99'er HOME COMPUTER Magazine

Covering the Texas Instruments Brand of Home Computers

November, 1983
$3.50 in U.S.A.

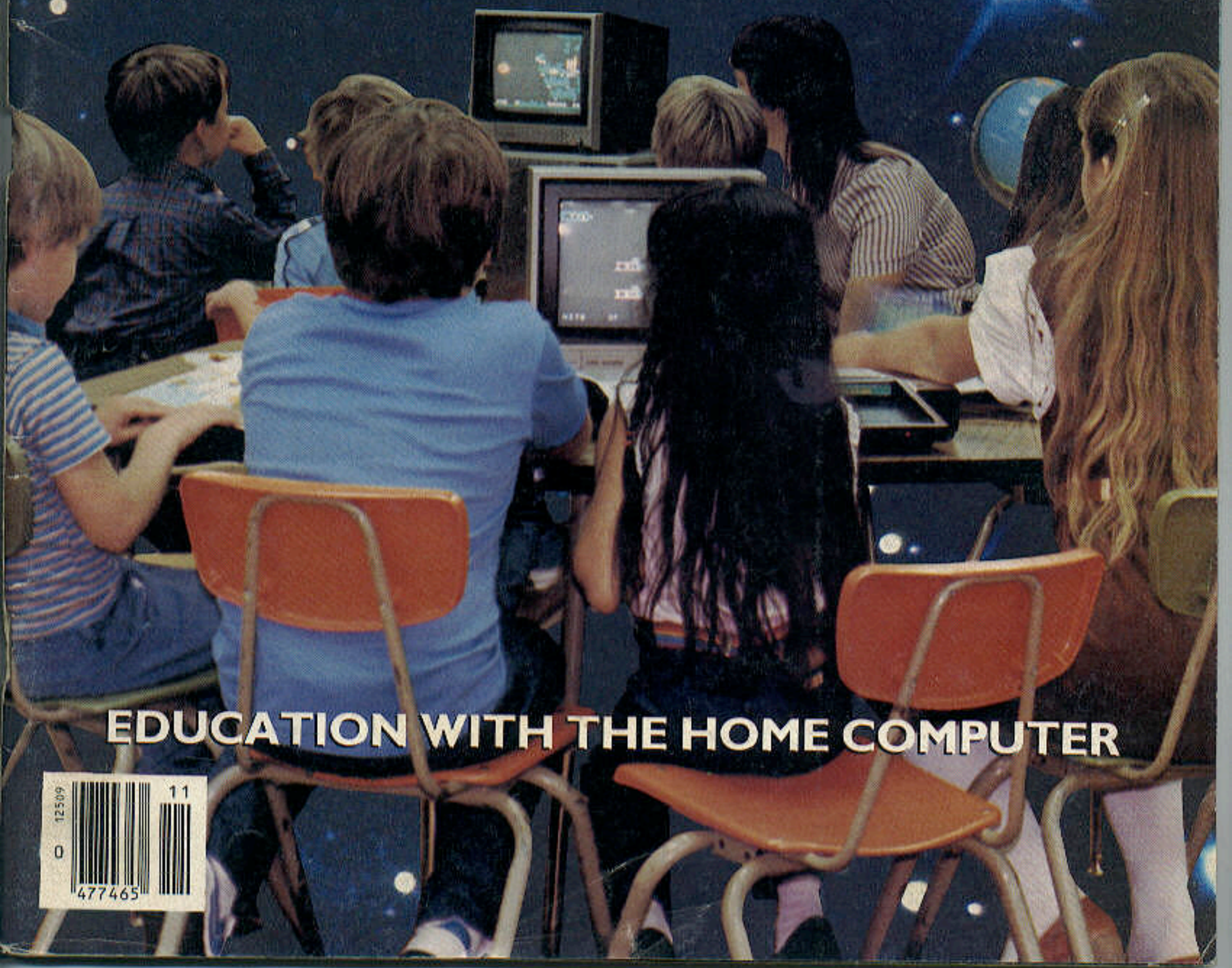
SQUEEZING THE MOST FROM BASIC

FIVE EASY LEARNING PROGRAMS

COMPUTER GAMING: RAMPAGING ROBOTS AND TANTALIZING TACOS

WORD PROCESSING FOR TI-WRITERS

ASSEMBLY LANGUAGE WITHOUT FEAR



EDUCATION WITH THE HOME COMPUTER

0 12509 11
477465

THE SECRETS OF PERFECT MEMORY: ONE AND ONE HALF EARTH DOLLARS

AT LAST: THE WHOLE TRUTH ABOUT FLOPPIES.

Amazing book reveals all!

How to keep from brainwashing your disk so it never loses it's memory.

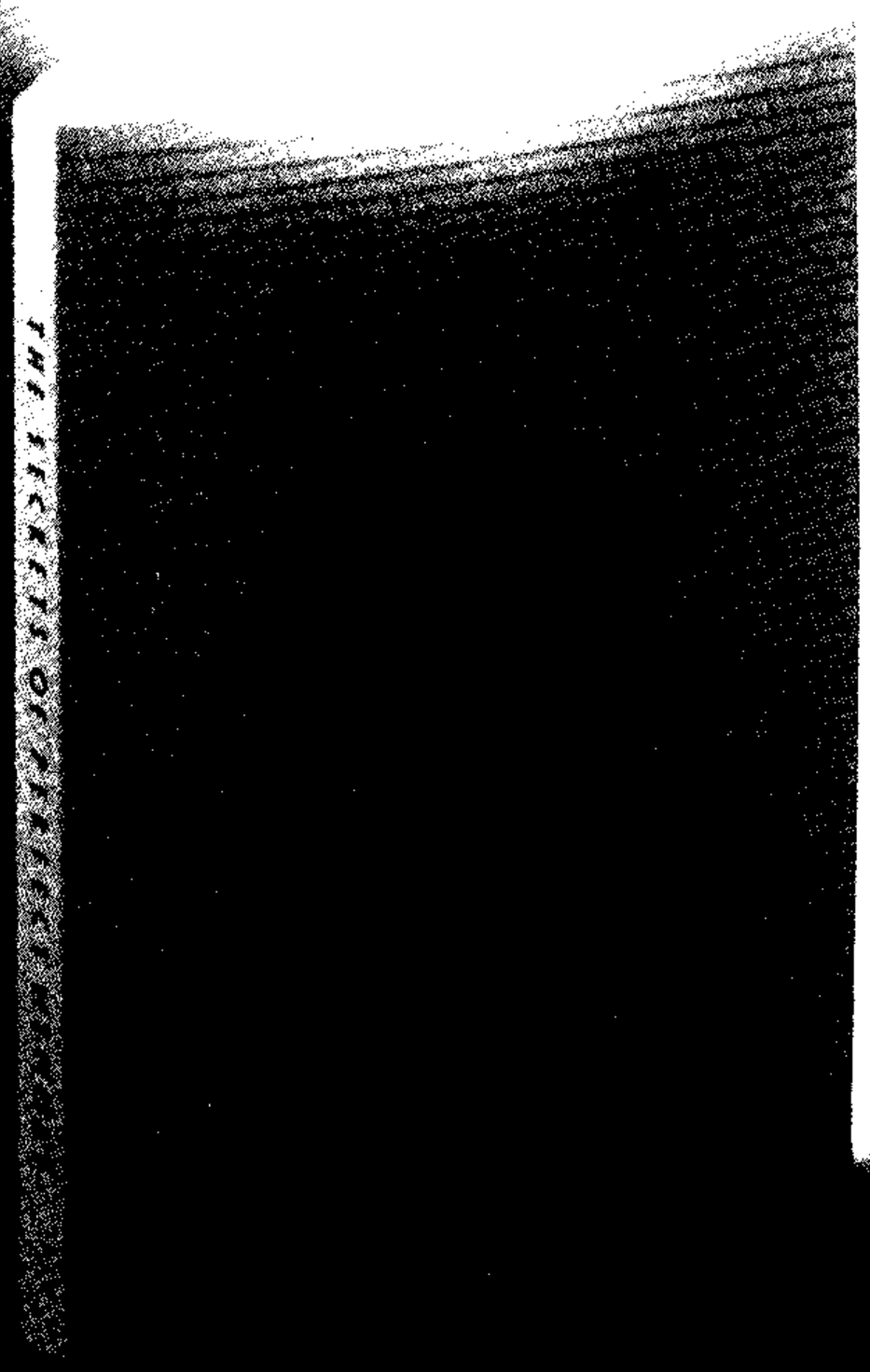
How fingerprints can actually damage disks. Unretouched Kirlian photographs of UFO's (Unidentified Floppy Objects)! The incredible importance of making copies: the Department of Redundancy Department—and what goes on when it goes on! Powerful secret methods that scientists claim can actually prevent computer amnesia! All this, and much more . . .

In short, it's an 80-page plain-English, graphically stunning, pocket-sized definitive guide to the care and feeding of flexible disks.

For The Book, ask your nearest computer store that sells Elephant™ disks, and bring along one and one half earth dollars.

For the name of the store, ask us.

ELEPHANT MEMORY SYSTEMS® Marketed exclusively by Leading Edge Products, Inc., Information Systems and Supplies Division, 55 Providence Highway, Norwood, MA 02062. Call toll free 1-800-343-8413. In Massachusetts, call collect (617) 769-8150, Telex 951-624.

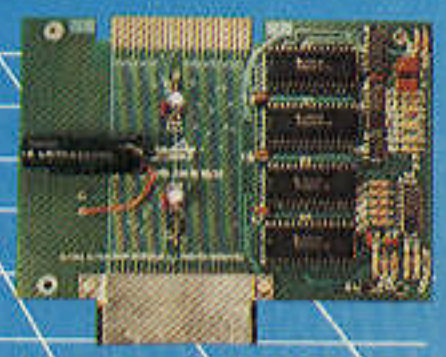


See us at BOOTH #3327

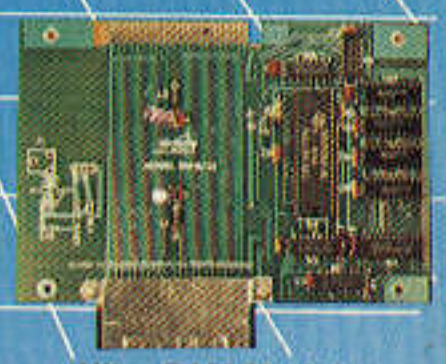
COMDEX™/Fall '83

November 28-December 2, 1983

AFFORDABLE EXPANSION



Ultra-Smart RAM Module



Ultra-RAM Module

with cost-effective, versatile Ultra-Modules®

UPGRADE your TI99/4A the easy way to use TI LOGO*, TI EXTENDED BASIC*, TI MINI-MEMORY*, ASSEMBLERS and numerous other software packages requiring expanded data storage. Adding either of these space saving modules gives you more system capability without costly expansion extras. The daisy chain type connector achieves a simple direct plug-in installation/expansion for a compact and more portable system.

Both the ULTRA-RAM and ULTRA-SMART RAM modules emulate the 32K memory module used in the TI PERIPHERAL EXPANSION SYSTEM* (smaller 8K and 16K sizes are available upon request). All 32K BYTES of storage are software accessible. The units come as a stand alone module (DM/CM series).

