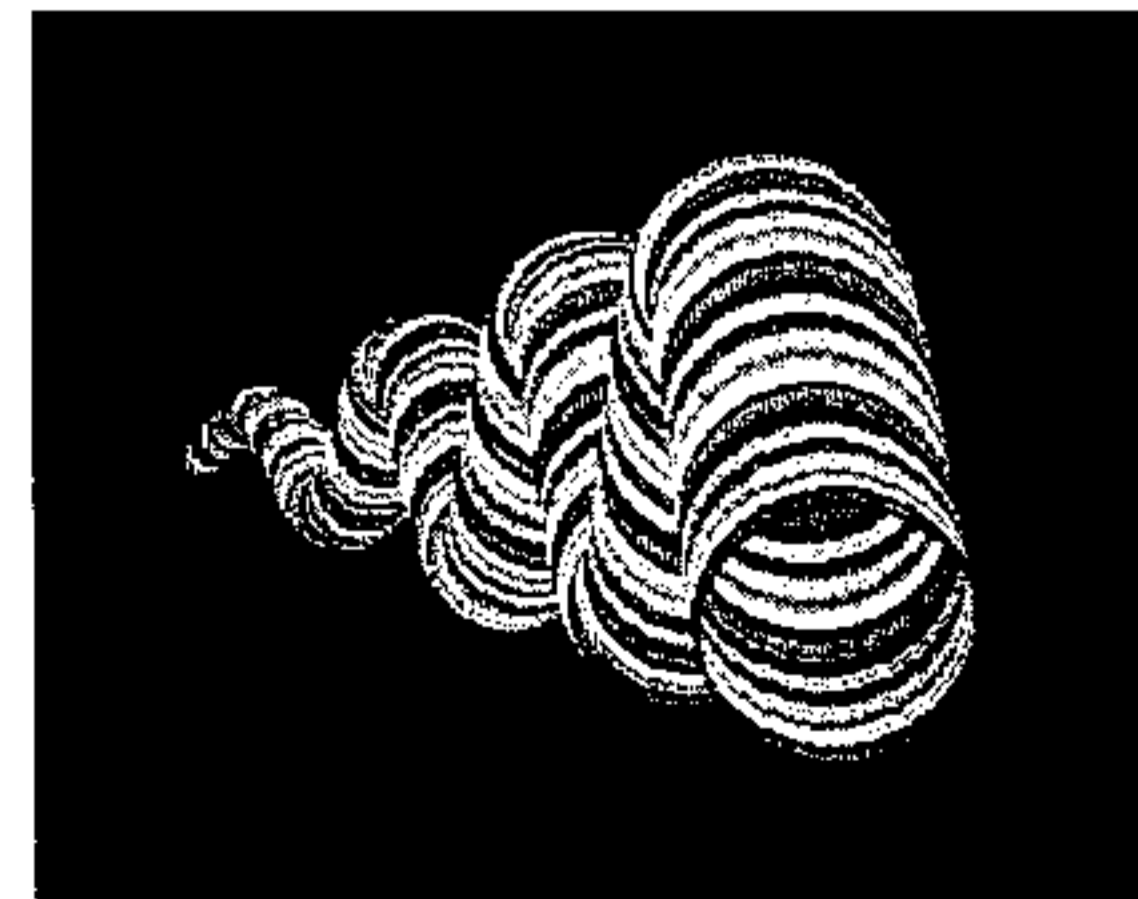
The Colorful Cosmic Worm

[Cartridge BASIC on the IBM PCjr]
Dear Sir:

Since I purchased my IBM PCjr it has convinced me that there is no other computer in its price range that can match its graphics capabilities. I created this one-line program as a demonstration to prove my point. I would like to challenge anyone with a non-PCjr system to duplicate the Cosmic Worm in one line.

William P. Scott
New York, NY 10028

```
1 CLS:KEY OFF:5:LEAR,4
2 4:FOR Z=1 TO 5:COLOR,STEP*10
3 000:FOR SIN(C+24)+C<15):N
4 0)Z*40,C=C-1:TO1000:PALETTE
5 +1:PALETTE(C<15):NEXT:GO
6 C,C:C=C*(C<15):NEXT:GO
```



All One-Liner submissions are subject to the same publishing criteria as the Letters to the Editor (explained in the magazine's masthead on page 4). If you have written a great One-Liner in any language on any computer covered by HCM, send it addressed to: Letters to the Editor, 1500 Valley River Drive, Suite 250, Eugene OR 97401. You too may win a cash prize and be immortalized in print!

HCM Review Criteria

Each month, *Home Computer Magazine (HCM)* reviews products designed for the Apple II Family, Commodore 64 and VIC-20, IBM PC and PCjr, and Texas Instruments 99/4A computers. *HCM* reviews take a detailed look at the quality, utility, and value of commercially available packages for these machines. Because our publishing charter forbids accepting outside advertising, we strive to make the scope and content of our review pages shine with a unique blend of humanistic frankness and objectivity.

Not only will you find all relevant information for making a wise purchase decision, but in some special cases we also provide nuggets of compu-prestidigitation.* For example, we frequently include essential documentation not furnished by the manufacturer. Additionally, each issue of *HCM* tries to review at least one outstanding product—a "Diamond in the Rough"—which, because of company size, marketing clout, or for some other reason, has not received the attention it deserves.

At the beginning of each review, a review-at-a-glance box provides the user with an instant assessment of the product. Each item will be evaluated, where relevant, with the criteria below.

HCM Review

<p>Name: _____</p> <p>Program Type: _____</p> <p>Machine: _____</p> <p>Distributor: _____</p> <p>Price: _____</p> <p>System Requirements: Disk Drive: _____</p> <p>Performance: _____</p> <p>Engagement: _____</p> <p>Documentation: _____</p>	<p>Disk Art: _____</p> <p>Graphics Quality: _____</p> <p>Sound Effects: _____</p> <p>Music: _____</p> <p>Speech: _____</p> <p>Engagement: _____</p> <p>Ease of Use: _____</p> <p>Ease of Set-up: _____</p> <p>Documentation: _____</p>
--	--

*** Performance—**
How well the product performs as intended; how well it takes advantage of a specific machine's capabilities; how well it responds to the user's commands; how effectively the graphics, sound effects, music, or speech are integrated with the software.

*** Engagement—**
Whether the game or activity has that intangible quality that holds players on the edge of their seats while the hours tick by unnoticed.

OR

*** Ease of Use—**
The degree to which a user can interact with the product without outside help; the ease and effectiveness of error-handling features; whether the actual reading level of the activity is appropriate for the suggested audience.

OR

*** Ease of Set-up—**
How well the product design facilitates easy installation.

*** Documentation—**
The quality of the printed matter that comes with the product, whether the instructions are clear and comprehensive; whether the machine configuration requirements are spelled out. Information such as how to load a program, use the keyboard, and restart an activity contributes to the documentation rating, as do tips on performance peculiarities.

Products may also be evaluated in the following areas:

*** Flexibility—**
Can the product be adapted to the specific needs of the users?

*** Cost/Benefit—**
Is the product worth the user's investment in time and money?

*** Necessity—**
Is the product a solution for which a problem already exists?

*** Originality—**
Is it unique in concept, or simply a "me too" product?

*** Longevity—**
The "Boredom Factor." Does the program sustain interest?

*** Rewards—**
Are the audio-visual rewards motivating and appropriate?

*** Concept Presentation—**
Are the concepts presented clearly, logically, and in depth?

*** Special Effects—**
How does quality of sound and visual effects rate? Do they enhance or detract from the product or learning process?

Attention Software Authors & Peripheral Inventors:

*** WANT TO BE DISCOVERED? ***

Home Computer Magazine Wants To Give You A Chance!

We are looking for home computer products that have not received the attention they deserve. Each month, we will be singling out one such package for special review. If you have a unique commercial product of exceptional quality—but your advertising and promotion budget has

not allowed you to capture major media attention—we want to see it. We will consider reviewing any product that meets our high standards.

We are an Equal Opportunity Reviewer!

In order to qualify for possible review, your product must:

1. Currently be available for purchase to readers of this magazine.
2. Make a unique and important contribution to the home computer industry.
3. Be of outstanding merit, quality, and value.
4. Be consistent with the type of machines and products we normally cover.

If you feel that your product qualifies, mail it to:

Home Computer Magazine
Attn: Editorial Submissions
1500 Valley River Drive, Suite 250
Eugene, OR. 97401

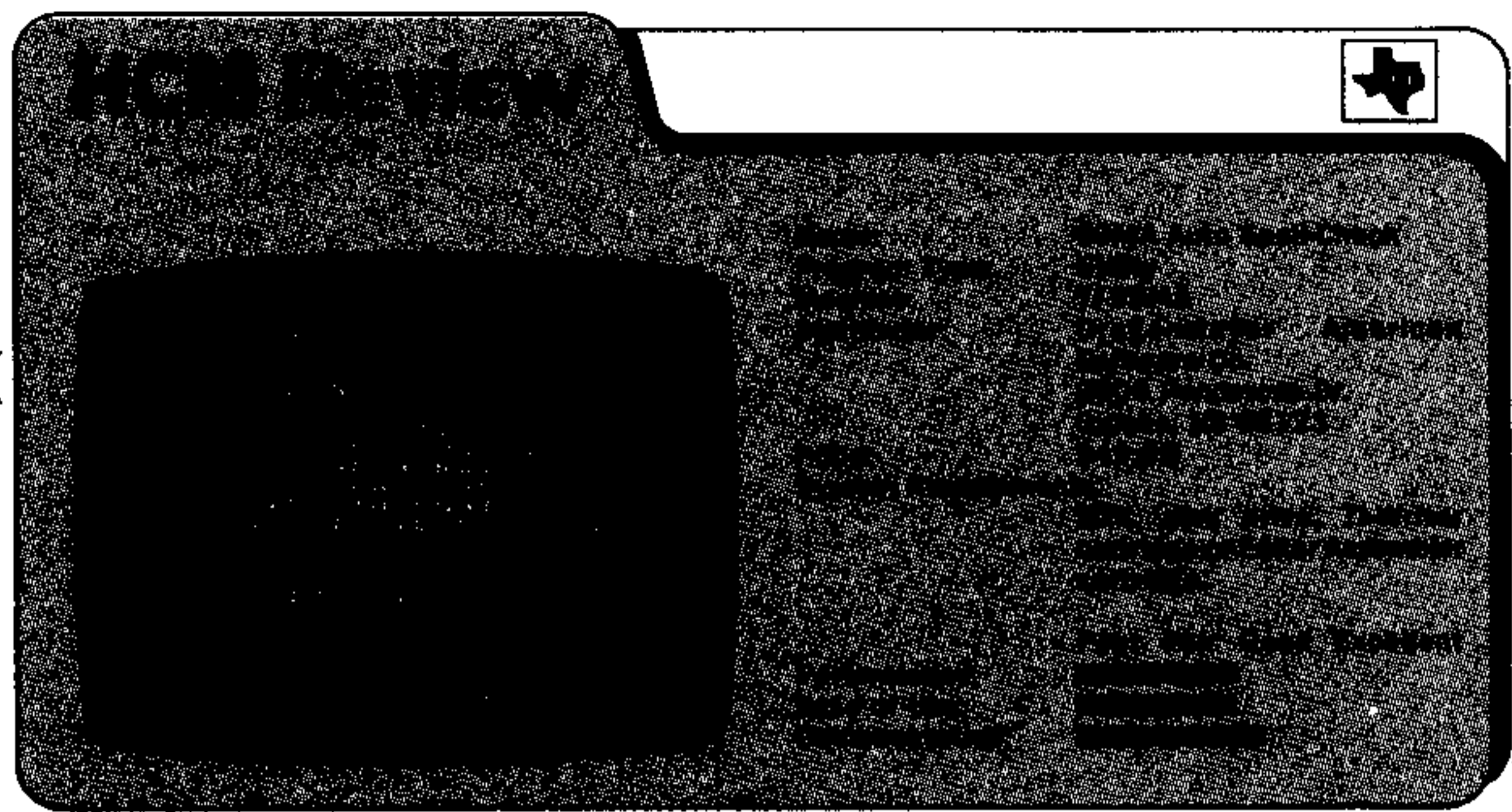
We reserve the right *not* to reply to each inquiry, so please do *not* contact us except to request return of your product. If you want your product to be returned, please include sufficient return postage.

***Compu-prestidigitation**

(kóm•pū•prēs•teh•dī•jeh•tā•shūn) —*n* 1. The magical quality of unexpected comprehension that results from presenting technical information about computers in a lively, entertaining, visually attractive and easy-to-understand format. 2. The magical tricks that make a computer sing, dance, and do all sorts of wonderfully useful things.

99/4A Auto Spell Check

A review
by **Steve Nelson**
HCM Staff



For help catching your misspelled words, check out this electronic spellchecker for the TI-99/4A. It's reliable. It's adaptable. But it's not exactly lightning fast.

Some people are lucky—they don't have a problem spelling words correctly. Other people, however, spell by trile and erur. And even the best spellers make mistakes once in a while.

In this computer-age automatic spell-checking programs have been developed, for most home computers, to help eliminate all spelling errors. But unfortunately, if you own a TI-99/4A, you have been out of luck—until now. Dragonslayer American Software Co. is offering a spelling checker for TI-99/4A owners called the *99/4A Auto Spell-Check*.

This program requires at least one disk drive, a 32K memory expansion, and a *TI-Writer* cartridge, or an *Editor Assembler* cartridge. (*99/4A Auto Spell-Check* will only check Display/Variable 80 formatted text files following *TI-Writer* or *Editor Assembler* character conventions.)

The main dictionary of 20,000 words, although somewhat small, is adequate when augmented by a user dictionary. *99/4A Auto Spell-Check* comes with an option called *Seedgen*, which lets you create user dictionaries of about 2,000 words each. This option is the program's best feature, because it allows you to tailor dictionaries to fit your specific word processing needs. For instance, if you use special words that are difficult to spell—such as medical or legal terminology—you can build a user dictionary of these terms, and use it to check your text.

How It Works

Auto Spell-Check scans your text, checking each word against the main dictionary, and stacking each word that it doesn't recognize into a "bad word stack." Once it completes its initial scan of your text, the program prompts you for a user dictionary name. It will check the words in the bad word stack against as many user dictionaries as you specify. After the program has completed checking your text against all dictionaries, you can review each remaining word in the bad word stack individually, and either correct the spelling, add it to your user dictionary, view the word in context, or disregard it. If you change the spelling of a word, it is flagged and—after you complete the review process—the program automatically updates your text file with the corrections. If you select a word for addition to your user dictionary, the dictionary is also updated. (You must have a user dictionary file already prepared in order to update it.) Because this program just stacks

words that it doesn't recognize, it leaves it up to you to look up and key in the correct spellings before it updates your text file.

Problems

Unfortunately, *Auto Spell-Check* has some real problems. The execution of the program is slow—slow enough that if your text length is small, like a single-page letter, you could probably proof it yourself and make corrections faster than the *99/4A Auto Spell-Check* could do it (assuming you can recognize a misspelled word).

If you are working with only one disk drive, the constant swapping of disks can be quite cumbersome. You can, however, cut the number of disk changes required if you have a multi-drive system.

“... an option called *Seedgen* lets you create user dictionaries of about 2,000 words each.”

If a corrected word is not equal in length to the word it is replacing when the program updates your text, the program deletes the misspelled word, breaks the sentence, and inserts the new word on the line below. This means that you may have to reformat your text after it has been updated—somewhat of an inconvenience, to say the least...