The ULTRA-SMART RAM uses CMOS technology memories which provide a reliable, non-volatile back-up for power failure, data and program storage. The included battery system allows storage of information for typically over 3 years. A low battery indication is provided to accommodate easy battery replacement without loss of memory.

All Modules Feature

- ★ Shielded connector design for RFI/EMI integrity
- ★ Daisy chain compatibility with the computer's built-in connector without extra cables
- ★ Durable and attractive enclosures
- ★ State of the art design and reliability
- ★ Easy to use installation/operation manual
- ★ 90 day limited warranty

New Products available soon

- User Programmable Modules
- Ultra-Bus® Expansion System and
- MINI-MOD Expansion Modules
- RS-232 Systems

ORDER TOLL FREE!

1-800-526-6050 ask for ext. #**1216**
Arizona 1-800-352-0458.
Delivery 2-4 weeks

Ultra-RAM Module
☐ DM-32 $159.95

Ultra-Smart RAM Module
☐ CM-32 $269.95

☐ I need additional information; please send technical brochures.
☐ Enclosed is $5.00 for each manual on the products checked above (applicable toward purchase price).
☐ Send info on Users Group quantity discounts.
Illinois deliveries add 6% sales tax.
Please add $5.00 for postage for each item ordered.
☐ Check or M.O. enclosed.
Charge to: VISA ☐ MASTER CARD ☐

Account Number _____ Expiration date _____

Signature _____

Name _____

Address _____

City _____ State _____ Zip _____

Ultracomp Systems

Ultracomp Systems
1001 Ogden Avenue #5
Downers Grove, IL 60515-9228

On Screen

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

" The machine's history is a perfect example of full-circle product positioning in a rapidly changing market."

Why is it that events more often than not swing around full-circle? This pattern is obvious, of course, in the fashion industry, where lapel widths and hemlines wax and wane on a cyclical basis . . . where the suit you bought ten years ago suddenly becomes stylish again.

We really shouldn't be too surprised, then, when product orientation in our home computer industry describes a full circle—but we are. Somehow, we just haven't regarded this infant business with the same set of expectations we've learned to apply to, say, the automobile, clothing, or music industries. Nevertheless, even in this relatively young enterprise, the rule of cycles is starting to emerge.

The current marketing focus on educational use of the Home Computer is a case in point: When the original 99/4 Home Computer appeared on the scene, the product's strongest appeal was to a market composed of progressive, more affluent consumers who valued their children's educations enough to plunk down $1300 (for the console, color monitor, and speech synthesizer) so their progeny could benefit from such innovative software cartridges as *Early Reading*, *Early Learning Fun*, and *Beginning Grammar*. Since then, we've seen the TI product line positioned every-which-way—as a home-management appliance, professional's tool, game console, home-business machine, and inexpensive computer literacy aid. Finally—nearly four years after the launch date—the 99/4A appears in its present affordable $100 incarnation as a versatile learning machine for early education. The machine's history is a perfect example of full-circle product positioning in a rapidly changing market.

With TI's new fourth-quarter educational thrust, it now appears that the 99/4A has finally found its ideal niche—one in which its inherent value and unique speech and graphics features are in perfect harmony with its targeted market. For educational use in the home or schoolroom it is superb. Among home computers the Texas Instruments machine enjoys the largest number, by far, of educational cartridge-based programs. In addition, it is a very easy machine on which to program full-featured education applications in the BASIC language. And finally, for more serious educational use, it provides the only system configuration in town that allows users to program in Pascal, Pilot, and LOGO and run the extensive PLATO courseware library for less than $800. We here at *99'er Home Computer Magazine* applaud TI's decision to abandon the price orientation of its most recent marketing effort for the current soft-sell emphasis on education, ease, and software.

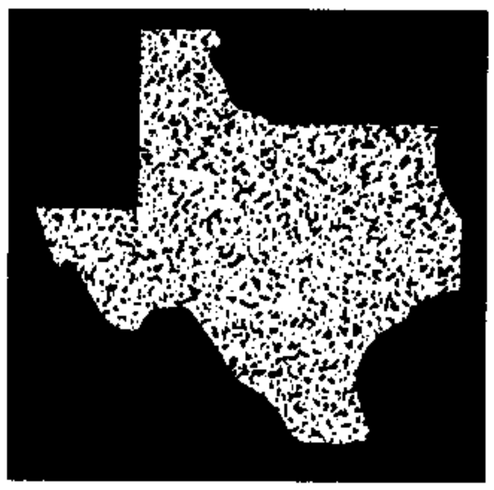
Here at editorial headquarters, we plan each issue's theme several months in advance. So it was gratifying to discover that our November education issue would appear at just the right time—to coincide with the new educational focus of the Home Computer. We hope our readers will show their copies of this November issue to "the powers that be" within their local school communities so that more of our teachers will become aware of just what a great little Educational Home Computer we have. As a rule, each 99/4A we see placed in our schools will ultimately be responsible for selling four to five additional machines to home users. We all stand to benefit from this much larger user base. And while you're at it, let the educators and parents know how you value (highly, we hope) your subscription to *99'er HCM*.

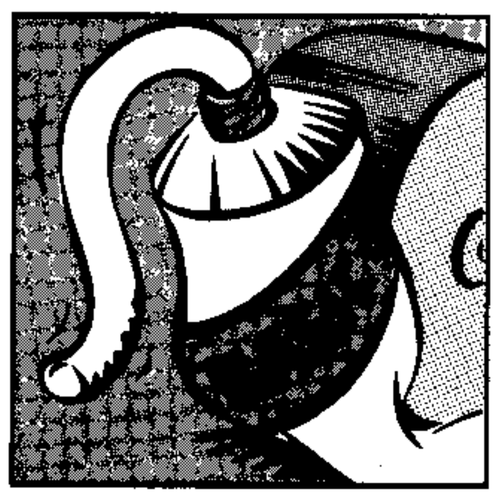
HOME COMPUTER MAGAZINE

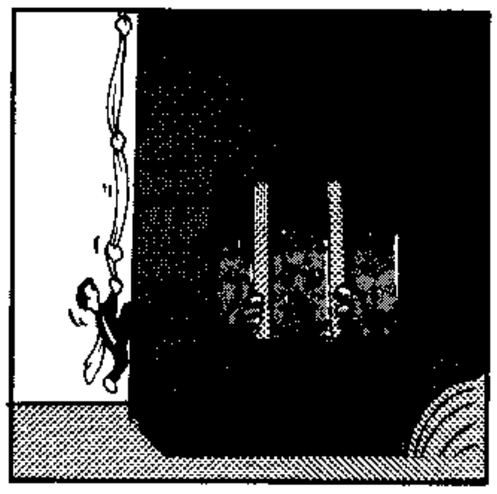
This month's cover photograph captures the moment just before the dawn of a new era. The young people clustered around the computers have already begun practicing the educational techniques of the future. What do they see as they look into the windows of computer technology? Perhaps they see beyond the screens into even more compelling scenes—their own private visions of what the future may bring. These children of the Computer Age are growing up using a technology that is growing up right along with them. It may not seem possible for a little Home Computer to do all that, but as you will see in these pages, this versatile machine is a powerful educational tool—equally at home in the classroom and the family room.

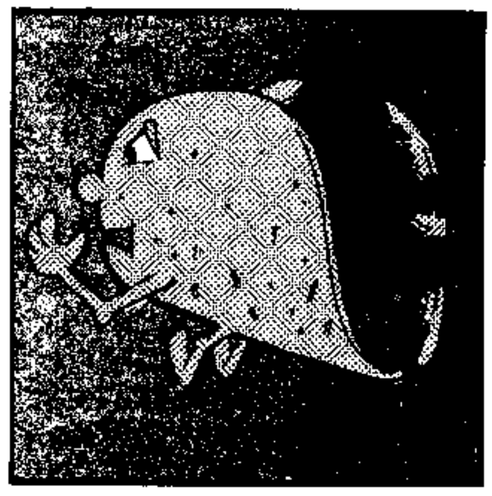
CONTENTS

November, 1983 Vol. 2, No. 13









99'er HOME COMPUTER magazine™

99'er Home Computer Magazine (ISSN 0745-6913) 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 $7 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 $4.00 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 address changes to **99'er 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 **99'er 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 **99'er Home Computer Magazine**. All mail directed 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 **99'er Home Computer Magazine's** unrestricted right to edit and comment. **99'er Home Computer Magazine** assumes no liability for errors in articles 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 **99'er Home Computer Magazine** or the publisher unless explicitly stated.

INSIDE 99'er

Back in the first century Plutarch observed that, "The very spring and root of honesty and virtue lie in good education." We still believe that, but nineteen centuries later we pursue the ideal of "good education" with tools quite different from Plutarch's. Anyone looking at quality educational process now would conclude that the Home Computer is at the hub of good education.

Its contributions take many forms. In *Creative Learning Activities for Children* are five programs that provide practice in letter and number concepts, using money, telling time and recognizing musical tones. Entertaining programs like these will engage even the most reluctant students.

Our LOGO turtle offers the aspiring student an essential tool of education in *LOGO Lexicon*. This spelling program will help students of all ages to really power up their vocabularies!

Now that we're tuning into learning, let's check up on what's happening in the Educational Software Division of Texas Instruments in an *Interview with Dale Osborn*. Not only does this interview confirm TI's commitment to education, but it hints at new directions to be taken in the near future.

That longing to travel to new places is alive and well in our Extended BASIC program *Let's Build America*. In the spirit of pioneer travelers, you can go from shore to shore, state by state, mapping out the entire United States. Homebodies and jet-setters alike will agree that this is a good way to visit more of our country on less of their money.

Getting more for less is also the focus of *Squeezing the Most Out of BASIC*. This tutorial takes you through a flight simulation program, sharing secrets to help your own BASIC programs straighten up and fly right.

While we're righting things, let's write prose like the pros do. *TI-WRITER Tutorial* explains how to move mountains of text and use Word Wrap mode to create multi-purpose charts. Our tutorial's traveling companion is *At Home in the Office*. A manufacturers representative saves time and effort in his small business by putting all his order forms, letters and accounts on *TI-WRITER* files. He's definitely headed for smooth sale-ing.

Oceans of possibilities await after our *Multiplan Medium* explains how the eXternal Copy command can be used in inventory control. You'll think of countless applications for this inventory program. It can save you from running short of crucial supplies in your home, business, or classroom.

Direct from a hypothetical classroom come students Rick, Buddy and Sarah to discuss *Computer-Arrested Instruction*. Our interview with these representatives of the tongue-in-cheek generation will clear up some common misconceptions about computers in the classroom.

Back in the real world of computer-assisted instruction we join *PLATO's Progress*. This visit to the Academy will show us how to use the *Reading Skills Survey* package to identify areas in which students may need help in order to improve their reading skills.

Embellishing your Assembly Language skills is the subject of *Have No Fear: Assembly Language Won't Byte, Part Two*. By article's end, you will have learned your first words in Assembly Language. Would you believe you'll really be able to communicate using mnemonics like BLWP @ VMBW?