After using *99/4A Auto Spell-Check* for a couple of weeks, I found myself resenting its relatively slow operating speed. But because the TI-99/4A is limited to 48K (even with the 32K memory expansion), its slow speed can be attributed to the program having to make frequent accesses of the disk. Its slowness, however, is something that should be considered when deciding whether or not to purchase this product.

If your typing skills are a bit on the sloppy side, or if you handle so much text that you can never catch every error, then the *99/4A Auto Spell-Check* may be worth looking at—provided you don't mind the product's limitations.

HCM

Graphics Magic At Your Fingertips



A Review of Super Sketch
by Steve Nelson
HCM Staff

*It's a bit awkward—but you can draw detailed pictures
with this graphics tablet for the TI-99/4A and C-64.*

In the past, extensive programming was necessary to produce graphic images on a computer screen. Today, hardware/software companies are creating many different products designed to make computer graphics as easy and as familiar as drawing on paper.

Generally, computer graphics can be created 3 ways: (1) by brute-force programming (generating pre-planned graphics screens with a language like BASIC), (2) using graphics software tools (programs that make it easy to build preplanned screens), and (3) using special input devices (light-pens, mice, joysticks, tablets, or sketch pads).

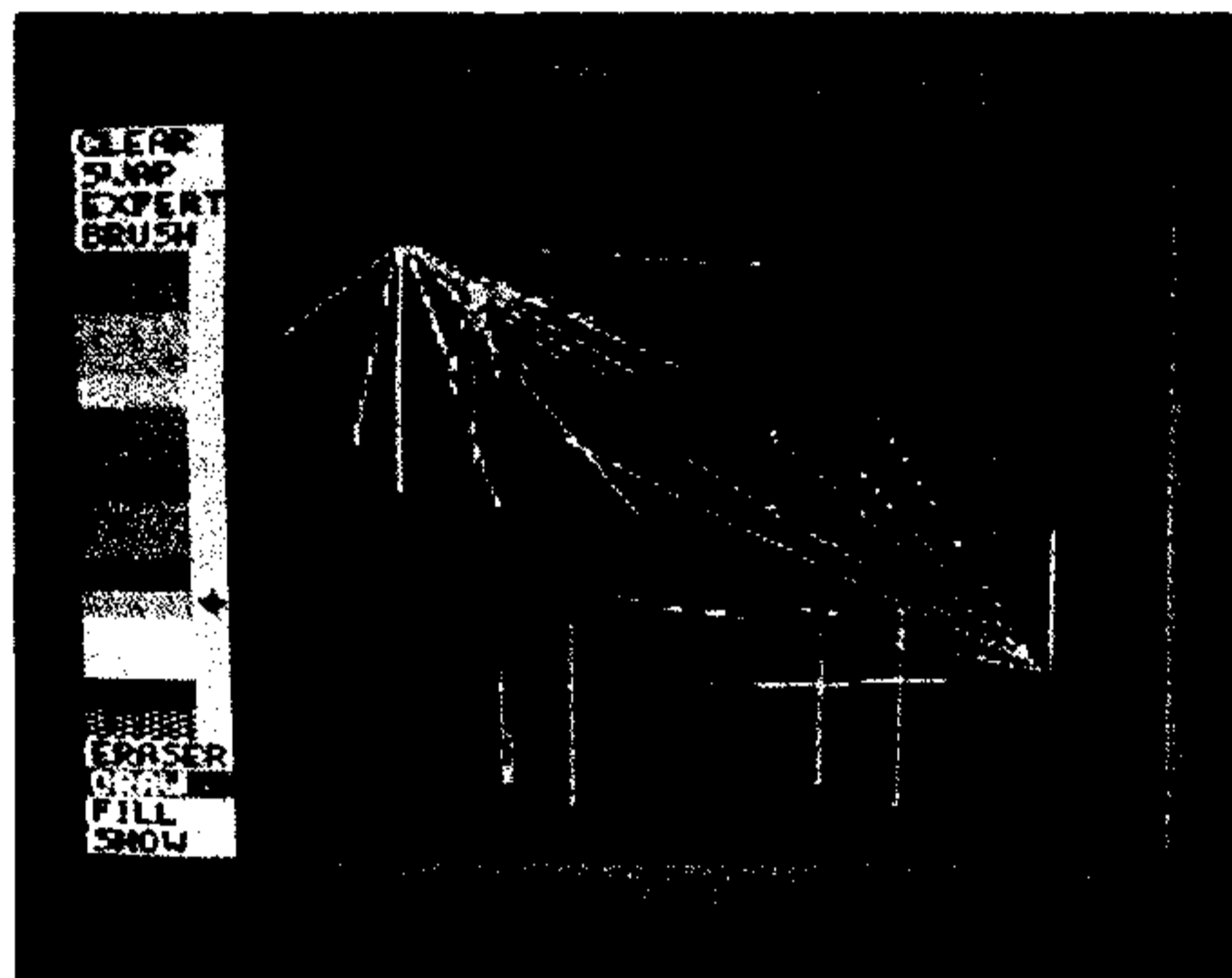
Super Sketch from Personal Peripherals is of the third variety—a sketch pad for the home computer, complete with graphics software included in the package.

Drawing

Sketch pads combine software and hardware to create a *frame buffer* that corresponds directly with the grid of the monitor screen. When you draw on the tablet or pad, its coordinates are registered in the buffer and transferred to the screen. The computer receives an X, Y coordinate and plots a "pixel" (picture element) on the screen in the color that you have selected. The software that controls this also lets you change colors, manipulate the brush size, and choose from a wide variety of options when creating your graphic design.

*"My first attempt to trace even
the most simple of designs—the
bluebird picture supplied with
the package—came out looking
like a reject from a
finish-the-drawing contest."*

Unfortunately, the image that actually appears on the screen falls prey to the same problem that almost all computer graphics programs succumb to—"blocking." The line you draw is thicker than intended, or jagged around the edges. This happens because of the nature of digital computers, which "draw" on the monitor screen by creating a graph made up of horizontal rows and vertical columns. Each tiny square is called a pixel. Each pixel on the grid is represented by an X, Y coordinate. The computer "turns on" each pixel as directed by the software, and assigns it the color that you have selected. Occasional



Some of the "expert functions" on the TI-99/4A.

blocking results, even with the highest quality monitors, due to the digital signal nature of the signal.

User-Friendliness is More Than Sketchy

Super Sketch is a hard plastic tablet with a stylus-tipped mechanical arm that you move to draw or trace. Its plug-in cartridges and fingertip controls make it very user-friendly. The two versions of *Super Sketch* software (for the TI-99/4A and Commodore 64) are quite different—the C-64 version is much more flexible, offering several options that are unavailable on the TI version.

For instance, the C-64 version has added features like the ability to copy portions of your picture to other areas, a window option, and even a mirror function which allows you to draw on one part of the screen while the program mirrors your drawing on another part. The most notable difference between the two versions is their method of storing pictures. The C-64 allows you to save to either disk or tape; the TI version lets you save only to tape.

The documentation for both versions is very good, with step-by-step instructions that take you through each individual function.


Performance

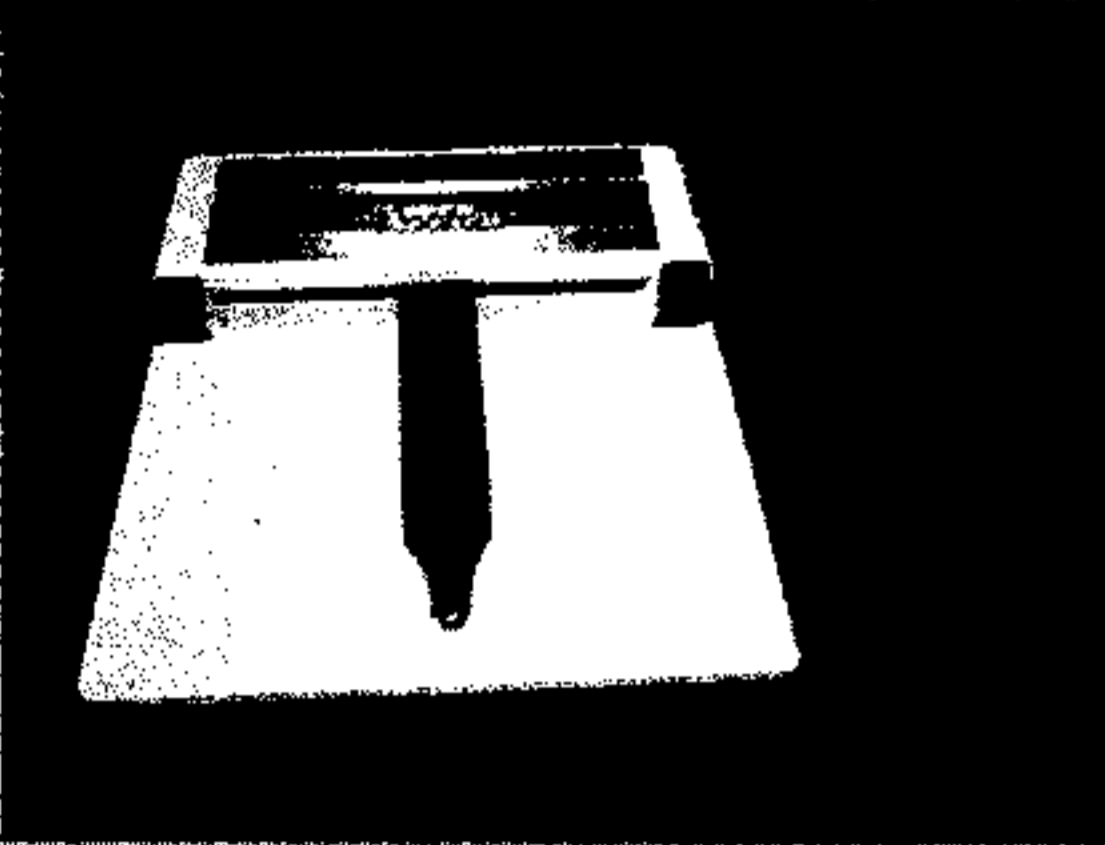
Setting up is a snap, and with a brief look at the instructions, you can begin drawing right away. The first thing you will notice about *Super Sketch* is the stickiness of its drawing arm, making it very

difficult to draw a smooth line freehand. This problem becomes especially evident when tracing. My first attempt to trace even the most simple of designs—the bluebird picture supplied with the package—came out looking like a reject from a finish-the-drawing contest. I find this problem to be a real drawback (no pun intended), as tracing should be one of the things sketch pads do best.

The response of the cursor—which indicates the position of the stylus in relation to the screen picture—is good when you draw with a slow, deliberate motion. If, however, you like to just scribble

HCM Review





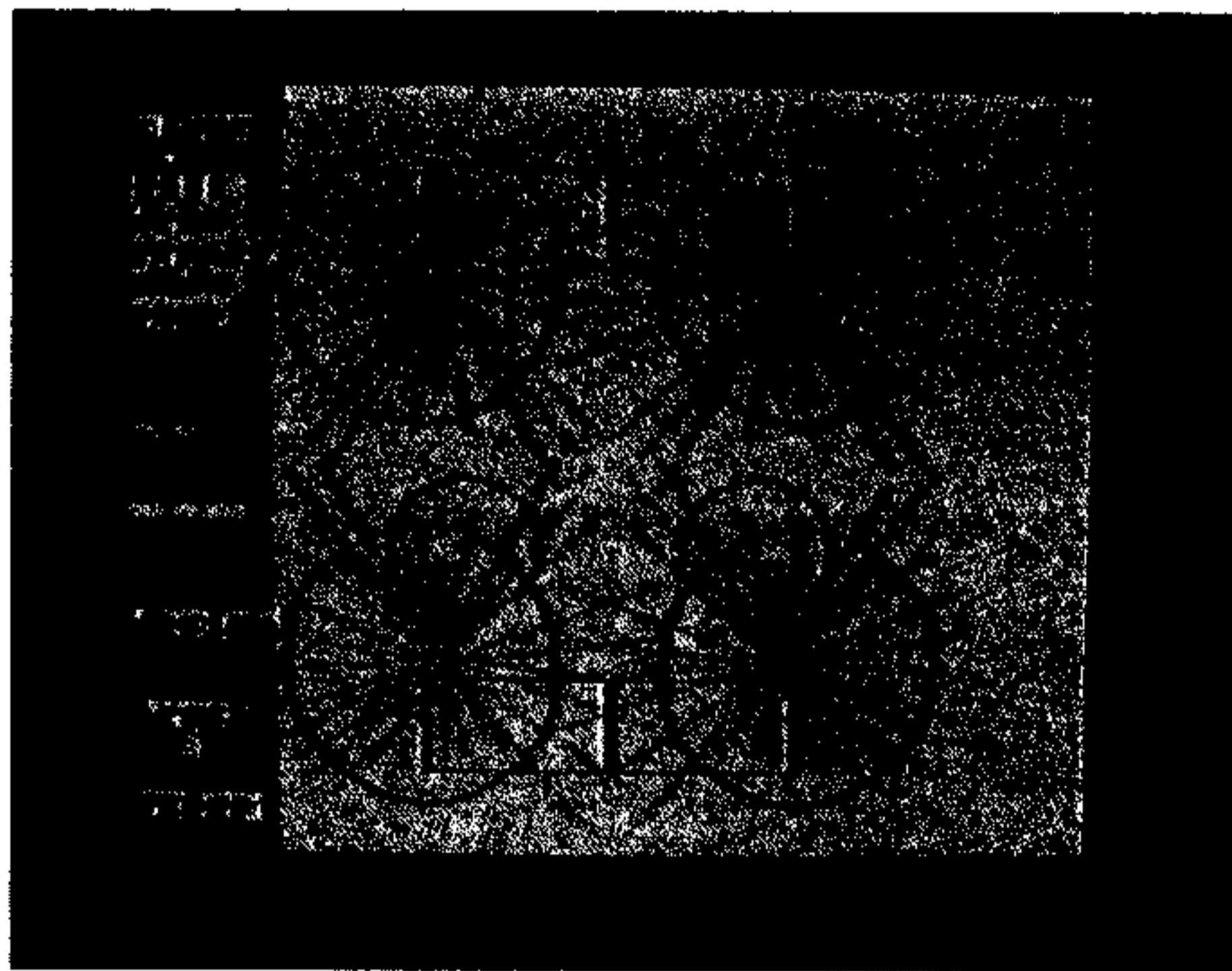
Name: Super Sketch
Program Type: Graphics Tablet
Machines: TI-99/4A, C-64
 (Apple II and IBM PCjr versions were not available in time for this review)

Distributor: Personal Peripherals
 930 N. Beltline Suite 120
 Irving, TX 75061

Price: \$59.95

Performance:
Ease of Use:
Documentation:

Poor Fair Good Excellent



Some of the "expert functions" on the C-64 version.

away freehand, drawing as the spirit moves you, the cursor lags noticeably—especially when using a thick brush style. This is a real problem, as it can lag far enough behind to actually lose track of your drawing. There is also another problem with the cursor—it is too large. When you are drawing with a fine line, or using the Eraser or Fill functions to touch up tiny portions of your picture, the cursor can actually obscure your view.

All of the special features of *Super Sketch* can be accessed from the main menu. These include choosing colors (the C-64 version has 16 different colors, and the TI-99/4A has 15), changing brush sizes, filling your drawing with color, adding texture, and erasing. Selection of an "expert" menu helps you draw straight lines, boxes, circles, and rays—and, on the C-64, accesses the added features already mentioned.

"Setting up is a snap, and with a brief look at the instructions, you can begin drawing right away."