If that sounds like far-fetched lingo, how about being transformed into an empty taco shell yearning for some hot sauce, cheese and chili peppers? All this fun can be yours in *Taco Man*. While madly gobbling up taco-innards, you are chased by hungry tacovores. If you begin to feel that you are out of control, perhaps *Robo Chase* will be more your style. In this game you force the mechanical menaces to crash into electrified barriers and self-destruct. You can be successful too, if you have learned how to avoid Spunky the Martian!

It all comes back to education again, and the age-old story of the hungry man on the seashore. You are faced with the dilemma of whether to give him a fish so he won't go hungry today, or to teach him to fish so he will never go hungry again. Clearly what we have tried to do through these articles is to provide you with some ways to use your trusty Home Computer to fish for more knowledge. Then you can put it to work preparing yourself (and the next generations) for the technological banquets of the future.

Until next month, keep on reading, learning and RUNing!

99'er

We at 99'er Home Computer Magazine extend our gratitude to the following for their contributions to this months cover: to our own W.K. Balthrop for taking the photograph, to the staff of Mt. Vernon Elementary School in Springfield, Oregon, for the use of their Media Center, and to William Suggs, the Planetarium Director at the Lane Education Service District in Eugene, Oregon, for providing us with the starscape used to create the cover background.

LETTERS TO THE EDITOR

One Program Runs the Other

I'd like to know how to write a program that will read the diskette catalog and then allow execution of any of the resident programs by simply entering a number or letter identifying the program without having to know *a priori* the programs that are on the diskette. Whenever I assign a program name to A$, for example, and then try to execute RUN A$—where A$ is equal to DSK1.PROGRAM2, for example—I get an error message.

And can you tell me how to suppress the automatic first page advance in the *TI-WRITER* word processor?

Victor Friedmann
Dhahran, Saudi Arabia

The first problem you mention has a long and hoary history in the letters to 99'er HCM. *Longtime readers will remember the "General Purpose Load Challenge" by Charles Ehninger in Volume 1, Number 2. Solutions came from John Clulow (Volume 1, Number 3), Tim MacEachern and "A. Kludge" (both in Volume 1, Number 4) and Rick Rothstein (Volume 1, Number 5). We reprint here Rothstein's* Poor Man's Program Loader, *so-called because it doesn't require expansion memory, which the others do.*

"The following listing should be SAVEd and then RUN on each and every diskette containing programs capable of being run in Extended BASIC. I have given this program the file name LOADER. (If you wish to give it another name, you will have to make the appropriate change in Line 270.) LOADER should also be RUN whenever programs (not data files) are added to, or deleted from, a diskette. Doing so will cause LOADER to read the diskette's catalog and create, in a MERGE format file, a separate program that displays the names of all programs *on the diskette, except LOADER, and will automatically RUN one at the touch of a button.*

"I have given this second program the file name CAT—short for catalog. (If you wish to give it another name, you will have to make the appropriate change in Line 290.) To enter CAT into your computer, simply type MERGE DSK1.CAT. (Extended BASIC does not respond to the singly-typed command MERGE DSK1.CAT : : RUN) The screen will clear, the name of the diskette will be printed, and a menu will be displayed showing the program contents of the diskette. In front of each program name will be a letter of the alphabet. Pressing one of them will automatically RUN the program listed next to it.

"When MERGEing the CAT program, it is not *necessary to erase an existing program already in memory unless it is extrememly large or uses the array name(s) K@ and/or S@. The line numbering of CAT starts with 1 and increases in increments of one. Hence, upon MERGEing, CAT will either load in front of, or over-write, as necessary, any program lines already in memory.*

"Some final points:

"1. Always *use Disk Drive One (DSK1) when executing either LOADER or CAT. This is because LOADER will always store the CAT file to Disk Drive One and CAT will always "open" Disk Drive One in order to RUN the selected program.*

"2. With no memory expansion unit attached, CAT will not be able to RUN any program which is so large that it requires a CALL FILES(1) command to be issued unless *CALL FILES(1) was executed prior to MERGEing CAT.*

"3. Once created, CAT is totally independent of LOADER and may be MERGEd and RUN at any time. The purpose of LOADER is to update CAT whenever changes are made in the program content of the diskette."

One additional comment: When the contents of a disk will not be changed frequently, the following sequence of commands may be entered to obtain a LOAD file:

RUN "DSK1.LOADER"
NEW
MERGE DSK1.CAT
SAVE DSK1.LOAD

The menu will then automatically appear upon entering Extended BASIC.

```
100 REM ******************
110 REM *                *
120 REM *   POOR MAN'S   *
130 REM * PROGRAM LOADER *
140 REM *       BY       *
150 REM * RICK ROTHSTEIN *
160 REM ******************
170 REM
180 CALL CLEAR :: PRINT "PROGRAM STATUS.......WORKING" :: CLS$="CLEAR" :: DIM A$(20) :: OPEN #1:"DSK1.",INPUT ,RELATIVE,INTERNAL
190 DEF LN$(N)=CHR$(0)&CHR$(N)
200 DEF DI$(R)=CHR$(162)&CHR$(240)&CHR$(183)&CHR$(200)&CHR$(LEN(STR$(R)))&STR$(R)&CHR$(179)&CHR$(200)&CHR$(1)&STR$(COL)&CHR$(182)&CHR$(181)
210 DEF IF$(N)=CHR$(132)&"K@"&CHR$(190)&CHR$(200)&CHR$(2)&STR$(N)&CHR$(176)&CHR$(169)&CHR$(199)&CHR$(LEN(A$(I-64))+5)&"DSK1."&A$(I-64)
220 FOR I=0 TO 20
230 J=J+1 :: INPUT #1:A$(I),B,C,D :: IF I=0 THEN 240 ELSE IF I>=127 OR LEN(A$(I))=0 THEN 250 ELSE IF ABS(B)<>5 OR A$(I)="LOADER" THEN 230
240 NEXT I
250 CLOSE #1 :: EN$=CHR$(181)&CHR$(199)&CHR$(28)&"PRESS <ERASE> TO END PROGRAM"&CHR$(0) :: COL=1 :: L=I-1 :: OPEN #2:"DSK1.CAT",VARIABLE 163
260 PRINT #2:LN$(1)&CHR$(157)&CHR$(200)&CHR$(5)&CLS$&CHR$(0)
270 PRINT #2:LN$(2)&DI$(1)&CHR$(199)&CHR$(28)&"CATALOG"&RPT$(" ",12-LEN(A$(0)))&"DISK NAME--"&A$(0)&CHR$(0)
280 COL=8 :: FOR I=1 TO L :: PRINT #2:LN$(I+2)&DI$(12+I-INT(L/2))&CHR$(199)&CHR$(3+LEN(A$(I)))&CHR$(I+64)&"--"&A$(I)&CHR$(0) :: NEXT I
290 PRINT #2:LN$(L+3)&CHR$(162)&CHR$(240)&CHR$(183)&CHR$(200)&CHR$(2)&"24"&CHR$(179)&CHR$(200)&CHR$(1)&"1"&CHR$(182)&CHR$(238)&EN$
300 PRINT #2:LN$(L+4)&CHR$(157)&CHR$(200)&CHR$(3)&"KEY"&CHR$(183)&CHR$(200)&CHR$(1)&"0"&CHR$(179)&"K@"&CHR$(179)&"S@"&CHR$(182)&CHR$(0)
```

Continued on p. 48

Entering 99'er Programs

New readers should be aware that within the magazine's pages are found actual computer programs that *you* can put into your Home Computer and enjoy.

Make sure you have any special system components required by the program (e.g., the Speech Synthesizer, Extended BASIC cartridge, etc.). Then, using the console keyboard, you can type the printed magazine listing (character for character, and line by line) into the computer's memory.

Before entering the program, connect a cassette recorder to the computer. Make sure you have two blank cassette tapes. For each 10-20 lines you type in, use SAVE CS1 to save that program segment onto one of the tapes. Alternate between the two tapes each time you save the program. Be sure to rewind to the beginning of each tape before saving, so that you always record over and replace the shorter segment of program lines with the longer segment. By following this procedure, you'll always retain most of your work even if the lights go out or someone turns off the computer.

Double check your typing against the program listing for errors, and *then* have someone else check it. The most common errors are typing the letter "O" instead of the number "0" (zero)—they are *not* interchangeable to the computer. This is also true for the letters "I" and "L" and number "1" (one). See "Key-In Reference"

Every time you make a correction to your program, SAVE CS1 and switch the tapes. Once all the errors are corrected, you will have a good copy of the program on the last tape. Before turning off the computer, put the other cassette tape in your recorder and once again SAVE CS1. Now, if one tape gets damaged, you won't have to enter the program listing via the keyboard all over again. Have fun and happy computing.

Programming Conventions

KEY-IN REFERENCE

```
100 REM 1234567890!@#$%^&*()=+ABCDEFGHIJKLMNOPQRSTUVWXYZ`[]_?'",.<>/-;:
```

=End of Program or Article

compu·prestidigitation

(kóm·pū·prĕs·teh·dĭ·jeh·tā·shŭn) —*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'ER VERSION

2 . 13 . 1 . XB AL MM EM

- volume no.
- issue no.
- version
  - 1 = original program
  - 2 . . . n = no. of update
- TI Extended BASIC
- Assembly Language
- Mini-Memory Required
- 32K Expansion Memory Required

CREATIVE LEARNING ACTIVITIES

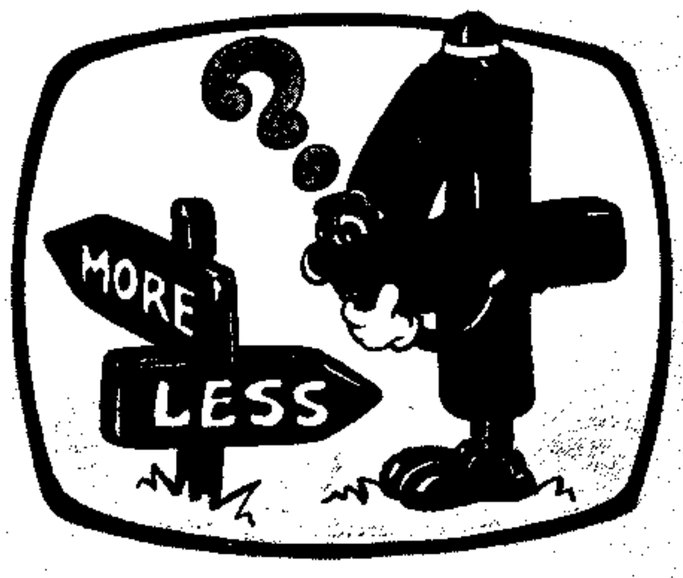
FOR CHILDREN

by Daniel R. Rozler
147 Lowell Lane
West Seneca, NY

As an educational tool, the computer's most important characteristics are its power to "simulate" and its ability to present drills and quizzes. At one moment the computer can be a nuclear reactor approaching "melt-down," a polluted body of water, or two companies competing in the open market. The next moment that same computer can be quizzing Johnny—asking him to add, subtract, multiply, divide, or spell words correctly! A simulation depends on a computer language's ability to generate random numbers. Many simulation programs are commercially available; some good, some not so good. I am interested in developing programs that invite a child to deal with the concepts of order and comparison within a set of objects (numbers, letters, musical notes, amounts of money, time, etc.). Utilizing the many fine features of Extended BASIC, I constructed five such programs for my children.

I used the DISPLAY AT, ACCEPT AT, CALL SAY, and SPRITES features to make the programs attractive (and powerful too)! All programs utilize the CALL SAY subprogram.

They say good things come in small packages, and indeed, a program need not be long to hold a child's attention or stimulate his or her curiosity. Hopefully, *99'er HCM* readers will find these programs educationally worthwhile.

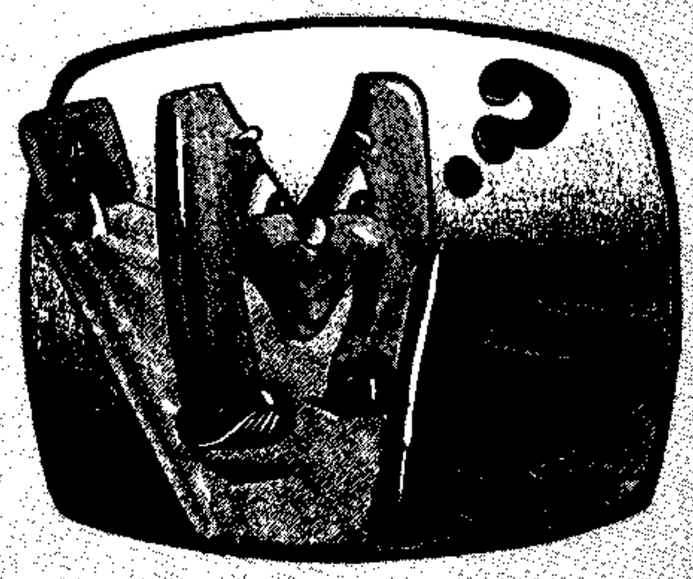


In *Guess a Number* the user has 5 chances to determine a random counting number. Helpful hints are given as the user "homes in" on the unknown number. The range of numbers varies from a starting value between 1 and 5 to an upper limit between 30 and 40.

Guess a Number

Guess a Number
Explanation of the Program

Line Nos.	
100-140	Program header.
150-220	Display the instructions and generate the random numbers.
230-400	Test the guess and display the hints.
410-430	Display a message after 5 incorrect responses.
440-500	Display a message if the guess is correct.
510-540	Display directions allowing the user to continue playing or terminate the program.
550-590	Game end routine.
600-650	Subroutines for delaying and erasing lines.

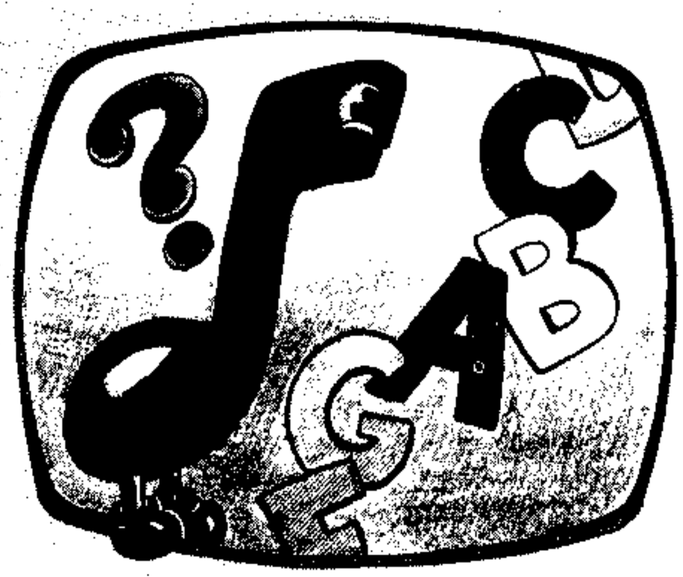


Guess a Letter is very similar to *Guess a Number*. The user has four tries to determine a random letter. Again, hints are displayed in order to aid the user.

Guess a Letter

Guess A Letter
Explanation of the Program

Line Nos.	
100-140	Program header.
150-210	Create special characters: a rectangle for display.
220-320	Display instructions and generate a random number from 65 to 90, ASCII for letters.
330-500	Test responses and give hints.
510-610	Display the correct letter in the form of a flashing sprite inside a rectangle.
620-690	Display of message when the user's response is correct. A sprite carries the letter across the top of the screen.
700-730	Display message for continued playing or termination of the program.
740-790	Game end routine.
800-810	Delay subroutine.
820-840	Erase subroutine.



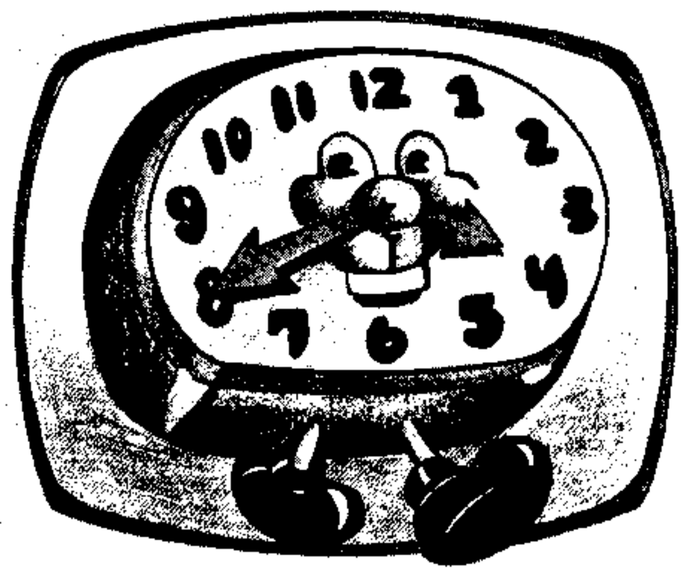
In *Guess a Musical Note* a random musical note(C Scale C, D, E, F, G, A, B, High C) is played. The user has two chances to identify the tone sounded.

Guess a Musical Note

Explanation of the Program

Line Nos.

100-140	Program header.
150-180	Define special characters for a note symbol and a High C.
190-210	Display title screen.
220-510	Read notes, frequencies and colors into arrays. Display and play the notes twice.
520-650	Display the notes and their symbols. Directions are shown describing how to enter a choice.
660-790	Generate and play a random tone from C to High C for 2 seconds.
800-890	Play the correct tone and display a flashing sprite representing the correct choice.
900-910	Display message after correct choice.
920-990	Play again routine.
1000-1050	Game end routine.
1060-1110	Subroutine to display sprite.
1120-1130	Delay subroutine.
1140	END



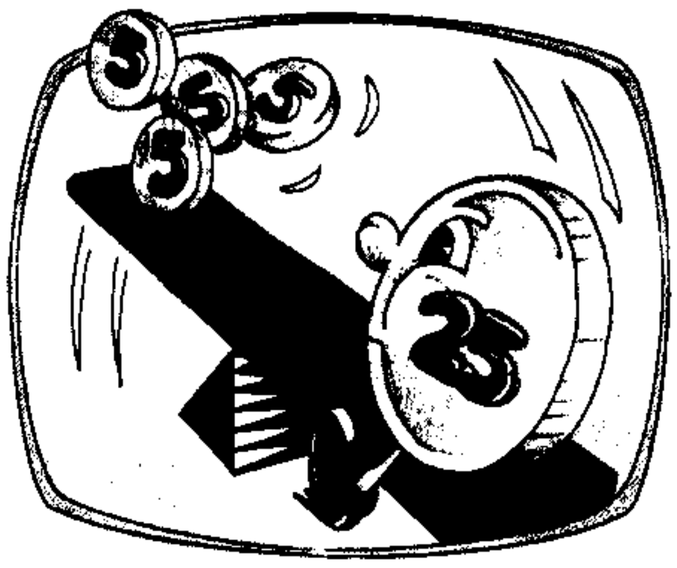
Tell the Time causes the computer to simulate the face of a clock. A random time is displayed (minutes are always in multiples of five: 5,10,15. . . 55). Time must be entered as hours followed by minutes. The user has two chances to answer correctly.

Tell the Time

Explanation of the Program

Line Nos.

100-140	Program header.
150-310	Definition of special characters for display lines and clock hands.
320-460	Store clock hand character codes and their row and column positions in arrays.
470-610	Store codes for clock numerals and their row and column positions in arrays.
620	Call the subroutine that displays the clock face.
630-720	Display directions.
730-950	Generate and display a random hour and minute hand. Enter and test user's choices.
960-990	Display of message when choice is wrong
1000-1030	Display of correct answer when 2 incorrect choices are entered.
1060-1120	Display of message when correct choice has been entered.
1130-1160	Play again routine.
1170-1290	Erase routine.
1300-1350	Subroutine for displaying the clock face.
1360-1410	Game end routine.



Working with Money requires the user to know the number of coins (pennies, dimes, quarters, half dollars) contained in a random amount (maximum of 4) of other coins (dollar). Again the user has two chances to answer correctly.

Working with Money

Explanation of the Program

Line Nos.

100-140	Program header.
150-160	Define the cent character.
170-250	Read amounts of money into arrays.
260-390	Display a table showing amounts of money.
400-460	Generate random numbers. Determine the amount of money and labels to be displayed.
470-530	Display the question. Accept the user's response and test the response.
530-600	Display message after wrong response.
610-680	Display message when the user correctly answers the question.
690-710	Continue or terminate the program.
720-750	Erase routine.
760-800	Game end routine.

Loader

Explanation of the Program

Line Nos.

100-180	Program header.
190-360	Display menu.
370-380	Accept and branch to option selection.
390-430	RUN selected option.
440	END.

Programming Notes

As they now stand, the six accompanying programs are designed to run on a disk system and to reside on one diskette in disk drive one. On selecting Extended BASIC it will automatically load Listing 6 (*if* it is named LOAD), which can then RUN any one of the other five programs, depending on the user's menu selection. (The names of the programs in lines 390-430 *must* correspond to the filenames on diskette.) When any of the five programs finishes, it RUNs the menu program again. The user can then either select another program or leave the system. This device, available in Extended BASIC, permits interrelated programs or program options to be written as short, independent programs. This avoids the potential problem in long programs of running out of memory.

If you don't have a disk system, however, you can still use these programs as entirely separate programs by taking the following steps. **Omit** *Loader*. Instead of the statement given in the following lines, type in **REM**:

Number:	line 580
Letter:	line 780
Note:	line 1040
Time:	line 1400
Money:	line 790

The programs as written also use Extended BASIC'S CALL SAY subprogram to generate speech. If you don't have a Speech Synthesizer, instead of the given statements in the following lines, type in **REM**:

Number: lines 220, 250, 310, 360, 410, 500, 520, 570;
Letter: lines 310, 350, 410, 460, 510, 690, 710, 770;
Note: lines 210, 360, 520, 590, 680, 770, 800, 810, 910, 930, 1020;
Time: lines 650, 670, 700, 100, 1060, 1130, 1390;
Money: lines 550, 620, 780.