Color Me Dangerous

Some of these special functions work better than others. For instance, the Fill function can be treacherous. This feature allows you to select an area of your drawing and fill it with a certain color. You

must, however, be certain not to have any gaps in the borderline around the area to be filled; otherwise, the fill color will spill over into the surrounding color and ruin your picture.

If you have a lot of intersecting lines in your drawing, the Fill option can be tricky here as well. If, in the process of adding color to your picture, the cursor touches a line, it can cause the line to change color. This in turn can cause every other line connected to it to also change color. It can even cause the entire picture to change color! (The C-64 version has an Undo option which will bring the screen back to the condition it was in before you made a mistake. This function is sorely missed on the TI-99/4A version.)

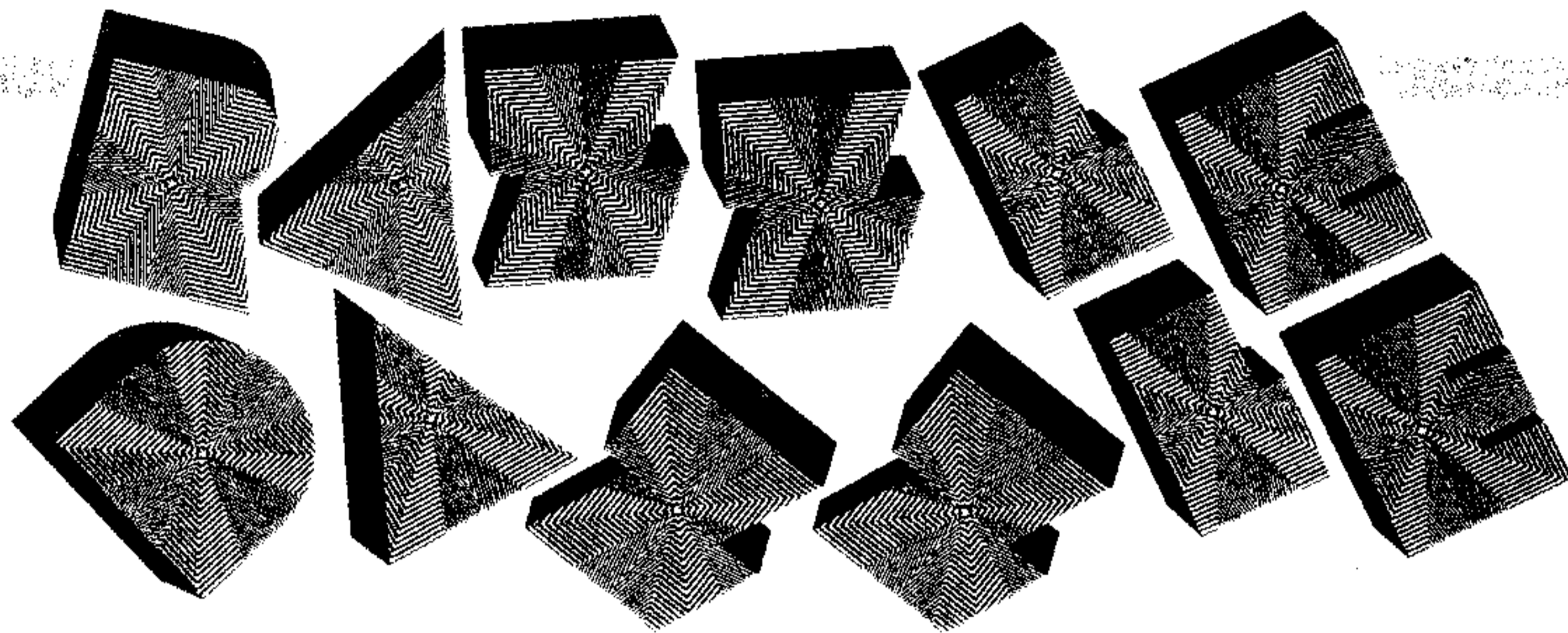
Drawing with *Super Sketch* may seem awkward at first, but with practice and patience you really can create spectacular graphics. The variety of bright colors and the special features of *Super Sketch* give you a powerful graphics package for the money. In addition, *Super Sketch* is very simple to use—especially for those who have little or no experience with computers.

Although I'm certain that there are people out there who trace with this sketch pad and are satisfied with its performance in that capacity, I certainly cannot recommend it for that purpose alone. As an artist's tool, the TI-99/4A version lacks flexibility, and both versions are much too clumsy to give reliable results.

However, as most graphics programs developed for the TI-99/4A have been programming aids rather than drawing tools, *Super Sketch* does offer an inexpensive and viable alternative. And although it's awkward to use, it does give you the ability to draw on your monitor without using the keyboard.

C-64 owners have a variety of graphics packages to choose from, including touch pads and light pens. The extra functions available on the C-64 version make *Super Sketch* very competitive with other graphics systems designed for this machine. [Peripheral Products plans to release a "souped-up" version (*Super Sketch II*) for the IBM PC and PCjr, and the Apple II family in early November—Ed.]

This graphics tool will not satisfy every budding computer artist. But some may accept the medium for what it is, anticipating its effects as part of their design. If you have been looking for something like *Super Sketch*, by all means try it before you buy it. **HCM**

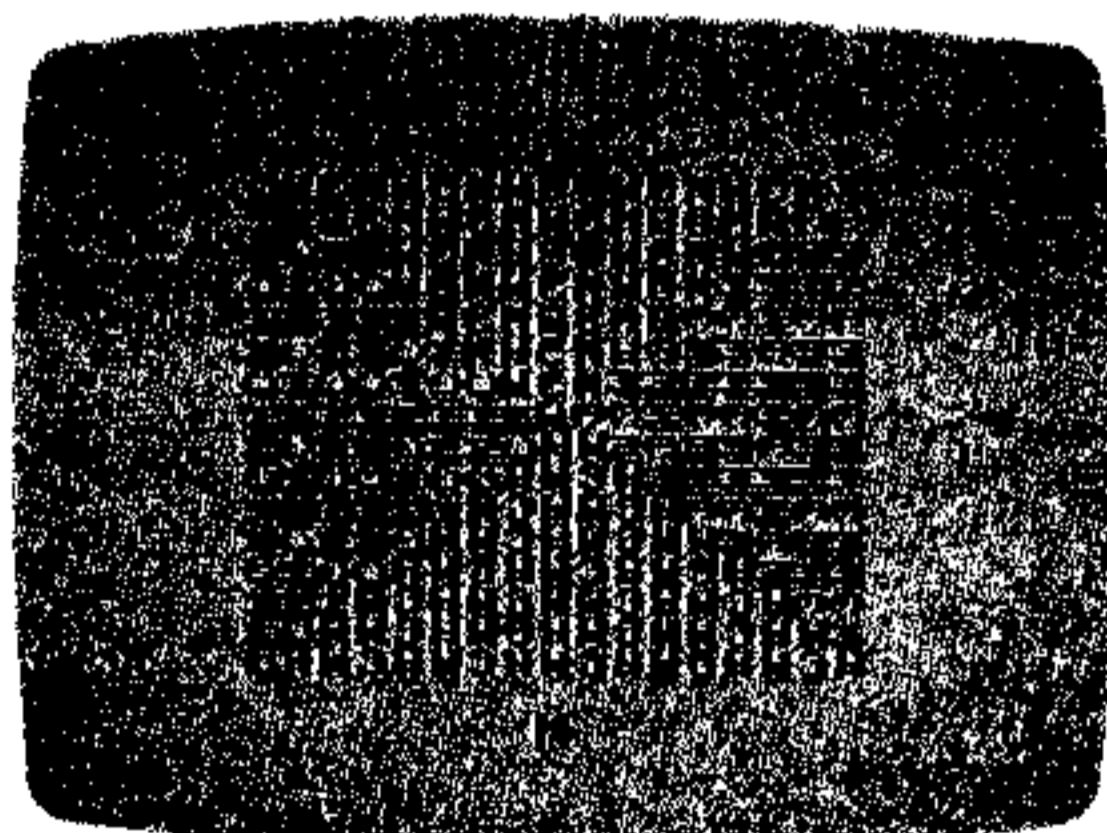


by **William K. Balthrop**
HCM Staff

Razzle 'n' dazzle 'em with spectacular video effects as you explore the mysteries of home computer graphics. Pixel magic is finally within your power . . .

With all the new technology on the market today, you may wonder whether the TI-99/4A can still compete as a powerful graphics computer. You bet it can! The TI is capable of a number of things that the newer, more expensive machines balk at.

Here, for example, we present one method of using the TI-99/4A as an effective graphics tool—in ways you may not have considered. This routine requires an Extended BASIC cartridge.



This photo shows how Turn can be used to create spectacular effects with very little programming effort.

Character Graphics

On the TI-99/4A, you can redefine the character set to create your own custom characters. This allows you to design spectacular graphics on the screen.

All of the characters you see on your screen are actually made up of very small dots called *pixels*. Each character has 64 pixels, laid out in an 8x8 grid. By telling the computer which pixels you want turned on (visible on the screen), you can change a character's shape. Each digit of hexadecimal code in a character's definition controls four pixels (dots within the character). It takes 16 hexadecimal digits to define the pattern for one character.

In Figure 1, pixels that are turned on are designated by a 1; those turned off are designated by a 0. The character "A" is formed by the *on* pixels.

Usually a programmer is left to figure out the hexadecimal codes after defining the new shapes with a piece of graph paper and pencil. But this program, *Characters*, contains three Extended BASIC sub-

FIGURE 1
CHARACTER BIT PATTERN

Row	Bit Pattern	HEXDigits
1	0 0 0 0 0 0 0 0	00
2	0 0 1 1 1 0 0 0	38
3	0 1 0 0 0 1 0 0	44
4	0 1 0 0 0 1 0 0	44
5	0 1 1 1 1 1 0 0	7C
6	0 1 0 0 0 1 0 0	44
7	0 1 0 0 0 1 0 0	44
8	0 1 0 0 0 1 0 0	44

Character definition string for the "A" graphics pattern = "003844447C444444"

routines which alter the character set automatically. The three routines here take the patterns for the characters passed to the routine, alter them, and then store the new shapes at the character location you specify.

Mirror

The first subroutine, Mirror, will cause characters to appear as their mirror images—turning them around as if you were looking at them from the other side of the screen. This routine takes the hexadecimal code for each row of pixels in the characters and reverses them.

Each character has eight pixels per row. Consider a row of pixels that has the following pattern:

11001010 CA

First we need to switch the location of each hexadecimal character to produce the following:

10101100 AC

The task now is to reverse the patterns within each four-pixel half of the row. This is done by reversing the bit pattern for each hexadecimal digit:

01010011 53

This process is then repeated for each of the other rows.



Flip

The second subroutine, Flip, will do just what the name implies: It will turn the characters it affects upside down. This process is easier to accomplish than the first routine—all that is needed is to reverse the order of the rows in the character's pattern definition. Remember, each row is made up of eight pixels, or two hexadecimal characters. The subroutine works on the character definition string as follows:

```
0123456789ABCDEF (Original hex codes)
01 23 45 67 89 AB CD EF (Row pairs)
EF CD AB 89 67 45 23 01 (Reverse order)
EFC DAB8967452301 (Final result—flipped)
```

“... by redefining the character set to create custom characters, you'll be able to design spectacular graphics right on the screen.”

Turn

This final subroutine, Turn, can be used to turn a character clockwise 90 degrees (on its side). If you repeatedly turn the same character, placing the modified shape back in that character's original location, you will continue to rotate it clockwise.

This routine is much more complex than the previous two, and consequently is much slower. Having explained Flip and Mirror, we'll leave it as an exercise for you to discover how Turn works.

Use Them Yourself

You can use these routines in your own programs to manipulate any characters available on the computer with Extended BASIC (ASCII 32 through 143).

All three subroutines require that you pass two parameters. The first parameter is a string containing the characters you want to convert. The second parameter is an ASCII value that indicates where the new characters will be placed. The following are examples:

CALL MIRROR("ABC",96) Place a mirror image of A in ASCII character 96, B at 97, and C at 98.

CALL FLIP("ABC",65) Turn the characters A, B, and C upside down and place the new shapes back into their original position, starting at ASCII 65.

CALL TURN("ABC",65) As used in the demo routine supplied with these subroutines, the letters ABC are turned 90 degrees clockwise with each call of the routine. In addition, before rotating the characters, we placed the shapes for the characters HCM at ASCII locations 65, 66, and 67.

HCM

XB

Razzle Dazzle requires TI Extended Basic.

Characters (TI-99/4A) Explanation of the Program

Line Nos.

100-160 Program header.
170-330 Demo to access the three subroutines.
Draws a spiral of the letters HCM.
340-400 Mirror subroutine.
410-460 Flip subroutine.
470-590 Turn subroutine.

```
100 REM *****
110 REM ***** CHARACTERS *****
120 REM *****
130 REM BY WILLIAM K. BALTHROP
140 REM HOME COMPUTER MAGAZINE
150 REM VERSION 4.5.1
160 REM TI EXTENDED BASIC
170 CALL CLEAR
180 DISPLAY AT (12,1) ERASE ALL: "WORKING I
190 N$="HCM" : FOR Z=0 TO LEN(N$)-1 :
: CALL CHARPAT(ASC(SEGS(N$,Z+1,1)),A
: CALL CHAR(96+Z,A$) : CALL CHAR
(65+Z,A$) : NEXT Z
200 N$="ABC" : CALL TURN(N$,65) : FOR
Z=0 TO LEN(N$)-1 : CALL CHARPAT(65
+Z,A$) : CALL CHAR(96+LEN(N$)+Z,A$)
: NEXT Z
210 CALL TURN(N$,65) : FOR Z=0 TO LEN(N
$)-1 : CALL CHARPAT(65+Z,A$) : CAL
L CHAR(96+(2*LEN(N$))+Z,A$) : NEXT
Z
220 CALL TURN(N$,65) : FOR Z=0 TO LEN(N
$)-1 : CALL CHARPAT(65+Z,A$) : CAL
L CHAR(96+(3*LEN(N$))+Z,A$) : NEXT
Z
230 X=12 : Y=16 : P=0 : FOR Z=2 TO 1
6 STEP 2 : FOR Z1=1 TO Z-1 : CALL
HCHAR(X,Y,96+P) : Y=Y+1 : P=P+1 :
: IF P=LEN(N$) THEN P=0
240 NEXT Z1
250 FOR Z2=1 TO Z-1 : CALL HCHAR(X,Y,9
6+P+LEN(N$)+P) : X=X+1 : P=P+1 : IF
P=LEN(N$) THEN P=0
260 NEXT Z2
270 FOR Z3=1 TO Z : CALL HCHAR(X,Y,96+
(2*LEN(N$))+P) : Y=Y-1 : P=P+1 :
: IF P=LEN(N$) THEN P=0
280 NEXT Z3
290 FOR Z4=1 TO Z : CALL HCHAR(X,Y,96+
(3*LEN(N$))+P) : X=X-1 : P=P+1 :
: IF P=LEN(N$) THEN P=0
300 NEXT Z4 : NEXT Z
310 CALL HCHAR(22,16,98) : GOSUB 330 :
CALL HCHAR(22,16,101) : GOSUB 330 :
CALL HCHAR(22,16,104) : GOSUB 33
320 GOSUB 330 : GOTO 310
330 FOR TD=1 TO 20 : NEXT TD : RETURN
```

```
340 REM
350 SUB MIRROR(A$,A)
360 X$="0123456789ABCDEF" : Y$="084C2A
6E195D3B7F" : FOR X=1 TO LEN(A$) :
CALL CHARPAT(ASC(SEGS(A$,X,1)),B$) :
D$&SEGS(Y$,POS(X$,SEGS(B$,Z,1)),1)
: NEXT Z
380 E$=" " : FOR Z=1 TO LEN(D$) STEP 2 :
: E$=E$&SEGS(D$,Z+1,1)&SEGS(D$,Z,1)
: NEXT Z : CALL CHAR(A,E$) : A=A+
1 : IF A>143 THEN SUBEXIT
390 NEXT X
400 SUBEND
410 REM
420 SUB FLIP(A$,A)
430 FOR X=1 TO LEN(A$) : CALL CHARPAT(A
SC(SEGS(A$,X,1)),B$) :
E$=" " : FOR Z=LEN(B$)-1 TO 1 STEP
-2 : E$=E$&SEGS(B$,Z,2) : NEXT Z :
CALL CHAR(A,E$) : A=A+1 : IF A>1
43 THEN SUBEXIT
450 NEXT X
460 SUBEND
470 REM
480 SUB TURN(A$,A)
490 FOR Z=0 TO LEN(A$)-1 : CALL CHARPA
T(ASC(SEGS(A$,Z+1,1)),B$) : FOR Y=1
TO 8 : P(Y)=0 : NP(Y)=0 : NEXT
Y
500 FOR Y=1 TO 15 STEP 2 : B1=ASC(SEGS
(B$,Y,1))-48 : IF B1>9 THEN B1=B1-
7
510 B2=ASC(SEGS(B$,Y+1,1))-48 : IF B2>
9 THEN B2=B2-7
520 P=(INT(Y/2)+1)*B1+16+B2 : NEXT Y
530 FOR Y=1 TO 8 : FOR X=0 TO 7 : IF
(P(Y)AND 2^X)=2^X THEN NP(8-X)=NP(8
-X)+2^(Y-1)
540 NEXT X : NEXT Y
550 NPS=" " : FOR Y=1 TO 8 : B1=INT(NP
(Y)/16) : B2=NP(Y)-B1*16 : IF B1>9
THEN B1=B1-9 : B1=B1+48 : ELSE B2=B2+48
560 IF B2>9 THEN B2=B2+55 : ELSE B2=B2+48
570 NP$=NP$&CHR$(B1)&CHR$(B2) : NEXT Y
: CALL CHAR(A,NP$) : A=A+1 : IF A
>143 THEN SUBEXIT
580 NEXT Z
590 SUBEND
```