Continued on p. 10

Creative . . . from p. 9

Guess a Number

```
100 REM ********************
110 REM *  GUESS A NUMBER  *
120 REM ********************
130 REM  BY DAN ROZLER
140 REM  99'ER VERSION 2.13.1XB
150 CALL CLEAR
160 RANDOMIZE
170 S=INT(RND*5+1)::F=INT(RND*
    11+30)::N=INT(RND*(F-2*S+1
    )+S)
180 DISPLAY AT(9,6):"GUESS A NU
    MBER FROM"
190 DISPLAY AT(11,5):S;"TO";F;"
    (IN 5 TRIES)"
200 CALL COLOR(12,7,7)
210 CALL HCHAR(7,3,120,28)::CA
    LL HCHAR(13,3,120,28)
220 CALL SAY("GUESS A NUMBER")
230 TRY=1
240 DISPLAY AT(15,10):"YOUR GUE
    SS?"
250 CALL SAY("YOUR GUESS")
260 ACCEPT AT(15,21)BEEP VALIDA
    TE(DIGIT)SIZE(2):L
270 IF L=N THEN 440 ELSE 280
280 IF TRY=5 THEN 410 ELSE 290
290 IF L<N THEN 350 ELSE 300
300 DISPLAY AT(18,10):"THE NUMB
    ER IS":"SMALLER":"THAN";L
310 CALL SAY("THE NUMBER IS SM
    ALLER THAN")
320 GOSUB 600
330 GOSUB 620
340 GOTO 390
350 DISPLAY AT(18,10):"THE NUMB
    ER IS":"BIGGER THAN";L
360 CALL SAY("THE NUMBER IS LA
    RGER THAN")
370 GOSUB 600
380 GOSUB 620
390 TRY=TRY+1
400 GOTO 250
410 CALL SAY("TIME IS OUT")
420 DISPLAY AT(17,3):"THE CORRE
    CT ANSWER IS";N
430 GOTO 510
440 DISPLAY AT(17,9)BEEP:"* VER
    Y GOOD! *"
450 FOR COUNT=1 TO 6
460 DISPLAY AT(18,12)BEEP:"CORR
    ECT!"
470 FOR PAUSE=1 TO 50 :: NEXT P
    AUSE
480 DISPLAY AT(18,12):"
    " :: FOR PAUSE=1 TO 50 :: N
    EXT PAUSE
490 NEXT COUNT
500 CALL SAY("THAT IS RIGHT VER
    Y GOOD")
510 DISPLAY AT(22,5):"DO AGAIN-
    YES OR NO?"
520 CALL SAY("DO AGAIN YES OR N
    O")
530 ACCEPT AT(22,24)BEEP VALIDA
    TE("YESNO")SIZE(3):ANS$
540 IF SEG$(ANS$,1,1)="Y" THEN
    150
550 CALL CLEAR
560 DISPLAY AT(12,12):"GOODBYE"
570 CALL SAY("GOODBYE")
580 RUN "DSK1.LOAD"
590 STOP
600 FOR DELAY=1 TO 200 :: NEXT
    DELAY
610 RETURN
620 CALL HCHAR(18,1,32,32)
630 CALL HCHAR(19,1,32,32)
640 RETURN
650 END
```

Guess a Letter

```
100 REM ********************
110 REM *  GUESS A LETTER  *
120 REM ********************
130 REM  BY DAN ROZLER
140 REM  99'ER VERSION 2.13.1XB
150 CALL CHAR(136,"0000000FFFF00
    0000")::CALL COLOR(14,5,8)
160 CALL CHAR(129,"0101010101010
    101")
170 CALL CHAR(130,"8080808080808
    080")
180 CALL CHAR(131,"FF0000000000F
    0000")
190 CALL CHAR(132,"0000000000000
    00FF")
200 CALL COLOR(13,2,8)
210 CALL CLEAR
220 CALL DELSPRITE(ALL)
230 RANDOMIZE
240 DISPLAY AT(9,6):"GUESS A LE
    TTER FROM"
250 DISPLAY AT(11,7):"A TO Z (I
    N 4 TRIES)"
260 CALL COLOR(12,7,7)
270 CALL HCHAR(7,3,120,28)::CA
    LL HCHAR(13,3,120,28)
280 FOR I=1 TO 13 :: CALL HCHAR
    (6,10+I,64+I)::NEXT I
290 FOR I=1 TO 13 :: CALL HCHAR
    (14,10+I,97+I)::NEXT I
300 CALL HCHAR(5,3,136,28)::CA
    LL HCHAR(15,3,136,28)
310 CALL SAY("GUESS A LETTER FR
    OM A TO Z")
320 N=INT(RND*26+65)
330 TRY=1
340 DISPLAY AT(16,10):"YOUR GUE
    SS?"
350 CALL SAY("YOUR GUESS")
360 ACCEPT AT(16,21)BEEP VALIDA
    TE(ALPHA)SIZE(1):L$
370 IF L$=CHR$(N)THEN 620 ELSE
    380
380 IF TRY=4 THEN 510 ELSE 390
390 IF L$<CHR$(N)THEN 450 ELSE
    400
400 DISPLAY AT(18,10):"THE LETT
    ER":"DISPLAY AT(19,10):"COM
    ES BEFORE";L$
410 CALL SPGET(L$,L$$)::CALL S
    AY("THE LETTER IS BEFORE",L
    $,"TRY AGAIN")
420 GOSUB 800
430 GOSUB 820
440 GOTO 490
450 DISPLAY AT(18,10):"THE LETT
    ER":"DISPLAY AT(19,10):"COM
    ES AFTER";L$
460 CALL SPGET(L$,L$$)::CALL S
    AY("THE LETTER IS AFTER",L$
    ,"TRY AGAIN")
470 GOSUB 800
480 GOSUB 820
490 TRY=TRY+1
```

Guess a Musical Note

```
100 REM *********************
110 REM *  GUESS  A  NOTE  *
120 REM *********************
130 REM  BY  DAN  ROZLER
140 REM  99'ER  VERSION  2.13.1XB
150 CALL  CLEAR
160 CALL  CHAR(143,"0000FFFFFFFF
    0000"):::  CALL  COLOR(14,7,8)
170 CALL  CHAR(128,"939498F8F898
    9493")
180 CALL  CHAR(136,"080C0A0868F8
    F868")
190 DISPLAY  AT(5,8):"NAME  THE
    TONE"
200 CALL  HCHAR(3,8,143,19)::  CA
    LL  HCHAR(7,8,143,19)
210 CALL  SAY("NAME  THE1  TONE")
220 CALL  COLOR(13,12,8)
230 CALL  MAGNIFY(2)
240 DIM  NOTE$(8),FREQ(8),C(8),C
    ODE(8)
250 FOR  I=1  TO  8
260 READ  NOTE$(I),FREQ(I),C(I),
    CODE(I)
270 DATA  C,262,5,67
280 DATA  D,294,7,68
290 DATA  E,330,11,69
300 DATA  F,349,14,70
310 DATA  G,392,9,71
320 DATA  A,440,2,65
330 DATA  B,494,10,66
340 DATA  HIGHC,523,13,128
350 NEXT  I
360 CALL  SPGET("HIGHER",HI$)::
    HI$=SEG$(HI$,1,38)::  CALL  S
    PGET("C",CC$)
370 R=170  ::  COL=50  ::  B=0
380 FOR  I=1  TO  8
390 SHAPE=136
400 GOSUB  1060
410 NEXT  I
420 GOSUB  1120
430 R=170  ::  COL=110  ::  B=8
440 FOR  I=1  TO  8
450 SHAPE=CODE(I)
460 GOSUB  1060
470 NEXT  I
480 GOSUB  1120
490 CALL  DELSPRITE(ALL)
500 GOSUB  1120
510 CALL  CLEAR
520 CALL  SAY("IF  THE1  TONE  IS")
530 DISPLAY  AT(1,1):"TONE"
540 R=0
550 FOR  I=8  TO  1  STEP  -1
560 CALL  SPRITE(#I,CODE(I),C(I)
    ,10+R,24)
570 R=R+24
580 NEXT  I
590 CALL  SAY("THEN  YOU  ENTER")
600 DISPLAY  AT(1,6):"ENTER"
610 R=0
620 FOR  I=8  TO  1  STEP  -1
630 DISPLAY  AT(3+R,8):NOTES(I)
640 R=R+3
650 NEXT  I
660 RANDOMIZE
670 DISPLAY  AT(12,15):"WHAT  IS"
    ::  DISPLAY  AT(13,15):"THE
    TONE?"
680 CALL  SAY("WHAT  IS  THE1  TONE
    ")
690 J=INT(8*RND+1)
700 TRY=1
710 CALL  SOUND(2000,FREQ(J),0)
720 GOSUB  1120
730 GOSUB  1120
740 ACCEPT  AT(13,24)BEEP  VALIDA
    TE("ABCDEFGHI"):CHOICE$
750 IF  CHOICE$=NOTE$(J)THEN  900
    ELSE  760
760 IF  TRY=2  THEN  800  ELSE  770
770 CALL  SAY("SORRY  TRY  AGAIN")
780 TRY=TRY+1
790 GOTO  710
800 IF  J=8  THEN  CALL  SAY("THE1
    CORRECT  TONE  IS",HI$,,CC$)::
    GOTO  820
810 CALL  SPGET(NOTE$(J),N$)::  C
    ALL  SAY("THE  CORRECT  TONE  I
    S",N$)
820 CALL  SOUND(2000,FREQ(J),0)
830 FOR  COUNT=1  TO  6
840 CALL  SPRITE(#9,CODE(J),C(J)
    ,120,160)
850 GOSUB  1120
860 CALL  DELSPRITE(#9)
500 GOTO  350
510 CALL  SAY("TIME  IS  OUT")
520 CALL  MAGNIFY(2)
530 DISPLAY  AT(18,2)BEEP:"THE  C
    ORRECT  ANSWER  IS"
540 CALL  HCHAR(16,27,131,2)::  C
    ALL  HCHAR(20,27,132,2)::  CA
    LL  VCHAR(16,26,129,5)::  CAL
    L  VCHAR(16,29,130,5)
550 FOR  C=1  TO  6
560 CALL  DELSPRITE(#2)
570 FOR  P=1  TO  75  ::  NEXT  P
580 CALL  SPRITE(#2,N,16,132,210
    )
590 FOR  P=1  TO  75  ::  NEXT  P
600 NEXT  C
610 GOTO  700
620 DISPLAY  AT(18,9)BEEP:"*  VER
    Y  GOOD!  *"
630 CALL  MAGNIFY(2)::  CALL  SPRI
    TE(#1,N,13,7,1,0,10)
640 FOR  COUNT=1  TO  6
650 DISPLAY  AT(19,12)BEEP:"CORR
    ECT!"
660 FOR  PAUSE=1  TO  50  ::  NEXT  P
    AUSE
670 DISPLAY  AT(19,12):"  ::  FOR  PAUSE=1  TO  50  ::  N
    EXT  PAUSE
680 NEXT  COUNT
690 CALL  SAY("VERY  GOOD")
700 DISPLAY  AT(22,5):"DO  AGAIN-
    YES  OR  NO?"
710 CALL  SAY("DO  AGAIN  YES  OR  N
    O")
720 ACCEPT  AT(22,24)BEEP  VALIDA
    TE("YESNO")SIZE(3):ANS$
730 IF  SEG$(ANS$,1,1)="Y"  THEN
    210
740 CALL  DELSPRITE(ALL)
750 CALL  CLEAR
760 DISPLAY  AT(12,12):"GOODBYE"
770 CALL  SAY("GOODBYE")
780 RUN  "DSK1.LOAD"
790 STOP
800 FOR  DELAY=1  TO  200  ::  NEXT
    DELAY
810 RETURN
820 CALL  HCHAR(18,1,32,32)
830 CALL  HCHAR(19,1,32,32)
840 RETURN
```

Continued on p. 18

TI's Home Computer. The one that can grow with you.