The **Pacific Northwest IBM PC Users Group** in Bellevue, Washington gets together the second Tuesday of each month to exchange information and ideas. This group formed in the summer of 1981 and has a staggering 550 members! The diverse interests of the members make this group quite interesting; their Special Interest Groups include Lotus 1-2-3, DOS, C language, generic financial model, communications, and BASIC. If you would like to learn more about this large group, call: Larry Shaw, P. O. Box 3363, Bellevue, WA 98004, (206) 628-1141.

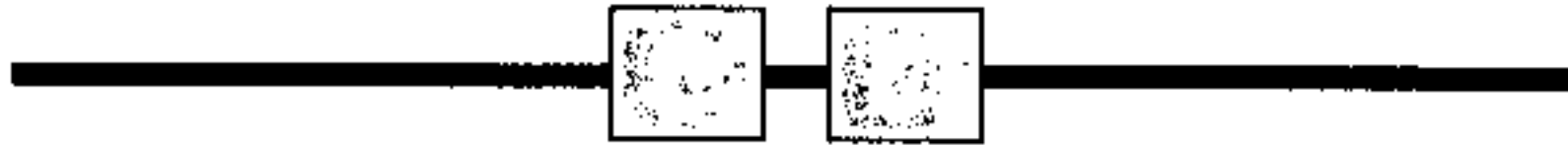


Group Grapevine is pleased to announce the formation of the **Queensborough Community College 99'er User Group** in Bayside, NY. According to Frank Cotty, the group will begin meeting this fall and in keeping with the spirit of the community college, the group will be open to students and non-students alike. For more information, contact: Frank Cotty, 56 Avenue & Springfield Boulevard, Bayside, NY 11364.

Salute! Group Grapevine has received a letter from the 50-member **TI 99 IT User Group** in Montecchio, Italy. The group has a library of over 400 TI BASIC programs, more than 400 Extended BASIC programs, and a good quantity (and quality) of assembler programs. Their bimonthly bulletin includes free listings, tricks, tips, and descriptions of new programs. TI 99 IT is in contact with clubs in Belgium and Australia and would like to expand their contacts to include the U.S. If you would like to exchange newsletters, software, or program listings, write (in English): Perlini Settimio, TI 99 IT User Group, Via 21 Gennaio 152, 61020 Montecchio (PS), ITALY.

Chip Ragsdale, president of the **MIT Lincoln Laboratory TI-99/4A User Group** has informed Group Grapevine that they are no longer in existence (in Massachusetts) due to his transfer to Florida. Since Chip has been in Florida, he has organized a new MIT group and they are 52 members strong at the present. Their group meets once a month, and they are presently working on their first newsletter. They also have a cassette library which boasts 480 programs, with many more yet to be keyed in. For more information, contact: Chip Ragsdale, 8820 90 Way North, Seminole, FL 33543.

Larry Robinson, vice president of the **Macon Area TI-99/4A User Group**, has informed Group Grapevine of the formation of the **Warner Robins Chapter**, which meets at the Warner Robins Recreation Center from 10 a.m. to 1 p.m. every Saturday. A lecture series on BASIC programming techniques and hardware use has been continuing for some time. The group also has tentative plans for hosting a computer exposition featuring displays by area users' groups. This is designed to spread the word about users' groups in the middle Georgia area. If you are interested in more information about the Macon Area group or the Warner Robins Chapter meetings, write: Larry Robinson, 503 Third Avenue, Bonaire, GA 31005.



Here's a group that has come a long way in a very short time. They are the **Crossroads Commodore Users Group** in Victoria, Texas. They have been in existence for about eight months and boast a membership of 45. According to spokesperson Jerry Guy, they were originally a support group for the Commodore users of Victoria and vicinity who are out there in the "wasteland" between Houston, San Antonio, Austin, and Corpus Christi. (Group Grapevine looked up Victoria in an atlas and they really are stuck out there—if you draw lines between the cities mentioned above, it is a "crossroads.") The original members of the group began swapping programs, which has resulted in a library of about 500 programs. They are affiliated with the Commodore Houston Users Group (HUG) and the Toronto PET users. Membership is \$3.00/month. If you are interested and would like more information, contact: Jerry Guy, 417 Irma Drive, Victoria, TX 77901, (512) 575-0342.

Bloodthirsty! That's how Mr. Ikram of a Commodore users group in **Rochdale, England** describes his group with reference to exchanging good American software for English software. The group prefers cassette software unless American groups or individuals can duplicate some of their disk or cartridge software to tape. The group would also enjoy receiving newsletters and any other information from American Commodore groups. If you are interested in contacting this group, write to: M. Ikram, 20 Equitable Street, Rochdale OL11 1JQ, Lancashire, England, UK.

We received greetings from the **Lansing Area Commodore Club** in East Lansing, Michigan via president Jae Walker. LACC has been in existence for a year-and-a-half and is a whopping 240 members strong. They meet on the second Thursday of each month at All Saints Episcopal Church at 7 p.m. Anyone wishing further information can contact Jae at: P. O. Box 1065, East Lansing, MI 48823-1065, (517) 351-7061.

Group Grapevine just received notice of the birth of another Commodore 64 users group. The **Akron Area C-64 Users Group** in Cuyahoga Falls, Ohio has a membership of well over 100, with interests ranging from business and education to gameplaying, ham radio, etc. They meet the fourth Saturday of each month from 1 to 4 p.m. at the Green Middle School, Green Township, just south of Akron. The Green Middle School has a computer lab with 15 Commodore 64's that are available for use by the membership. If you are a Commodore 64 owner and would like more information about this users group, write or call: Paul M. Hardy, 2453 Second Street, Cuyahoga Falls, OH 44221, (216) 923-4396.

HCM

Midnite Mason

A review by
Steve Nelson
HCM Staff

HCM Review

◆ Diamond-in-The-Rough



Name:	Midnite Mason
Program Type:	Arcade game
Machines:	T1-9844
Distributor:	Software Specialties, Inc. P.O. Box 3304 Evergreen, CO 80439
Price:	\$24.95 Cartridge
System Requirements:	Joystick optional
Performance:	_____
Engagement:	_____
Documentation:	_____

A mason's work is never done—especially when the building he is working in is haunted by four nocturnal nasties.

Midnite Mason is an entertaining little game that pits your speed and strategy against a group of pesky ghosts. It's four against one, and if you take a wrong turn, you lose. The scenario is this: You are a mason worker who has left his tools inside a building. No problem, just go back and get them, right? Wrong! At night, the building is haunted by four nasty ghosts who seem to like your tools where they are. What's worse, all of your tools are scattered around the building and the only way to get them is to climb ladders to several different floors, dodge the ghosts, and grab your tools and run. To recover a tool, just run past it, but watch out for those spooks—one touch and you're a goner!

When you successfully recover all 7 tools, the ghosts vaporize, and you advance to the next level of play. *Midnite Mason* has 6 different mazes (buildings), and 4 levels of difficulty—the ghosts actually get more intelligent as they pursue the mason. Once you successfully complete the fourth level, the ghosts' intelligence decreases, but their speed increases. Got all that? In other words, the ghosts get smarter, then dumber and faster. The game continues like this indefinitely.

Making the game even more difficult is a time limit for gathering up your tools. A timer counts backwards from 900 to 000 in increments of 10, forcing you to keep moving. If you run out of time, you lose one of the three masons.

The response of the game is very good, except that the mason sometimes hesitates at the top of ladders before moving left or right, and as a result, usually gets mugged by a passing ghost. I didn't have this problem when using the keyboard, which makes me suspect the joystick-read routine. I tried out several different joysticks and the problem was still there. (It doesn't happen very often but it can cost you a mason when it does.) The mason can elude the ghosts by running past ladders, because the ghosts tend to climb them instead, even in the heat of the chase.

Pick A Hole

The mason, however, is not totally defenseless. He can chop a hole in the maze with his pick very quickly and cause a ghost to fall to the lower level. He can also fill in a hole amazingly fast, allowing him to escape from one tier to another, then quickly turn around and chop a hole again, forcing any pursuing ghosts to find another way to get to him, and giving him more time to gather his tools.

Although *Midnite Mason's* game plan sounds complicated, it is very simple to play. The mason's

speed is agonizingly slow, especially when a couple of the ghosts are hot on his trail; but with strategy, you can avoid the touch of death—at least for a while.

I found *Midnite Mason* to be an enjoyable game, and one that is difficult enough to require a determined effort on the part of the player in order to score points. Once you earn 5000 points, you receive a new mason—and by then you will probably need one.

The documentation is brief, but thorough, except that it fails to tell you to release the (ALPHA LOCK) key when using joysticks.

Midnite Mason is a good game, but it could be better. For instance, as you get past the first four levels, the number of tools that the mason must retrieve should increase. Also, the mason should be able to jump off the ladders if the ghosts trap him. And finally, it would be nice if the program could keep track of the high score

“The mason's speed is agonizingly slow, especially when a couple of the ghosts are hot on his trail; but with strategy, you can avoid the touch of death—at least for a while.”

while the cartridge is engaged, giving you something to shoot for on each repeat game you play. Incorporating these changes would make *Midnite Mason* one of the better games of its type.

Video game players who are used to complex game plans and hordes of monsters or aliens to kill will probably find *Midnite Mason* a bit on the slow side. Compared to *Buck Rogers and the Planet of Zoom* (Reviewed in Vol. 4, No. 4 of *Home Computer Magazine*), its level of excitement and story line are somewhat mundane, but I found its relative slowness and simplicity to be refreshing, and a lot easier on the eyes.

The game's sound effects are humorous and well done, and I love the little dance the mason does when he gets all 7 tools and the ghosts vaporize.

Overall, this is a very good version of a well-used story line—get prizes in the maze, watch out for big meanies. The screen graphics are excellent, however, except for a few minor complaints, *Midnite Mason* is a winner.

HCM

INDUSTRY WATCH

ONE YEAR AFTER THE LUBBOCK FIRE—IT'S DECISION TIME

Many consumers who purchased the TI-99/4A at a fire-sale price one year ago (when TI exited the consumer-computer business) are now making decisions regarding the fate of their entry machines. Many consoles have already been placed in new locations—collecting dust in attics and closets. Large numbers, however, are still being well-utilized with a collection of software cartridges and tape cassettes for home productivity, education, and entertainment. With the bulk of the “closeting” now past, a fairly large percentage—approaching 40% at last survey—are “adding on” in two ways: (1) purchasing a second brand of computer, and/or (2) buying up additional memory, printers, and disk drives—even spare TI-99/4A's (yes, there are some still out there) for existing systems. One result is a marked increase in disk software sales for this user base.

SOFTWARE COMPATIBILITY—THAT IS THE C-64 QUESTION

Although the new machine was announced last January, Commodore's Plus/4 computer has just recently been shipped to retail outlets around the country. The computer, priced at about \$300, offers word processing, spreadsheet, graphics, and file manager programs built into its ROM. Commodore, however, is reportedly wavering on whether to make the Plus/4 software-compatible with its C-64 model—a machine which has spawned a large software base, and which will continue to spawn more software as Commodore keeps the machine alive with an expected 128K memory expansion. The off-the-shelf Plus/4 does not run C-64 software, but Commodore may be providing an add-on board if the demand for compatibility appears strong. When and how they will decide on adding the board is not known, but this “wait and see” approach to compatibility may signal a change in Commodore's tradition of making their machines incompatible with each other. Meanwhile, it appears that the Plus/4's little brother, the recently introduced Commodore 16, will have less impact in the U.S than in Europe, where it is expected to give Sinclair's Spectrum a run for its money.

PRICES ON TI CARTS TAKE A TUMBLE

In an effort to clear remaining inventory from retail channels—so as to avoid having to take back unsold inventory—TI has dropped their billed-invoice prices on most software cartridges. Some popular cartridges such as Parsec, Personal Real Estate, and Beginning Grammar, available from mail-order sources, are going for \$4.95, while other titles like MicroSurgeon and Typing Tutor are being offered at under \$10. As sources of TI's Extended BASIC cartridges have all but dried up, other manufacturers have stepped in. Some cartridges are licensed versions of the TI language, while others have a few differences and are not compatible with some existing TI Extended BASIC software.

TEST-DRIVEN A MAC YET? HOP ABOARD THE APPLE BUS

The Apple Bus—the network that will connect the Macintosh, IIe, and Lisa computers—is expected to be available in the first quarter of 1985. Originally announced with the Macintosh in January of 1984, development has been slow. According to sources at Apple, a laser printer with two megabytes of memory and a hard-disk file server are among the first products that will be released for this network. It is not certain, however, how long the Lisa will remain in the picture—the new Apple Bus does not require it to function. And, with a bevy of soon-to-be-released new products designed for the 512K Fat Mac—rather than Lisa—such as Lotus Development Corp.'s new integrated software package, Jazz (\$595 including a word processor, spreadsheet, data base manager, graphics, and communications), plus an internal hard disk for a “fatter” Macintosh expected to be released sometime in 1985, we may actually see this “pioneer lady” phased out in coming months. Apple's current “Test Drive a Macintosh” campaign—in which anyone with a valid major credit card can take a Macintosh home for one night of fun and frolic—may prove to be the nails in the coffin for the more-expensive Lisa architecture.

$$24 \div 6 = ?$$



Division Tutor

by Steve Lisonbee

and the HCM Staff

Divide and Compute! This is the modern way to conquer the ancient practice of arithmetic.

Computers and calculators will never make good ol' arithmetic obsolete. We need to hang on to this ancient but sophisticated skill; it's something we can carry around in our heads, and with which we can accomplish many marvelous things. In fact, computers can help us to preserve this basic "r" (one of the "big three") by patiently exercising that biological computer—the human brain.

Past issues of *Home Computer Magazine* have included programs designed to teach addition, subtraction, and multiplication. Now we take on division—an arithmetic function that follows logically from the other three.

Division Tutor consists of two separate sections: Flashcard Division and Long Division. The first section presents simple problems with whole-number answers. The second section provides more difficult problems, to be worked out in "long-hand," which usually leave a remainder. Previous experience with subtraction and multiplication is necessary to perform these division exercises. *Division Tutor* is designed for the 8-14 age group, though certainly anyone who wants to practice their division skills may use it.

Flashcard Division

This section of the program simply displays one problem at a time and asks for a solution. Students have two chances to answer correctly. A prompt informs them of any wrong answers. After a second wrong answer, the right answer will appear, accompanied by a descending tone. If a correct answer is given, the message VERY GOOD WORK appears, along with a short musical "reward."

At the top of the screen, the numbers of right and wrong answers are displayed and updated after every problem. Near these numbers, the percentage of right answers—or "score"—is also displayed.

These flashcard exercises continue until the student chooses to return to the Main Menu.

Long Division

This section of the program takes the student step-by-step through the process of solving a problem in long division. The screen graphically depicts a

schoolroom, with a problem displayed on a chalkboard. Above the chalkboard is a list of each step of the long-division process. An arrow points to the step currently being worked on. The list of steps includes:

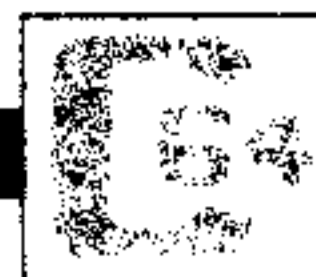
DIVIDE
MULTIPLY
SUBTRACT
BRING DOWN
REMAINDER

"... computers can help us to preserve this basic 'r' (one of the 'big three') by patiently exercising that biological computer—the human brain."

This part of *Division Tutor* takes the student through each step of the operation, just as the problem would be solved on paper. With helpful prompts and graphic aids, the program breaks down a complex problem into simple elements. Students enter their answers, and the program displays the numbers in the proper places. Visual aids show the student where to "bring down" numbers or place remainders.

For each step, a simple equation of the specific part of the problem being worked on is also displayed on the chalkboard, near the main problem. At the same time, the program checks for and informs the student of any wrong answers within this step. Each step allows three tries—after the third wrong answer, the program furnishes the correct answer, and then proceeds to the next step.

In the Long Division section are two difficulty levels: Beginning and Advanced. Exercises on the first level generally have single-number divisors and comprise more steps, each one relatively easy to solve. On the second level, problems more often use double-digit divisors and require fewer (but more difficult) steps to arrive at a solution. As in the Flashcard section, the



program continues to present exercises until the student chooses to return to the Main Menu.

Teaching

Of course, these and other exercises in division can also be worked out using only a stick or a piece of charcoal—this is the beauty of simple arithmetic. Stone Age man, with the right knowledge, could do as much. But computers can quickly tell you when you're right or wrong, which is what makes them an ideal teaching device, especially with a program like *Division Tutor*.

For your key-in listing see HCM PROGRAM LISTINGS Contents on page 85.



CONTROL CAPSULE *Division Tutor*

KEY	FUNCTION
1	Select Flashcard Division.
2	Select Long Division.
3	Exit the program.
Esc	Return to the Main Menu.

One of the features that makes this program so much fun and easy to use is the way the Long Division option interacts with the user. The program prompts the user to enter the answer at the same location within the problem, as if the user were solving with a pencil and paper. In addition, the user can only enter enough numbers to satisfy the problem. It is impossible to enter 20 numbers to a problem with an answer only 1 or 2 digits long.

The Applesoft routine that handles this input is between lines 3070 and 3410. This routine performs limit checking on each key as it is entered. In addition, it checks for the (Esc) key (ASCII 27). If the (Esc) key is encountered, the input routine will terminate. The calling routine must then check the input for the (Esc) character so that it can return to its calling routine—generally the Main Menu.

The routine also limits the user in the number of keys which can be pressed in response to an answer. Because of the strict formatting of the equation on the screen, and the user's ability to enter values right into the equation, it is necessary to limit the number of keys which can be pressed. The maximum length for any input in the routine is three (3) characters. After entering three characters, the routine will exit automatically using the characters you've entered as your answer. You can enter a value with less than three digits simply by pressing (RETURN).



CONTROL CAPSULE *Division Tutor*

KEY	FUNCTION
F1	Select Flashcard Division.
F3	Select Long Division.
F5	Exit the program from the menu or return to menu from division routines.

The *Commodore Programmer's Manual* contains a comprehensive list of addresses used by the BASIC interpreter to do certain functions. One address (location 783) is used in this program to clear the Status register before calling a machine routine. This address was mistakenly labeled in the manual as the SP (Stack Pointer) instead of the ST register.

Two other addresses are used to load the X and Y registers. These three addresses are used in this program to locate the cursor on the screen without using the PRINT statement. (See the "Home Computer Tech Note" for the Commodore 64 in this issue.) These addresses are not the actual registers in the processor—if they were, you would really mess things up by POKEing values into them at unpredictable times. Instead, when you call an assembly language program from BASIC, the system loads the values from those addresses into the the X, Y and Status registers before entering the routine.



The beginning of a long-division problem as seen on the C-64.



CONTROL CAPSULE *Division Tutor*

KEY	FUNCTION
1	Select Flashcard Division.
2	Select Long Division.
3	Exit program.
0	Return to Main Menu if entered as first response to new problem.

The IBM machines are very versatile computers, even in screen mode 0 (text mode). They have a number of useful graphics characters built into the ASCII character set. By making use of these, you gain two advantages: your program will display these special characters much faster than it could draw graphics on the graphics screens, and you have access to all sixteen colors for drawing.

A good use of this technique is demonstrated in the long-division routine. The computer generates a red brick background, complete with a black chalkboard and a white pull-down chart with the long division steps listed on it.

The brick background is created by using ASCII characters 193 and 194. These characters are placed in two strings (B\$(0) and B\$(1)), and then displayed on the screen:

```
FOR Z = 1 TO 20: B$(0) = B$(0) + CHR$(193) + CHR$(194): B$(1) = B$(1) + CHR$(194) + CHR$(193): NEXT Z
COLOR 0,4,4: FOR Z = 1 TO 12: PRINT B$(0);: PRINT B$(1);: NEXT
```



CONTROL CAPSULE
Division Tutor

KEY	FUNCTION
1	Select Flashcard Division.
2	Select Long Division.
3	Exit the program.
0	Return to Main Menu if 0 is the first response to a new problem.

XB Division Tutor requires TI Extended Basic.

The Long Division option displays a schoolroom wall complete with a red brick background, chalkboard, and pull-down chart showing the various steps in solving the equation.

Let's take a look at how easy it is on the TI-99/4A to create the background in the schoolroom. Two shapes are needed to create the staggered brick effect. We choose ASCII characters 104 (h) and 105 (i). Each character is assigned a pattern for one-half of a brick.

By staggering the characters on the screen to create a pattern, as shown below, you can create the brick pattern used for the wall:

```

hiiiiihiiiiihiiiiihiiiiihiiii
iiiiihiiiiihiiiiihiiiiihiiii

```

Because of the large number of graphics characters needed to define the schoolroom, it is necessary to use some of the characters in color group #8. This group includes the letters X, Y, and Z. It is necessary to use the letter Y in several of the messages displayed on the screen, which presents a problem—one character is a different color than the others. We solve this problem by substituting the letter Y for the @ symbol. This is done by getting the pattern for the letter Y with the CHARPAT statement. This statement returns the hexadecimal code used to define graphics characters. That pattern is then assigned to ASCII character 64, the @ symbol.

So, if you see text in this listing that looks like it should have had a Y in it, but you see @ instead, it's not a typo, it's there for a reason. This is the code that makes the transfer:

```

CALL CHARPAT(89,Y$)
CALL CHAR(64,Y$)

```

Division Tutor (TI-99/4A)
Explanation of the Program

Line Nos.	Explanation
100-170	Program header.
180-330	Initialize program graphics and display the title screen.
340-430	Main Menu screen.
440-460	Flashcard Division title screen.
470-510	Long Division menu screen.
520-670	Flashcard routine.
680-740	Check Flashcard answer.
750-850	Set up Long Division screen.
860-940	Select and display a problem.
950-1160	Main control loop.
1170-1250	Display the remainder. Get set up for the next problem.
1260-1310	Graphics display routines.
1320-1340	Incorrect answer routine.
1350-1430	Give the correct answer.
1440-1450	End of program message.
1460	Routine to change colors.
1470-1490	Program data.

Division Tutor (C-64)
Explanation of the Program

Line Nos.	Explanation
100-190	Program header.
200-240	Initialize program and display the title screen.
250-340	Main Menu screen. Select operation.
350-400	Initialize Long Division routine.
410-780	Flashcard Division routine.
790-1000	Display Long Division screen.
1010-2100	Main control loop for Long Division.
2110-2160	Routine to locate the cursor.
2170-2350	Input routine.
2360-2480	Incorrect answer.
2490-2590	Display the correct answer.
2600	Exit routine.

Division Tutor (Apple II Family)
Explanation of the Program

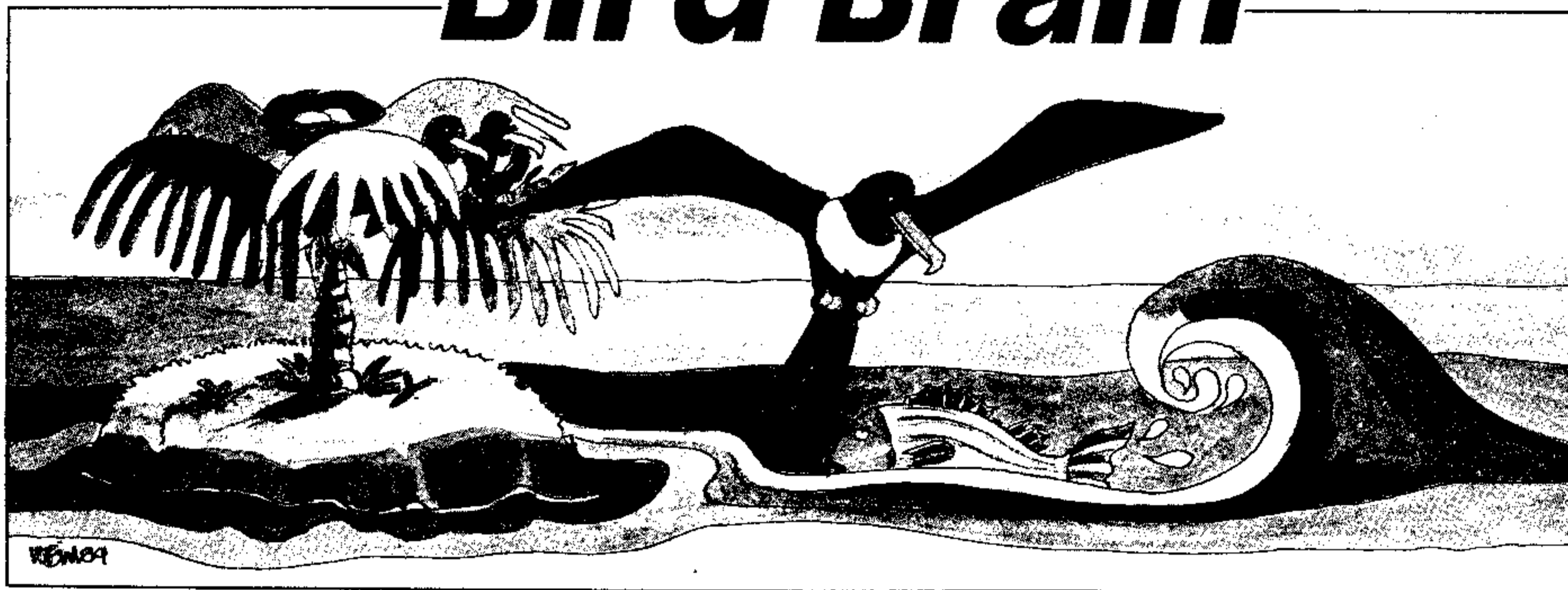
Line Nos.	Explanation
100-180	Program header.
190-270	Master control loop to set up and run the program.
280-440	Initialization routine.
450-570	Main Menu screen.
580-730	Menu screen graphics routines.
740-1250	Flashcard Division.
1260-1530	Long Division control loop.
1540-1730	Enter Long Division level of difficulty.
1740-1900	Display Long Division background.
1910-2000	Long Division subroutines.
2010-2690	Work the problem.
2700-2880	Display remainder.
2890-2910	Time delay loop.
2920-3060	Option to continue with Long Division or exit.
3070-3230	Input routine.
3240-3410	Read a character with a prompt.
3420-3460	Read a character.
3470-3500	Error sound.
3510-3620	Correct sound.
3630-3700	More sound effects.

Division Tutor (IBM PC, PCjr)
Explanation of the Program

Line Nos.	Explanation
100-200	Program header.
210-250	Initialize program.
260-280	Display the title screen.
290-310	Main Menu screen.
320-420	Flashcard Division routine.
430-500	Display the Long Division screen.
510-680	Main control loop for the Long Division problem.
690-720	Display the remainder.
730-740	Incorrect answer routine.
750-800	Display correct answer.
810-840	Display routines for small equation.
850-870	Display routine for the menu and borders.
880-910	Input routine for integer numbers.
920-930	Music for the correct answer.
940-970	Display the arrow in the Long Division routine.

HCM

Bird Brain



*You may be great at fishing, but how good are you on the wing?
Learn some new angles from this flying fish-catcher—
but keep your feathers dry!*

by Craig Blazakis
and the HCM Staff

Watch that wave, Bird Brain! Catch that fish and fly! Here in this tropical clime, the endless surf syncopates with the rhythm of survival. Mother Frigate Bird—spying a plump, juicy prey just beneath the surface—dips down once more and times her catch to avoid the next onrushing wave. Then it's back to the nest for a quick meal and a short rest on her sun-warmed eggs before she tries again. But this time—if she makes only a slight miscalculation—her dive may end in a trip to Davy Jones' locker, as a powerful wave rears up and slaps her to the bottom. Luckily for her precious eggs, two more birds wait near the nest to take her place. And if they should meet the same fate? Well, in this case, birds, fish, and tropical beach all exist inside a computer—where reincarnation is part of the program.

Fly Fishing

At first, Bird Brain may sound easy—a real frigate bird will usually snatch a fish without even wetting a feather. But no fish is caught without effort, especially the fish in this computer game. Control the bird with joystick or keyboard. To flap her wings and give her lift, press either the fire button on the joystick or the spacebar on the keyboard. To move her left or right, move the joystick in the desired direction, or press the S key (for left), or the D key (for right). To really affect the bird's direction, press each key repeatedly, rather than pressing a key once and holding it down. This bird is subject to gravity—once she has left the nest, she must flap her wings to gain or maintain altitude. Without frequent flapping, she will plummet into the rough waters.

To make matters worse, large waves periodically travel from left to right across the screen. If the bird hits a wave, it will be knocked about, losing any control it might have had. Of course the fish, which swims along just under the surface of the water, is unaffected by the waves.