No other home computer in this price range gives you more features, more ability to expand, or more fun.

Instantly useful. A true family computer. Only Texas Instruments gives you a choice of more than 80 Solid State Software™ cartridges. Programs for education. Information management. Entertainment. Computer programming. Finance. Word processing. And many more. Just plug them into your TI-99/4A Home Computer and you're ready to go.

Grows with you. Then, when you're ready to do more, you'll discover how easily your TI-99/4A Home Computer can expand with low-cost peripherals. Plug in more memory. Add disk drive or cassette storage, telecommunications, a speech synthesizer, or printer. It's the kind of flexibility you only expect in computers costing much, much more.

With an expanded system, you'll be able to use our more than 1300

diskette- and cassette-based programs. Take advantage of advanced languages. Get more programming flexibility. Or, connect to the outside world for weather forecasts, Dow Jones reports, even shop at home *electronically.*

Compare. Feature for feature, TI gives you more now. And more to grow with. Ask your TI dealer for information or call (800) 858-4565.

Creating useful products and services for you.

TEXAS INSTRUMENTS

THANK YOU

and simply return your B.A.R.C. BACK selection.

PLEASE HELP

IF YOU HAVE ALREADY ANSWERED OUR QUESTIONNAIRE . . . Please check here ■ and

Detach, fold and mail. See other side for instructions.

Think of it—

This 4-minute Questionnaire can *actually* impact the Home Computer revolution!!!

FOR ALL READERS

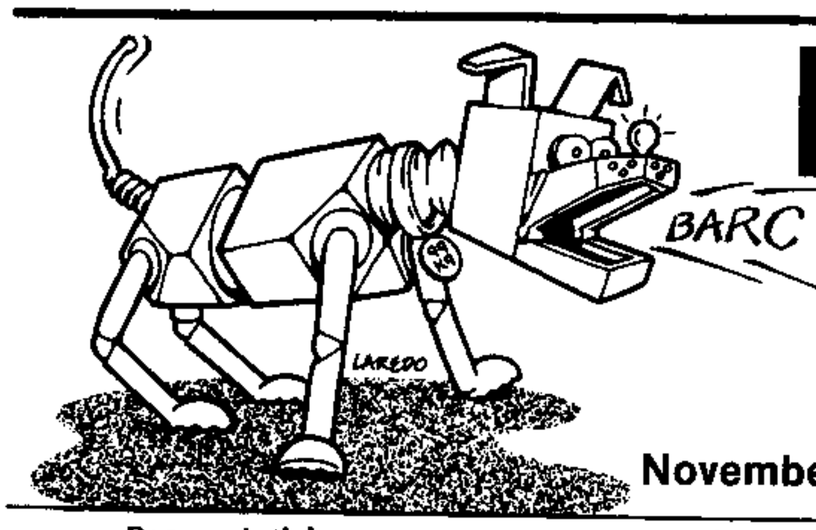
1. **Are you presently a subscriber?** ☐Yes ☐No
2. **If not, do you intend to become one within the next 3 months?** ☐Yes ☐No
3. **If not a subscriber, where did you get your copy?** ☐Newsstand ☐Supermarket ☐Bookstore ☐Airport ☐Users group ☐Computer store ☐Chain/department store ☐Borrowed from friend ☐Other place
4. **What category of articles do you enjoy the most?** ☐BASIC programming tutorials ☐System tutorials ☐Photo features & news items ☐Game programs ☐Education programs ☐Utility programs ☐Product reviews ☐LOGO articles
5. **How much total time do you spend with each issue?** ☐Less than 2 hours ☐2-4 hours ☐5-7 hours ☐8-10 hours ☐11-13 hours ☐over 14 hours
6. **How many other computer-related magazines do you currently read?** ☐None ☐1 ☐2-4 ☐5 or more
7. **Are you** ☐Male ☐Female ☐Under 16 years of age ☐16-20 ☐21-25 ☐26-30 ☐31-35 ☐36-40 ☐41-50 ☐over 50
8. **Are you a student?** ☐Yes ☐No
9. **What is your annual household income?** ☐Under $5000 ☐$5000-$9999 ☐$10,000-$14,999 ☐$15,000-$19,999 ☐$20,000-$24,999 ☐$25,000-$30,000 ☐over $30,000
10. **What is your ZIP code?** ☐☐☐☐☐

FOR READERS WHO DON'T YET HAVE A TI COMPUTER

1. **Do you intend to buy a TI computer?** ☐No ☐Yes (within 3 months) ☐Yes (within 3-6 months) ☐Yes (within 6-12 months)
2. **Which do you think you'll purchase?** ☐TI-99/4A Home Computer ☐Compact Computer 40
3. **What do you anticipate your primary use of a TI computer will be?** ☐Entertainment ☐Education ☐Computer literacy ☐Household management ☐Job-related homework ☐Business ☐Professional use

FOR PRESENT TEXAS INSTRUMENTS COMPUTER USERS

1. **Which system(s) do you currently own?** ☐99/4 ☐99/4A ☐CC-40
2. **What was your primary reason for buying it?** ☐Entertainment ☐Education ☐ Computer literacy ☐Household management ☐Job-related homework ☐Business ☐Professional use
3. **What was your primary reason for buying the Texas Instruments brand?** ☐Company name/reputation ☐Features for the money ☐16-bit microprocessor ☐Convinced by friends/relatives ☐Ease of use ☐Prior use in course or "Advantage Club"
4. **Which additional TI computer are you likely to purchase within the next 6 months?** ☐None ☐99/4A ☐CC-40
5. **What peripherals do you currently use?** ☐Cassette recorder ☐Disk controller & drive(s) ☐Peripheral Expansion Box ☐RS232 ☐32K Memory Expansion ☐TV ☐B/W monitor ☐Color Monitor ☐Speech Synthesizer ☐Joysticks ☐Printer ☐Modem ☐p-Code Card ☐Hex-bus Adapter ☐Wafertape Drive
6. **Put a CIRCLE around the above peripheral you are most likely to buy within the next 6 months.**
7. **Mark all TI language software you own or plan to buy within 6 months.** ☐Extended BASIC ☐99/4A Editor/Assembler ☐UCSD Pascal ☐LOGO ☐Forth ☐Mini Memory ☐Pilot ☐CC-40 Editor/Assembler
8. **How much money do you expect to spend within the next 12 months on your computer system?**
   - **Software** . . . . . ☐None ☐less than $30 ☐$30-50 ☐$51-100 ☐$101-250 ☐over $250
   - **Peripherals** . . . . . ☐None ☐less than $50 ☐$50-100 ☐$101-250 ☐$251-500 ☐over $500
   - **Books** . . . . . ☐None ☐less than $10 ☐$10-25 ☐$26-50 ☐over $50
   - **Blank tapes & disks** . . . . . ☐None ☐less than $15 ☐$15-35 ☐$36-75 ☐over $75
   - **Furniture, dust covers, & accessories** . . . ☐None ☐less than $25 ☐$25-100 ☐over $100
9. **How many software CARTRIDGES do you expect to purchase within the next 12 months?** ☐None ☐1-3 ☐4-7 ☐8-12 ☐over 12
10. **What % of the above CARTRIDGES will be for entertainment?** ☐0% ☐less than 25% ☐25-50% ☐51-75% ☐76-100%
11. **Circle above what % of the CARTRIDGES will be for education.**
12. **Have you purchased from any of our advertisers in the magazine within the last 6 months?** ☐No ☐Yes, Software ☐Yes, Peripherals ☐Yes, Books ☐Yes, Blank tapes & disks ☐Yes, Furniture, dust covers & accessories
13. **About how much money have you spent on the above purchases?** ☐less than $25 ☐$25-50 ☐$51-100 ☐$101-250 ☐$251-500 ☐$501-1000 ☐over $1000
14. **On the average, about how many program listings in each issue do you key into your computer and use?** ☐None ☐1 ☐2 or 3 ☐4 or more



B.A.R.C.* BACK

*(Best Article—Reader's Choice)

Let us know what you like by voting for your favorite article or program in this issue. The winning author will receive a bonus of $100.00

November, 1983

Part II

Have No Fear:

Assembly Language Won't Byte!

by Peter Lottrup

Catamarca 95
Accossuso (1640)
Buenos Aires, Argentina

In the first part of this series, we saw how to use the first three of the seven directives of the *Line-by-Line Assembler* included with the Mini Memory Command Cartridge and how to structure an Assembly Language program line. Now we will start to write small programs and in that way learn our first words of Assembly Language vocabulary. We will also learn how to END and execute a program once it has been typed in and how to save it on tape. Finally, we will learn a new assembler directive: SYM.

Displaying a Message

First we will write a small program that will display the words TEST PROGRAM in the center of the screen. Before we begin to do this though, we must remember a little about the display screen's structure. As we know from BASIC, the screen is divided into 24 rows and 32 columns—giving it 24 x 32 squares, or 768 positions. The top left corner is the first position, and the bottom right is position 768.

To display the message, you have to tell the computer several things. You must first tell it in which screen position you want the message to begin. This number can be in decimal or in hexadecimal (base sixteen) form; if it is in the latter, it must include the > sign first to indicate this. You will also have to tell the computer how long the message will be—exactly how many characters, including spaces. Finally, you will have to inform the computer what the address (or label) of the text is, so it will know where to find what it has to display. Your listing will look like this:

```
7D00 045B       LI R0,392
7D04 0354       LI R1,ST
7D08 0244       LI R2,12
7D0C A100       BLWP @>6028
7D10 0007 NN JMP NN
7D12 6100  ST   TEXT 'TEST PROGRAM'
```

(The first column of numbers is one memory location *address;* the second column is the present *contents* of that address, which at this stage could be any hexadecimal number.) Follow the instructions in part 1 of this series to enter the *Line-by-Line Assembler* and then begin entering these program lines.

Explanation of the Program

The first line means *load register number zero with the immediately following decimal value of 392.* The instruction LI (*Load Immediate*) simply means "place the following value in . . . " R0 specifies a certain *register,* which is defined in the Editor/Assembler manual as a memory word that serves a specific purpose. For the time being, just think of it as a box in the computer's memory where you place a value that you want the computer to remember. Seeing it in this way, you should understand from the first line of the program that we are telling the computer to load the value of 392 (the decimal position of the center of the screen) into "box" (register) number zero. We found this number in the following manner: The message had to appear in row 12, halfway down the screen. As each row has 32 characters, the value of the first position in that twelfth row was found by multiplying 12 x 32. Then, we added 8 characters to end up at the place where we wanted the message to begin.