Each bird has a time limit in which it must catch a fish and return to the nest (as a real bird might exhaust its energy and fail to return). A bar across the bottom of the screen grows shorter as this time is used up. If the time limit is reached, the bird will die and take a trip to video heaven. When the bird successfully delivers a fish to the nest, the timer resets.

Of course, if the bird loses control and hits the water—through either some bird-brained maneuver, or a direct hit from a wave—it will perish.

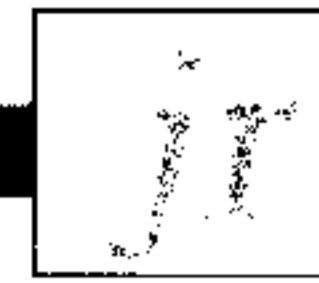
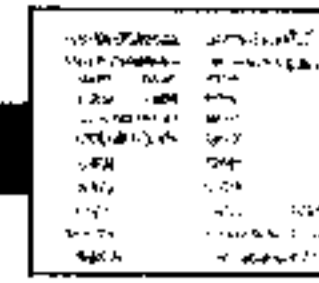
A Heavy Load

After picking up a fish and heading for home, the bird must work harder to carry the load. If she catches the fish while descending, no amount of furious flapping will

“. . . in this case, birds, fish and tropical beach all exist inside a computer—where reincarnation is part of the program.”

suffice to overcome her downward momentum, and she will meet a watery grave. A better method is to hover at the same level as the fish, and then start climbing as you grab it. (All that is necessary to actually “grab” the fish is to position the bird on the fish itself.)

You start the game with three birds—one in the nest and two on a branch. When all three birds have died, the game ends. But of course, you can always begin again. Scoring is based on accumulated time (in seconds) left on the clock for each fish eaten, which is then multiplied by the skill level (1, 2, or 3) for a total number of points. Your score is computed and displayed at the end of the game.



Bird Brain has three skill levels. As the skill level increases, you will find it more difficult to catch a fish and make it back to the nest. The higher skill levels have the following effects: you will have less time to catch the fish; the bird will weigh more, making it more difficult to stay in the air; and the wave will move faster.

For your Key-in listing see HCM PROGRAM LISTINGS Contents on page 85.

CONTROL CAPSULE	
<i>Bird Brain</i>	
KEYBOARD	
Space Bar	Flap wings
S	Move bird left
D	Move bird right
JOYSTICK	
Fire Button	Flap wings
Stick left	Move bird left
Stick right	Move bird right



The TI version of *Bird Brain* makes use of the best features of the TI-99/4A home computer. Sprites provide an ideal medium for the graphics animation in this program. The ability to set a sprite in motion and let it glide across the screen makes the bird in this game appear to be really flying. In addition, animation is added by changing the sprite's shape. Two sprite shapes are used for the bird while it's in flight, and a third shape for the bird at rest. The two shapes used for the bird create the animation of the bird flapping its wings.

Controlling the velocity of the bird in any direction is a simple matter. Two variables, U and S, contain the X and Y velocity. Gravity's affect on the bird is accomplished by decreasing the bird's upward motion with every pass through the program. If the variable U is decreased to the point of being a negative number, the bird will begin traveling downward.

The joystick and keyboard are checked to see if you have tried to flap the bird's wings. If you have, then L is added to U. L contains the amount of lift the bird gets for each flap. The initial value of L is determined by your playing level. The table below illustrates the lift and fall speeds for each of the 3 levels.

LEVEL	L	M	TIME
1	4	1	160
2	3	1.8	128
3	2	2	96

L = Rate of lift
M = Rate of fall
TIME = Time limit



This program proves that the Apple II is still a powerful graphics machine. Most programmers shy away

from creating animation on the Apple using Applesoft BASIC because of its limitations in speed, and the cumbersome shape tables it requires. This version of *Bird Brain*, however, may change their minds.

Only two shapes are required to create the bird's animated flight, shapes 7 and 8 in the shape table. The current shape of the bird is contained in the variable BS. Because the bird may change its shape while flapping its wings, the variable SD keeps track of the bird's previous shape. In doing this, it's a simple matter of using the XDRAW command to erase the bird at its old location, and redraw it at its new location.

To give the bird motion, two variables containing the bird's X and Y velocity are added to BX and BY. These two variables, U for up/down, and S for sideways, are adjusted by your actions to move the bird back and forth and by flapping the wings.

The difficulty level you choose makes a major difference in this program's playing ease. Several key factors are involved, as listed below:

LEVEL	L	M	TIME
1	4	1	160
2	3	1.8	128
3	2	2	96

L = Rate of lift
M = Rate of fall
TIME = Time limit



Both IBM machines are extremely powerful graphics computers because of three versatile commands—DRAW, GET, and PUT.

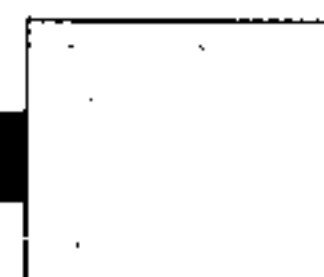
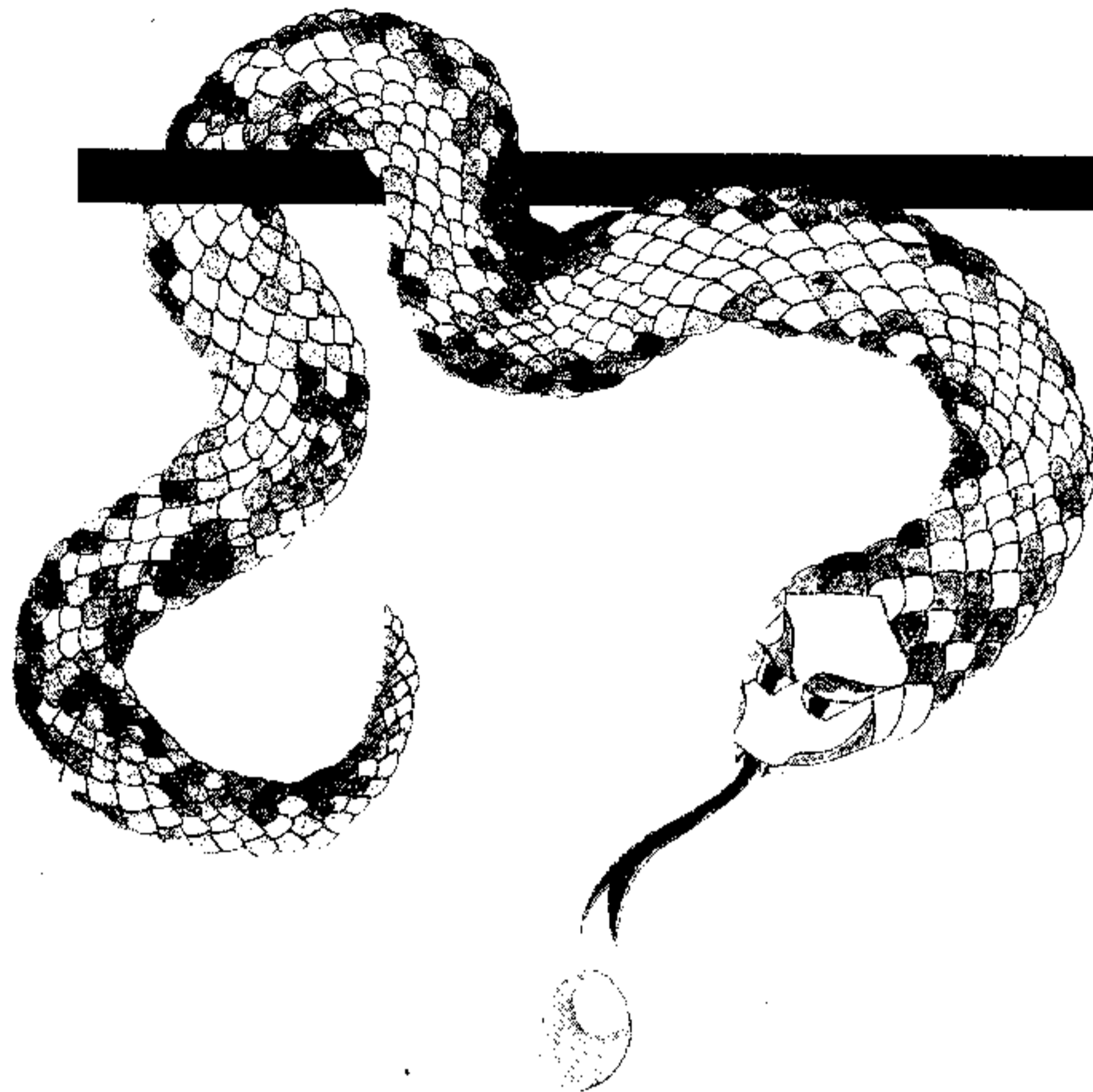
The DRAW command allows you to create shapes by drawing on the screen the same as you would on a piece of paper with a pen: simply tell the computer in which direction and how far to move the pen.

Animation is made possible with the GET and PUT statements. Once you have a shape drawn on the screen, you can save it in an array with a GET statement. Here, the two shapes for the bird as well as the shapes for the wave and the fish are all stored this way. Because these shapes are not moved automatically, all of the action stops when the birds is at rest in the nest. The wave and the fish will resume their motion when the bird leaves the nest.

Once an image is stored in an array with the GET statement, you can place that image anywhere on the screen with a PUT statement. Options in the PUT statement allow you to mix the image you place on the screen with existing graphics in a number of ways. If you XOR the image to the screen, as is done in this program, the image reverses the color of any pixel on the screen that comes in contact with a pixel in the image. By writing the same image to the screen in the same location twice, you erase the first image created, and return the screen to its original state before the image was placed on the screen.

LEVEL	LIFT	FALL	TIME
1	1.5	.18	862
2	1.0	.26	602
3	0.8	.35	456

LIFT = Rate of lift
FALL = Rate of fall
TIME = Time limit



SLITHER

By Aaron Chew
and the HCM Staff

Ooooooh! It's slimy! And it slithers! Yes, folks, it's a slimy, slithery, egg-eating snake on the loose—and it's growing with every bite. Just don't let it bite (or even touch) its tail, or it will literally be "the end."

Slither is a simple game, but one that has a way of "winding on." You may be surprised at just how involving and difficult it can be. On the first screen—nearly bare—the snake is quite short. Making it capture the first few eggs is an easy matter. But each time it eats one of these free-rolling eggs, the snake becomes a little longer. Finally, it becomes long enough to get in its own way—which, in this scenario, means sudden death. You must keep your snake from running into anything but the eggs themselves, including the sides of the box it is slithering around in.

Only two keys are used to control the snake's direction: **G** and **H**. The **G** key will turn the snake to the left, while **H** turns it to the right. This left/right action is from the snake's perspective, not the player's. Thus, if the snake is moving down the screen and you press the **G** key (for left), the snake will move to its left (your right).

Complications

Slither includes several levels of difficulty. You always start the game at level 1, with three tries (snakes). Every time you gain 500 points, you will advance to another level and add an extra snake to your "nest." Each egg is worth 20 points. If all of your snakes are wiped out, the game will end, with the option to replay.

At each higher level, the screens become more complicated. On the first level, you have only the four outside walls to contend with. Another inner wall appears in level 2. This wall is only two snake-widths from the outer wall and has only one entrance. On each new level, another wall appears—each one two snake-widths from the previous inner wall. Entrances to each inner wall are staggered top to bottom. If you were challenged by this simple game in the beginning, imagine how you will interact as it becomes more complicated.

Although *Slither* is simple in concept, it will not slide right by you. You may spend more hours mastering this game than you do some others with fancier plots and less humble beginnings.

For your Key-in listing see HCM PROGRAM LISTINGS Contents on page 85.

CONTROL CAPSULE

Slither

KEY	FUNCTION
G	Turn snake to its left.
H	Turn snake to its right.

The Commodore version of *Slither* has four levels. Each new level adds another wall to the screen. Two arrays used in this program are of primary importance: The **T**() array is used to keep track of where the snake has been. Every time the snake advances, the new location is placed in the array. Every time the snake's tail moves, the cell in the array containing the location of the tail is cleared, and the variable **LT** (which contains the pointer to the tail) is updated. The second array is **B**(). This array is a mirror image of the screen, except that the values stored there are different. This array is used to keep track of obstacles which the snake may run into. Joysticks are a bonus option for Commodore users. To turn the snake left or right, simply move the joystick to the corresponding direction.

The TI version of *Slither* has four levels of difficulty; on each level there are more walls for the snake to maneuver around. This program uses one array to track the progress of the snake. The snake is actually moved by updating the head, and erasing the tail. The index into the array for the head and tail positions are contained in the variables **SP** and **SLA**. The array **SN**() is two-dimensional, and contains the X and Y coordinates for each segment of the snake. Obstacles are easy to check for by using the **GCHAR** command. This command allows you to check the contents of any position on the screen.

HOME COMPUTER™

product news

Each month we publish items of interest and news of recently or soon-to-be released computer products. Our publication of information from manufacturers of computer, peripherals, software, and accessories is not to be construed as product endorsement. Prices quoted are the manufacturers' suggested retail prices and are subject to change.

Send press releases to:

Product News Editor
Home Computer Magazine
1500 Valley River Drive., Suite 250
Eugene, OR 97401

The Great Claus Caper

Christmas Adventures On Disk

Just in time for Christmas is A Christmas Adventure, an adventure program for Commodore 64 and Apple II family computers. Set in and around Santa Claus' ice castle at the North Pole, the player must unravel the mystery of Santa's disappearance only hours before his annual gift-delivery run. An enhanced version of A Christmas Adventure allows users to customize the program to add per-



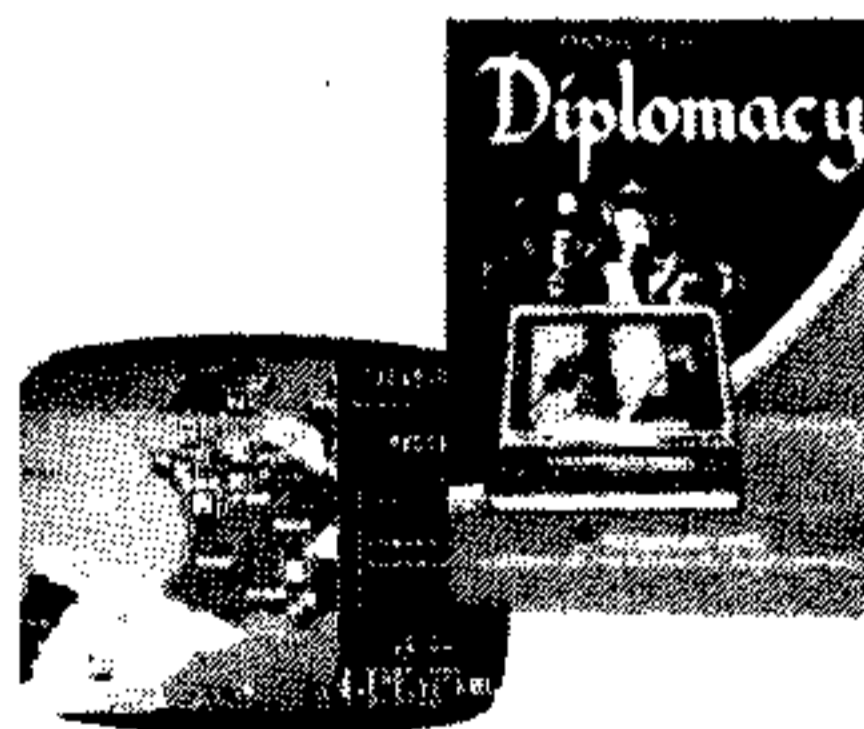
sonal references. The adventure alone (ACA.N) is \$14.95, and the enhanced version (ACA.C) is \$16.95. For \$17.95 a customized version will be prepared by BitCards.

BitCards Inc.
30 West Service Rd.
Champlain, NY 12919
(514) 274-1103

Diskette Diplomacy

Computer Diplomacy Game for the IBM PC

The board game played by Henry Kissinger and David Eisenhower is now here in computer form from the Avalon Hill Game Company. Computer Diplomacy, for the IBM PC, features a full-color screen map of Europe, which from 1 to 7 players divide up as representatives of various countries. When less than 7 people are playing, the computer fills in for the others. Using diplomatic tactics (as well as backstabbing and colluding), a player must occupy half of Europe with his or her



armies and navies to win. Computer Diplomacy retails for \$50, and requires 256K memory, a double-sided disk drive, and a color graphics board.

The Avalon Hill Game Company
4517 Harford Rd.
Baltimore, MD 21214
(301) 254-5300

New TI Application Development Programs

Data Entry, Report Writing, Sketching

Easyware has announced the release of an application development package for the TI-99/4A featuring: Etch, a data dictionary program which allows users to define the parameters for their database through a menu-driven system of screens; Sketcher, a screen painter/ designer which enables users to produce quality screens with

features such as menu and multi-level screens, line drawing, and color defining; Sketch, a data entry program made easy with pattern-matching selection, modification, delete and list features; and Fetch, a report writer with sorting, page headings, sort break footings, and pattern-matching options. The package is priced at \$49.95.

Easyware
P.O. Box 3130 Station D
Ottawa, Ontario K1P 6H7

Double Duty For The Commodore

Text Editor, And Graphics Kit Available

Two new software packages have been introduced by APCAD. TexED is a text editor for the Commodore 64 that functions as both a program editor and a word processor. It features both line-editing and visual-editing modes, a full-screen window, a complete range of editing functions, and a print option which allows document formatting. The TexED cassette is menu-driven and retails for \$19.95. The PLOTVIC for the

unexpanded VIC-20 is a high-resolution graphics kit featuring full screen window, geometric figure and text generation and positioning, eight-element color selection, three-dimensional perspective generation, and hi-res printing capacity. PLOTVIC3 and PLOTVIC8 for the expanded VIC-20 accept lightpen input and have more versatility. All three versions retail for \$19.95 each.

APCAD
P.O. Box 83
Saline, MI 48176

Tune In Tomorrow On TI

Fantasy Game Is A Continuing Saga

A fantasy adventure role-playing system in which the hero appears in a continuous saga on several game disks has been released by Creations Unlimited. Fantasy, for the TI-99/4A, features adventures in an unknown wilderness, mystical castles, magic, murky swamps, and deep caverns.

Purchase of the Fantasy game system includes the first adventure, The Tomb of Gorgoroth, and costs \$29.95. Subsequent story disks (for \$12.95 apiece) include Where Evil Dwells, Assassin's Plot, Dragons of Doom, Curse of Targon, and Secret of Stonekeep. A disk drive and 32K Extended BASIC is required.

Creations Unlimited
20 Tilton Lane
Andover, MA 01810

Old Stories, New Games

Children's Classics Become Text Adventures

A new line of graphic and text adventures based on well-known children's classics will soon be released by Spinnaker for Apple, Commodore, and IBM computers. The Windham Classics series includes *Swiss Family Robinson*, *The Wizard of Oz*, *Gulliver's Travels*, *Below The Root*, *Treasure Island*,

Spinnaker Software Corp.
1 Kendall Square
Cambridge, MA 02139
(617) 494-1200

and *The Wind in the Willows*. The games involve characters and adventures from the above classics, and are intended to encourage players to read the books they're based on, although the books are not required to play the games. Packages retail for \$26.95 each.

Boot Up And Break!

A Dance-Off On The C-64

Break Street, by Creative Software, makes street gymnastics, mime, and showing-off into a computer game for the C-64. Players take on the Stingrays, a neighborhood gang, in a break dance turf battle. Using a keyboard or joystick, players manipulate the dancer through moves such as head spins, the moonwalk, snaking, and the tut. Missing a key sequence move causes a fall. Entire dance sequences may be



strung together, recorded, and replayed in the future. *Break Street* retails for \$24.95.

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

You Can Count On It

Advanced Number Skills Offered For Kids

Skip Counting, a TI-99/4A program to help children build "number sense" by counting, has been introduced by B5 Software. The child selects a number to count by, the beginning and ending number of each counting sequence, and then counts

by 2's, 5's, 10's, 100's, etc. Students may review odd and even numerals, multiplication products, or numeration concepts. Lessons end with a song and a graphic reward. *Skip Counting* is \$16.95, and requires Extended BASIC.

B5 Software
1024 Bainbridge Pl.
Columbus, OH 43228
(614) 276-2752

School Subjects Hit Home

History, Biology, French Offered

The second generation of software programs in American Educational Computer's Matchmaker series (for the IBM PC and PCjr, Apple II family, and Commodore 64) has been released. The company claims that the new titles—U.S. Government, World History, Biology, French, and Science I, II, and III—include most subjects in standard elementary and junior high school curricula. The programs use a variety of standard quiz formats to test a student's knowledge in the various subjects. Parents may also add their own material to the programs. Visual



awards and game play are offered to the student after successful completion of a question series. Each grade level program retails for \$39.95.

American Educational Computer, Inc.
2450 Embarcadero Way
Palo Alto, CA 94303
(415) 494-2021

Big Blue Belts Out A Few

IBM Rolls Out 10 New Home Programs

IBM has introduced 7 new entertainment programs and 3 educational programs for their PC and PCjr computers. The entertainment programs include *Trivia 101: The Introductory Course*, and *TV and Cinema 101*, each with more than 5,000 questions; *PC Pool Challenges*, a simulation of straight pool and eight ball billiard games; *Touchdown Football*, a football simulation program that is only available for the PCjr; *Jumpman* and *Shamus*, two arcade games; and *Zyll*, a text adventure game set in the medieval ages. *Jumpman*, *Shamus*,

and *Zyll* have a suggested retail price of \$35, and the others, \$30.

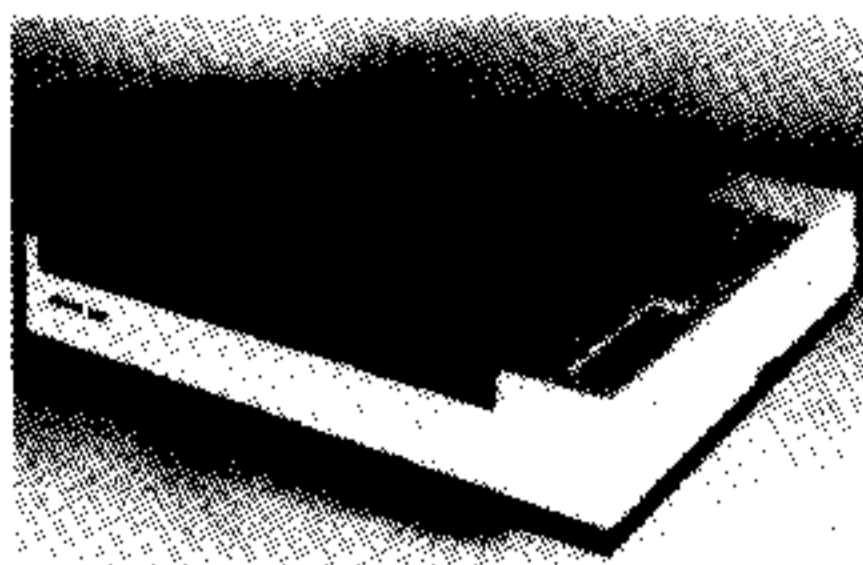
The educational programs are part of IBM's *Electric Literature Series*, based the new *Electric Poet* program. *Electric Poet* (\$75) gives teachers and parents the ability to create animated, musical lessons for language arts, social studies, math, science, or other topics. *Comma Cat* and *Dictionary Dog*, \$45 each, were designed using *Electric Poet* and use its interactive teaching capabilities to teach the use of punctuation marks and the dictionary, respectively.

IBM Entry Systems Division
P.O. Box 1328
Boca Raton, FL 33432
(305) 982-3474

A Legend In Printing

New Printer Has Square Dots

Cal-Abco has announced the Legend 880 dot-matrix printer, which uses a new square dot to produce high print quality. The printer prints 80-column lines at 80 characters per second, and offers over 40 different character fonts, mixable on a single line. It is bi-directional and comes with an 8-bit standard Centronics Parallel interface. In addition, the Legend 880 generates 228 ASCII characters and



high-resolution graphics with a 9-wire print head warranted for 50 million characters. The printer retails for \$279.

Cal-Abco
14722 Oxnard St.
Van Nuys, CA 91401
(818) 994-0909

Precise Calculations

A Professional Equation-Solver

A program that interactively solves science, engineering, and business equations is available from Interactive Microware, Inc. Varicalc, for the Apple II family of computers, will solve for any one of 19 variables on the right or

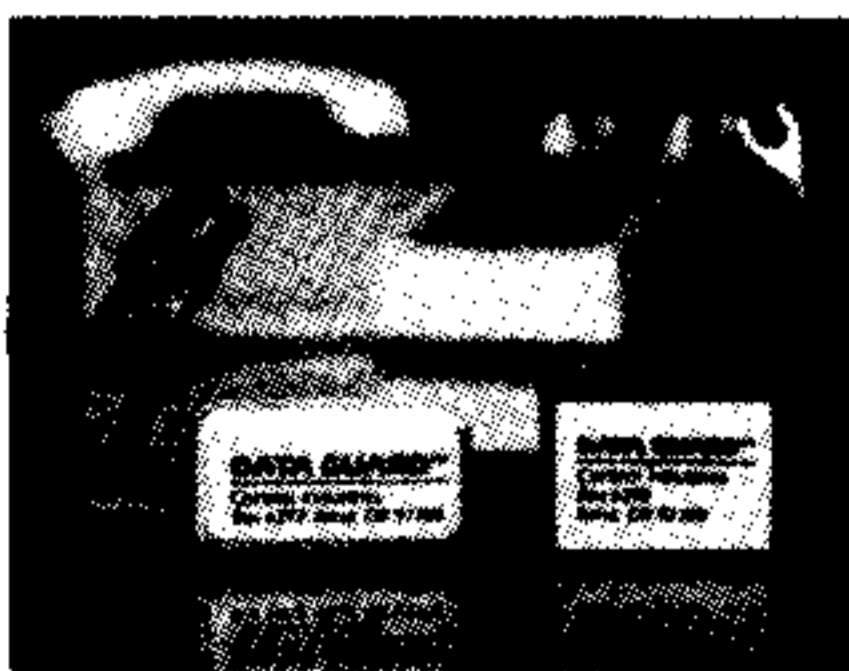
left side of a formula without rearranging the formula. Results can be plotted on the screen or printed, and up to 255 equations may be stored on disk for recall. Varicalc is priced at \$100.

Interactive Microware, Inc.
P.O. Box 139
State College, PA 16804-0139
(814) 238-8294

Guarding Against Modem Hangups

Set Up An Automatic Dedicated Line

Dataguard will prevent accidental data loss or disconnection of modem users by setting up an automatic dedicated line. Control Industries' product gives the modem user priority on the line when another phone is picked up, without disruption of normal telephone functions. Dataguard comes as an In-Phone model, or as a 12-foot Snap-In-Cord model which replaces your present



phone cord. It is FCC approved and retails for \$39.95.

Control Industries
Box 6292
Bend, OR 97708
(503) 389-1969

Watch Out For That . . .

Hole in the Ground Is First Of Adventures

Tesseract Software has announced the first in their series of interactive graphic adventures for the TI-99/4A. The first program, Hole in the Ground, is programmed in TI BASIC and includes a recorded oral explanation and set-up of the game. It is priced at \$15.

Tesseract Software
701 Park Lane
Derby, KS 67037



Pop Goes The IBM

Desk Tools Available For The PC/PCjr

Bellsoft has introduced Pop-Up desk tools, a series of programs that, once loaded, can be accessed with one keystroke while the user is running other applications. The first six Pop-Ups, which run on the IBM PC and PCjr, are Calculator, Notepad, TeleComm, Alarm Clock, Calendar, and PopDOS.

Pop-Up Calendar, like the other programs, pops up in its own window at any time, even when users are running other applications. It displays 3 months at a time for any year, or one month with holidays and appointments marked. It retails for \$19.95.

PopDOS allows users to access DOS commands at any time. Users can look at a directory, copy, rename, delete, or print files in a choice of typefaces, as well as check on how much memory is available. It costs \$39.95.

Pop-Up Notepad, also \$39.95, is a scratchpad program that lets users jot down quick notes, create lists, and keep track of things. It moves information between applications,

Bellsoft, Inc.
2820 Northup Way
Bellevue, WA 98004
(206) 828-7282

and has the same editing features of a word processor. Notepad notes can be stored as permanent files.

Pop-Up Calculator, \$39.95, works like a real calculator, operates on values in up to 10 memories, and displays results that can be printed. It will also pass the results of its calculations to the user's other applications.

With Pop-Up TeleComm, users can dial their most-often-called phone numbers automatically, enter them at the keyboard, or take them from a phone list or database. Users can instantly connect to their choice of information services, bulletin boards or other computers, and receive and transmit information without leaving the program they're running. TeleComm is \$79.95.

Pop-Up Alarm Clock displays the time, sets alarms with reminder messages, and lets users run programs when they're away from the computer. It has a stopwatch, and copies are being offered free to IBM users as a promotion.

Running On Adventure

Escape The Kryon Empire

EB Software has announced TI Runner, a program written in assembly language with music, a demo mode, and graphics/animation. As the TI Runner, the player is a highly trained commando who has been captured and imprisoned in the Kryon

Empire. The TI Runner must conquer 50 levels while avoiding guards and collecting treasure on the way to the surface and freedom. The game retails for \$24.95 and requires an Editor/Assembler or Mini-Memory module, and 32K memory expansion.

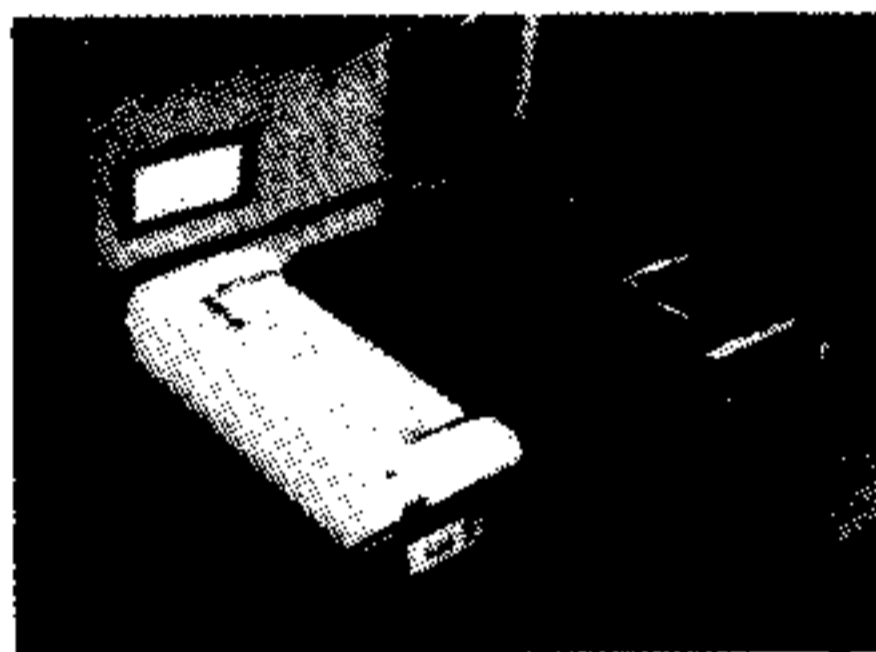
EB Software
12912 Villa Rose Dr.
Santa Ana, CA 92705



Shh! It's The HUSH 80

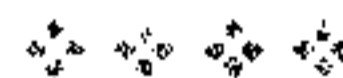
A Lightweight, Low Cost Printer

An 80-column, 28-ounce, dot-matrix thermal printer for \$159.99 has been announced by Ergo Systems, Inc. The HUSH 80 portable thermal printer features 80-column, bidirectional printing at 80 characters per second, and graphics at 4800 dots per square inch. It comes in three models, each of which can be equipped with a built-in rechargeable nickel-cadmium battery pack. The HUSH 80CD provides direct interfacing to the Commodore line of computers, the HUSH 80P has a Centronics-type parallel



interface, and the HUSH 80S provides a serial RS232 interface. All models include the interface, interface cable, 100-foot roll of thermal paper, and a 9-volt a.c. wall transformer with power cable, and will fit into a conventional briefcase.