"Note that the JMP instruction in Assembly Language is the same as the GOTO statement in BASIC; neither is subject to any condition."

The next line is quite similar to the one we have just seen: We are telling the computer to load into register number one the label of the place containing the text to be displayed. In reality, the computer converts the label, ST, into a number (or memory address) for its use. In this way, the computer knows where to find the text to display.

The third line is similar to the other two: We are telling the computer to load the value 12 into register number two. This will tell the computer how long the message is going to be.

The fourth line is the one that actually causes the message to be displayed on the screen. In this case, we are trying to write a message which is contained in the computer's memory (CPU RAM) to the screen (VDP RAM). There is an Assembly Language routine which does this for us: the VMBW, or VDP Multiple Byte Write routine. As the name tells us, this routine takes bytes from CPU memory and writes them into specified locations in the VDP memory (on the screen). If we were using the Editor/Assembler, we could use the *mnemonic* VMBW in the following line:

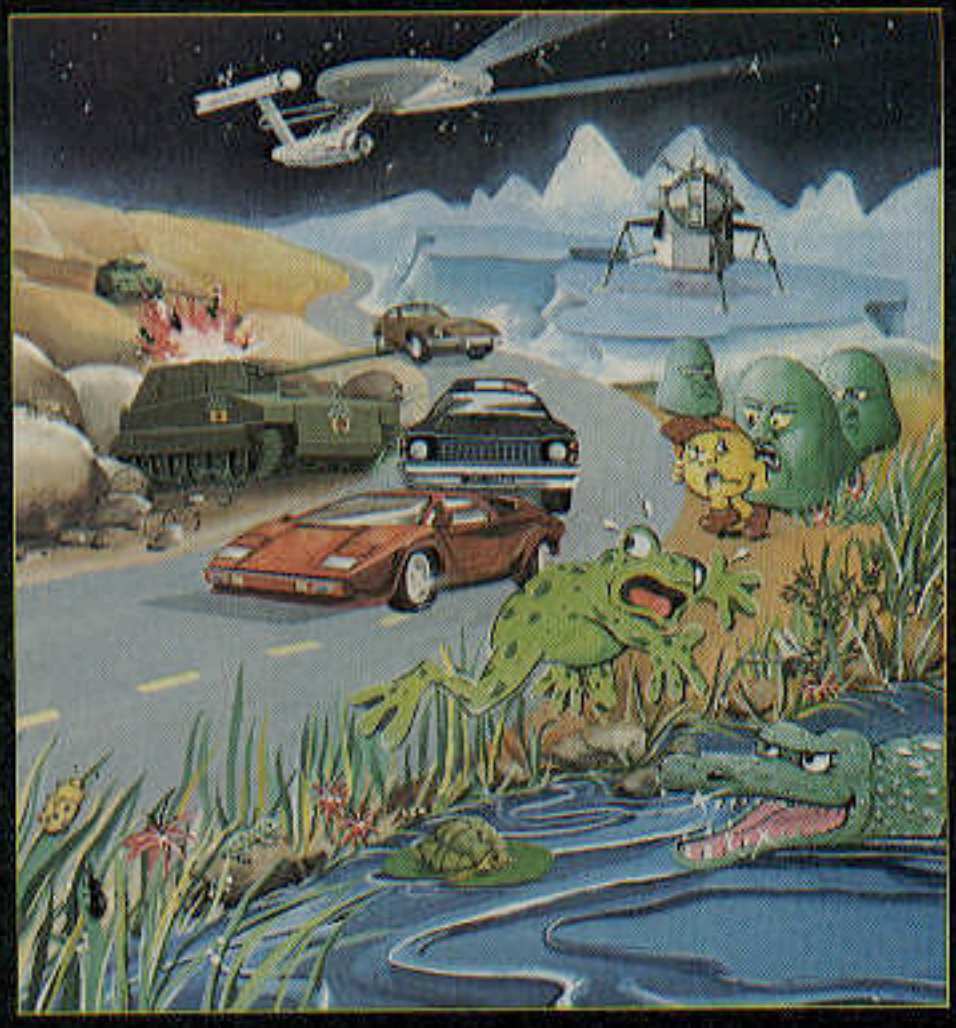
```
BLWP @VMBW
```

BLWP is similar to GOSUB in BASIC. It tells the computer to branch and execute the routine (VMBW) we are specifying. We can't write the line like this when using the *Line-by-Line Assembler* unless we add some extra lines to the program. This is because this assembler does not recognize what "VMBW" stands for. But from the Mini Memory manual (page 35), we know that the VMBW routine is at hexadecimal address >6028. So in the *Line-by-Line Assembler,* if we write:

```
BLWP @>6028
```

the line will be correct. Then, using the information we have already placed into the computer's memory, the message will be displayed on the screen. The VMBW routine beginning at address >6028 expects to find certain values in certain registers: In Register 0, it expects to find

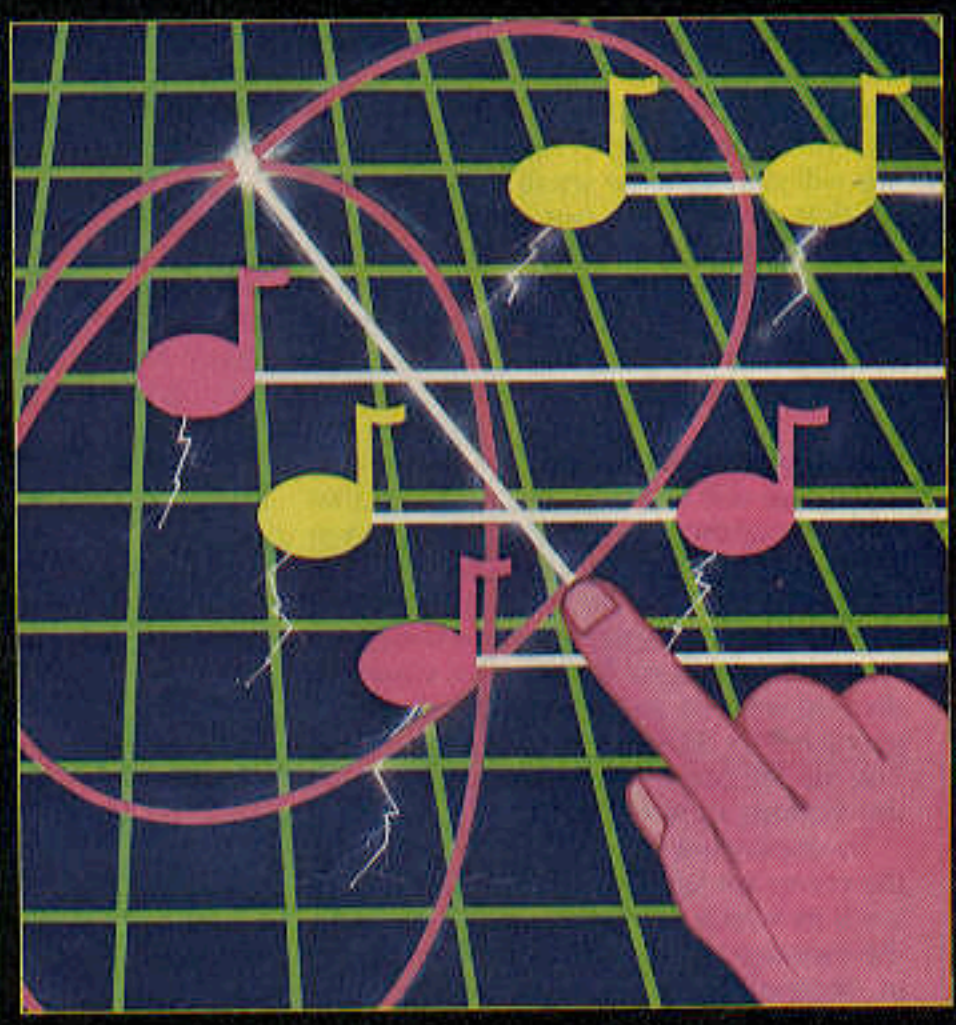
Continued on p. 15



Dealers
Don't make your customers look elsewhere. Contact us for Dealership/Distributorship opportunities. Make Norton Software your #1 software line.

Customers
Most TI Dealers are already carrying or will assist you in obtaining our quality software. If you lack a TI dealer, look for our logo appearing elsewhere in this magazine.

Quality software requires quality packaging. The above illustrations represent part of our artistic packaging design.



THE NAME OF THE GAME™

1. Attack-man
2. Cross Country Car Rally
3. Graphics Package
4. Killer Caterpillar
5. Lunar Lander
6. Music Synthesizer
7. 3-D Startrek
8. Super Frogger
9. Tank

Norton software ltd.™

P.O. Box 575, Picton, Ontario, Canada K0K 2T0

Have No Fear . . . from p. 13

the first screen *location* to use in its display. In Register 1, it expects to find the *address* of the message to display (in this case, a label which it interprets as the address >7D12). In Register 2, it expects to find the *length* of the string to display.

If we just stop the program there, however, the message flashes on and off before we can even see it. So the next line creates an endless loop, causing the computer to jump (JMP) to that same line over and over. Note that the JMP instruction in Assembly Language is the same as the GOTO statement in BASIC; neither is subject to any condition.

Finally, the only thing we need to include in our program is the text we want to display using the TEXT directive, which we discussed in part one. This text must be labeled with the same label we used to refer to it in the second line of the program: ST.

Running the Program

When you have finished typing in the lines as they appear in the program listing above, you are ready to end. Type a space, type END, and press [ENTER]. The following message will be displayed:

```
0000 UNRESOLVED REFERENCES
```

This means that all the labels mentioned in the program have been used in some way or other—that is, that you have not mentioned a label in an operand field which did not exist in a label field. For instance, imagine that we had not included the line with the TEXT directive and the ST label. Then, the second line of the program would be mentioning a label which did not exist in the program. If you ended the program at that point, the message displayed would be:

```
0001 UNRESOLVED REFERENCES
```

Do not end the program if all the references have not been resolved because it will not run correctly. If you press any key except [ENTER], you will return to the memory location where you left off. Then you can use the SYM directive (discussed below) to find the unresolved label or labels. If you do get the message:

```
0000 UNRESOLVED REFERENCES
```

meaning that all labels are correct, then press [ENTER] twice and you will return to the Mini Memory main menu.

Now let's look at how to RUN our program using EASY BUG. (Next time, we will learn how to SAVE the program by name so that you will be able to run it directly, like the *LINES* demonstration program.) Return to the title screen by pressing [QUIT], and choose the EASY BUG option. Press any key to skip the menu, and when the question mark appears, type:

```
?E7D00
```

and press [ENTER]. This means *execute the Assembly Language program which starts at memory location number 7D00* (which is where we began to write our program). You will see that the message we wanted to show is displayed immediately on the center of the screen. Note that the only way to regain control of the computer in this program is by turning off the power, and even if we do this, the program is still kept in Mini Memory.