Ergo Systems, Inc.
1360 Willow Rd.
Menlo Park, CA 94025
(415) 322-ERGO



Keep Your Dungeons To Yourself

An Integrated Game-Creation Package

New for TI-99/4A game-players is Dunjon Creator, a package of three integrated programs: Dunjon Definition, a character editor which gives the user complete control over the color and definition of monsters and other features; Dunjon Maker, which allows a floorplan to be designed

using the defined characters; and Dunjon Play, which may then be used to form a user-defined world and move a hero through it. The game requires Extended BASIC, and retails for \$17.95 for the cassette version, and \$19.95 for the disk version.

Phoenix Computer Enterprises
8 Jay Circle
Windsor, CT 06095



At The Tone, Daylight Time Will Be . . .

Card Adds Clock, Calendar, Print Spooler

Legacy Technologies, Ltd. has introduced The Legacy CPS multi-function expansion card for the IBM PCjr L-Bus. The card adds a clock, calendar, and parallel printer port to the PCjr through the Legacy expansion system. When the system is off, a battery powers the clock/calendar. Software that comes with the card allows the user to automatically set time and date, set system prompt with time,



and print spooler. The print spooler allows the CPU to be "slaved" for printing functions so that the user can perform 2 functions concurrently on the Junior.

Legacy Technologies, Ltd.
4817 North 56th St.
Lincoln, NE 68504
(800) 228-7257



A Solar Shower Of "Edware"

Sunburst Adds 3 To Educational Line

Sunburst Communications, Inc. has added three programs to its home educational software line. The Incredible Laboratory, for ages 7 to adult, uses trial and error and note-taking skills to discover what combinations of mysterious chemicals make up crazy monsters. Challenge Math, for ages 6 to 11, contains three programs that let

children practice basic math, estimation, and problem-solving skills. Getting Ready to Read and Add gives preschoolers a chance to practice letter and number recognition. Each program is \$39.95, and they are all available for Apple II systems. Challenge Math is also available on the Commodore 64.

Sunburst Communications Inc.
Pleasantville, NY 10570
(800) 431-6616



Achtung! Some WW II For Big Blues

Castle Wolfenstein Now Out For PC/PCjr

Following three years of popularity on other systems, Castle Wolfenstein by Muse is now available for the IBM PC and PCjr. The player assumes the role of a World War II G.I. who has been captured behind

enemy lines and taken to an ancient castle to face interrogation and death. The G.I. must find the Nazi war plans, battle guards, and escape from the fortress. Castle Wolfenstein costs \$29.95 retail.

Muse Software
347 North Charles St.
Baltimore, MD 21201
(301) 659-7212



TECH NOTES



Doing Without Extended BASIC "ACCEPT AT"

A very useful feature missing from TI BASIC is the ability to accept input from any location on the screen. The **INPUT** statement only allows information to be typed at the bottom of the screen. The TI Tech Note of the Vol. 4, No. 2 issue of HCM showed, among other things, how to display information on the screen at any location. Here is a routine that will input information from any line on the screen.

The only drawback of this routine is its absence of sophisticated editing keys, which are available when using the **INPUT** statement. However, the left and right cursor control keys move the cursor left and right anywhere on the line, so you can type over mistakes. This subroutine is a one-line input routine, meaning that you can't type in more than one line at a time without recalling the routine again.

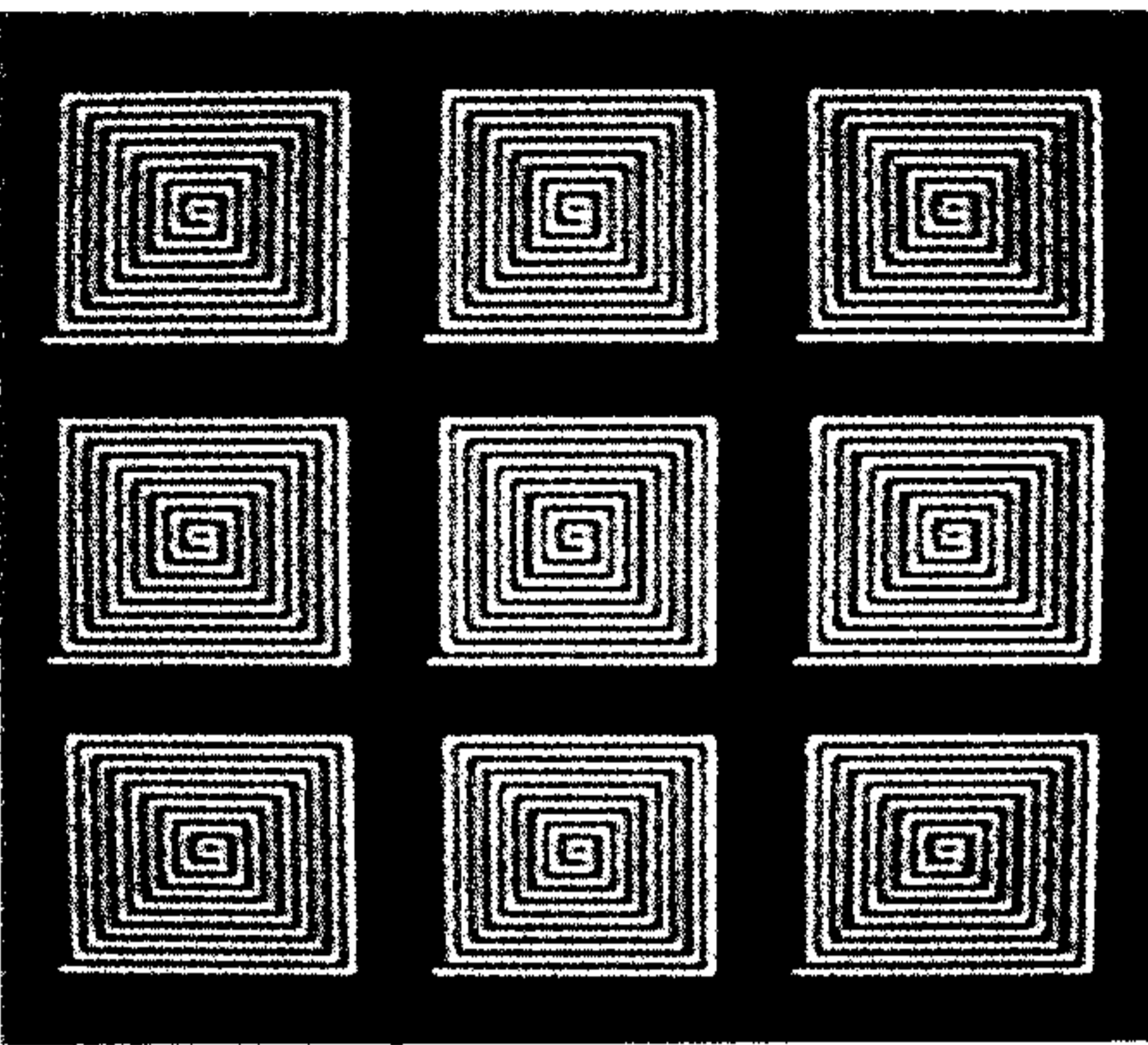
To use this routine, set up the value of **XP** to indicate the input line. Branching to the subroutine without assigning a value to **XP** will produce an error message. To terminate input, simply press **[ENTER]**—the input will be returned in **KS**. To illustrate how this routine can be used we have supplied a short demo in the beginning of the program. The demo will clear the screen, set the row pointer **XP** to 12, and branch to the Input routine. After coming back from the routine, the line entered will be printed, and the program will halt.

```

100  REM * DEMO FOR INPUT ROUTINE *
110  CALL CLEAR
120  XP=12
130  GOSUB 10000
140  PRINT KS
150  END
160  REM * INPUT ROUTINE *
10000 CALL CC$=""
10010 CALL CP$=""
10020 CALL CR$=""
10030 CALL GCHAR(XP,CP+2,C)
10040 CALL KEY(0,K,S)
10050 CALL ABS(CR-1)
10060 IF CR=1 THEN 10090
10070 CALL HCHAR(XP,CP+2,95)
10080 GOTO 10100
10090 CALL HCHAR(XP,CP+2,C)
10100 IF S=0 THEN 10040
10110 CALL HCHAR(XP,CP+2,C)
10120 IF K=13 THEN 10340
10130 IF (K<>8)+(CP=1) THEN 10190
10140 CP=CP-1
10150 CALL GCHAR(XP,CP+2,CH)
10160 CALL KCH=CH
10170 GOSUB 10350
10180 GOTO 10030
10190 IF (K<>9)+(CP=28) THEN 10260
10200 CP=CP+1
10210 CALL GCHAR(XP,CP+2,CH)
10220 KCH=CH
10230 IF CP<LEN(K$)+2 THEN 10260
10240 K$=K$&CHR$(32)
10250 GOTO 10030
10260 IF CP-1=LEN(K$) THEN 10310
10270 GOSUB 10350
10280 IF CP=28 THEN 10030
10290 CP=CP+1
10300 GOTO 10030
10310 K$=K$&CHR$(K)
10320 CALL HCHAR(XP,CP+2,K)
10330 GOTO 10280
10340 RETURN
10350 SEG$(K$,1,CP-1)&CHR$(K)&SEG$(K$,
CP+1,LEN(K$)-CP)
10360 CALL HCHAR(XP,CP+2,K)
10370 RETURN
    
```

Line No.	Explanation of the Program
100-160	Demo routine.
10000-10020	Initialize the routine.
10030-10100	Flash cursor and wait for a key-press.
10110-10120	Check for the return.
10130-10180	Left cursor routine.
10190-10260	Right cursor routine.
10270-10300	Housekeeping on the line.
10310-10340	Add a character to the end of the line (append).
10350-10370	Insert a character into the line (type over).

—William K. Balthrop



LOGO CLONES: TI Graphics in a Turtle-Shell

by Sidney D. Nolte

*Learn the secret of the turtle clones!
Duplicate turtles draw on the screen
—with a single command.*

Have you ever wondered why the message OUT OF INK occurs in TI LOGO? It happens because only 256 characters can be used at any one time in graphics mode. The standard character set (tiles 32 to 96) removes 64 of these, leaving only 192 tiles for displaying all other graphics. When all 192 tiles are used, the OUT OF INK message appears. By employing the following procedures, you can use this apparent disadvantage to display some spectacular and unusual turtle graphics.

The CLEARSCREEN command clears all graphics and characters off the screen, except sprites. It actually fills the entire screen with the SPACE character (tile 32). When the turtle draws on the screen and there is a SPACE tile where it is instructed to draw, it replaces the SPACE with one of the 192 tiles available for graphics. At this time the new tile is cleared of any pattern that it might have contained, so that only what the turtle draws appears in that tile. When the turtle draws on the screen and encounters a tile other than a SPACE tile, it draws on that tile without erasing it first.

Now that we know about this unique scheme for drawing, we can place the same tile in several locations on the screen, and by drawing on that tile with the turtle, we cause the turtle's drawing to appear in each location that contains that tile. Try this short experiment on your system:

```
TELL TURTLE
CLEARSCREEN
PUTTILE 0 16 11
PUTTILE 0 10 10
FORWARD 10
```

Surprise! The FORWARD 10 command drew two lines because tile 0 was on the screen in two different

locations. If the screen had been filled with tile 0's, then you would have filled the screen with short lines.

The procedures listed below use this technique to create two different effects. The first procedure, STARS, draws a five-pointed star within a square area. The number of tiles on one side of the square is passed to it as the parameter M. With these procedures in memory enter:

STARS 4

There will be a little wait while the screen is prepared, but be patient—it's worth the wait. The result is many stars—each one within its own 16-tile area—being repeatedly drawn and erased throughout the screen.

The second procedure, called MANY, accepts two parameters: the length and the width of a rectangular area in which patterns will be drawn. For example, a 6 by 3 area would be used when you enter:

MANY 6 3

The key to success of these procedures is recursion. The DRAW procedure repeatedly calls itself, drawing and then erasing spirals on the screen. This effect is most spectacular when the spiral exceeds the size of the character blocks on the screen. The spiral actually spills over into an area which is a copy of what was already drawn, and the resulting patterns can be hypnotizing.

HCM

```
TO TEL TURTLE
HIDE TURTLE
HOME
SETHEADING 90
REPEAT 8 * M [FORWARD 8
  1 * RIGHT 90 FORWARD 8 * L
  EFT 90]
END

TO ROW J J
IF TILE = N STOP
TELL TILE CH
TEST 15 BOTH : I I + : I < 32
IF : J J + : J < : I : J J
MAKE ROW : J J + 1
END

TO COL I I
IF : I I = : M STOP
ROW : I I + 1
END

TO SPI L
IF FORWARD = L STOP
RIGHT 90 : L
SPI : L - 1
END

TO DRAW K * 16 * : N
SETHEADING 0
SPI K
DRAW
```

```
TO COLOR MANY BACK M N
TELL TURTLE UNDO 6
CLEARSCREEN
SPRAY COLOR
TELL TURTLE
HOME REVERSE
HIDE TURTLE
DRAW
END

TO SPRAY ROW J
IF : J > 23 STOP
MAKE : J CH 96
SPRAY ROW : J + : N
END

TO SPRAY COL I
IF : I > 31 STOP
SPRAY COL : I + : M
INIT
END

TO STARS M
TELL TURTLE M
CLEARSCREEN
MAKE ROW : M
SPRAY COLOR
TELL TURTLE
HIDE TURTLE
PEN REVERSE
DRAW
END

TO DRAW
HOME
SETHEADING 0
COLOR BACK GROUND RANDOM
REPEAT 40 [FORWARD 8 *
  : M RIGHT 144]
DRAW
```


ALL PROGRAMS IN THIS MAGAZINE



**ONLY \$3.95* DELIVERED
RIGHT TO YOUR DOOR!**

The same high-quality Apple, Commodore, IBM, and Texas Instruments programs with type-in-and-RUN listings in this issue are now available on ON DISK™ or ON TAPE™ to newsstand purchasers or subscribers of this magazine.

For only \$3.95* (barely covering the cost of a blank floppy disk or cassette tape), you receive all the programs for your particular brand of computer—truly a "Software Giveaway!"

To Order, Use The Bind-In Card Inside Rear Cover.

* Current Issue Price Only — See Center Bind-In Card For Back Issue Prices.