If you now want to save the program on cassette, select EASY BUG, skip the title screen and type:

```
?S
```

S means *save the contents in memory to tape.* The computer will then ask:

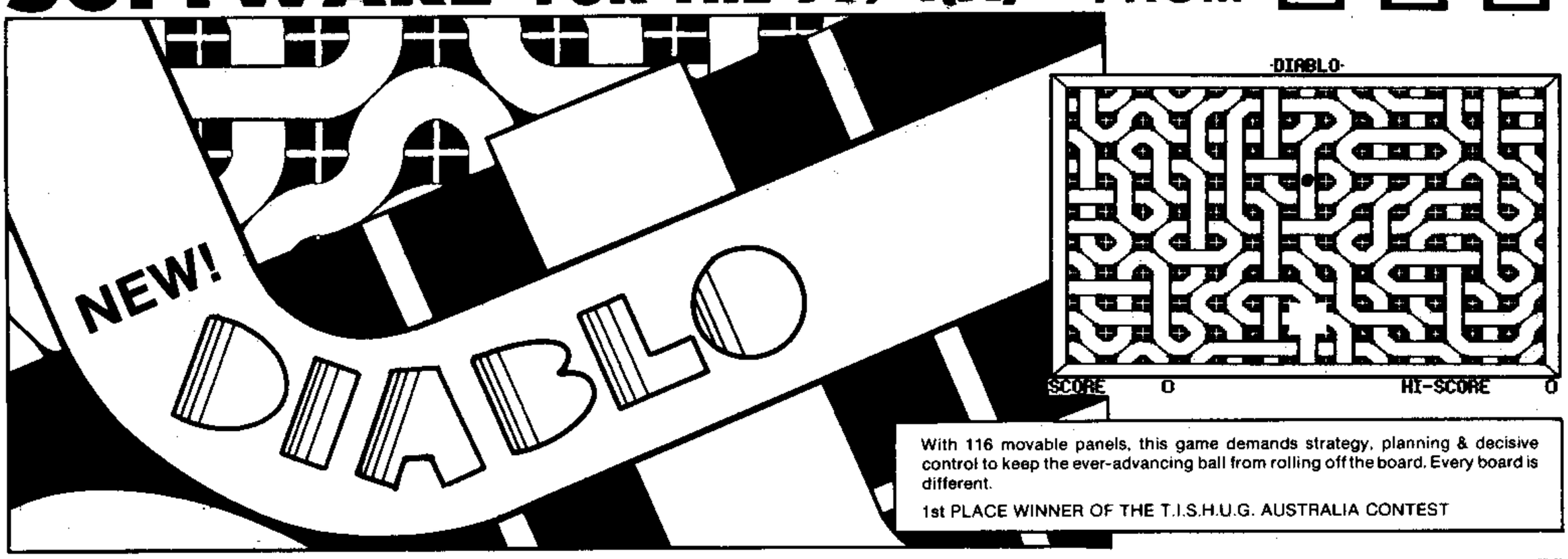
```
FROM?
```

With this prompt, the computer wants to know FROM what memory location you want to begin recording. To be sure that you get everything, it is best to type 7000, as it says in the manual. It is not necessary to include the > sign. The computer will then ask:

```
TO?
```

Continued on p. 17

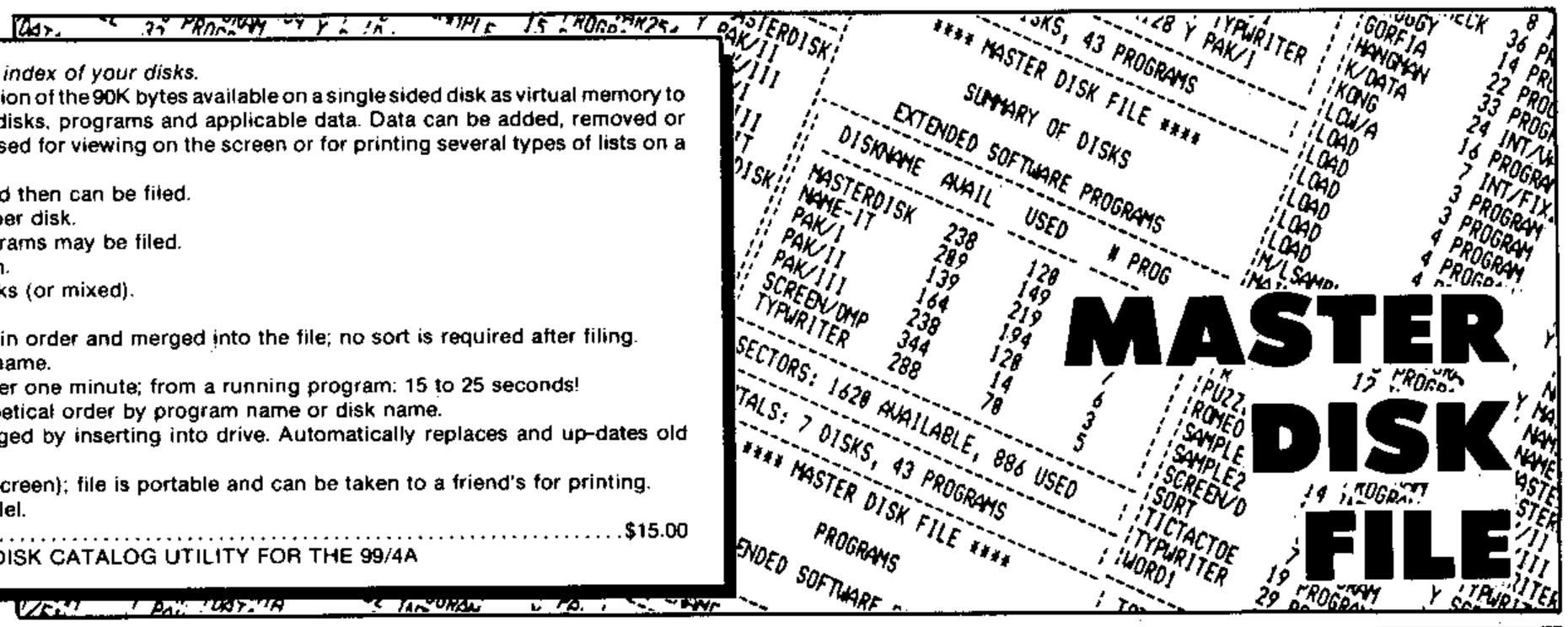
SOFTWARE FOR THE 99/4(A) FROM ESC
EXTENDED SOFTWARE COMPANY



With 116 movable panels, this game demands strategy, planning & decisive control to keep the ever-advancing ball from rolling off the board. Every board is different.

1st PLACE WINNER OF THE T.I.S.H.U.G. AUSTRALIA CONTEST

MASTER DISK FILE — *A master index of your disks.*
MASTER DISK FILE uses a large portion of the 90K bytes available on a single sided disk as virtual memory to create and store a perpetual file of disks, programs and applicable data. Data can be added, removed or updated. The file may then be accessed for viewing on the screen or for printing several types of lists on a printer.
Insert disk into drive, it is catalogued then can be filed.
Reads up to 95 programs and files per disk.
Maximum of 120 disks or 1100 programs may be filed.
Does not require memory expansion.
Supports single or double sided disks (or mixed).
Supports single or multiple drives.
Programs are catalogued from disk in order and merged into the file; no sort is required after filing.
SEARCH for disks or programs by name.
Look-up time from a cold start: under one minute; from a running program: 15 to 25 seconds!
List on screen or a printer in alphabetical order by program name or disk name.
Up-date disks that have been changed by inserting into drive. Automatically replaces and up-dates old information.
May be used without a printer (on screen); file is portable and can be taken to a friend's for printing.
Supports any printer: serial or parallel.
Diskette (only) $15.00
GUARANTEED TO BE THE BEST DISK CATALOG UTILITY FOR THE 99/4A



GAMES PAK/III

KONG - must fight his way past barrels, ladders & trap doors to save Roxanne from Igor's bomb. Six different screens. Joystick req. $15.00

BOUNCER — bounds from one trampoline to another, scoring points while avoiding the arrows which will burst him. 6 screens. Joysticks req. $15.00

ROMEO — must traverse a desert, stream (alligators & sharks) & cavern to obtain a fitting reward. Joysticks req. $15.00

GAMES PAK/II

ARTILLERY — The opposing force must be destroyed by determining angle and force of each shot. An ever-changing wind complicates matters. 1 or 2 players $9.95

DE-CYPHER — An encrypted message is displayed and guesses change all corresponding letters to the guess. Comes with 50 messages which can be changed $9.95

PUZZLE 15 — Slide alphabetic squares into order. Multiple squares may be moved. Computer keeps track of moves & has replay feature $9.95

FLIP CHECKERS — Sandwich opponent's piece to flip it to your color. 1 or 2 players. Real-time clock. Fast computer decisons. Joysticks req. $9.95

GAMES PAK/I

FROGGY — Jump FROGGY across 10 lanes of traffic, then across 6 logs. Keyboard or joysticks $9.95

EXTENDED BASEBALL - Joystick control of the pitcher & batter, and individual batting averages. Multi-base & multi-runner plays. Joysticks req. $9.95

GORFIA PESTULITIS - Joystick control of a laser sight or inertia influenced space mines to shoot down the invading Gorfians. Joysticks req. $9.95

EXTENDED HANGMAN - Quick graphics, music, color, speech (optional) and sound are added to keep the players entertained. Includes 580 words of 4 to 9 letters in length in easy, medium and difficult groups $9.95

TIC-TAC-TOE — Quick set-up and quick decision making at four levels of difficulty. The levels avoid the frustration of the novice never having a chance to win $9.95

TYPWRITER
A 16K WORD PROCESSOR on Cassette or Disk
Any Input/Output storage of text — disk, cassette, cassette input/disk output, or vice versa.
Selectable *right justify*.
Complete text *Editing* — by cursor control; including insert & delete lines, partial print, printer halt or abort without text loss, page FWD & BKWD, and more.
Complete *Software Control* of Printer (depending upon its capabilities) — for enhanced print, underlining, formatting, 28 to 254 characters per print line, etc.
No Special Equipment — monitor, console, Extended Basic module, C or D, printer.
Comes with a 20 page instruction booklet.
Cassette $32.00 Diskette $35.00

SCREEN/DUMP — Print the screen to a dot-addressable, 8-bit printer. Does not require extra memory! Disk version is simple to use. Cassette version requires mild programming knowledge $12.00

NAME-IT
DATA BASE for: *Mail Lists, Labels, Files*
Records: 250 records per diskette consisting of up to nine 28-character items per record.
Prompts: user designated prompts.
Complete *File Sort*: 250 records in *100 Seconds*.
Search; Pre-set; print labels & lists.
Includes a FORM LETTER program that uses NAME-IT data in TYPWRITER generated form letters
Cassette version differs from disk version.
Cassette $32.00 Diskette $35.00

NOTE: Should you decide to up-grade to the TI-WRITER module, TYPWRITER and NAME-IT data can be converted for use by that module. NAME-IT alone, will generate 250 TI-WRITER form letter records.
TI-WRITER is copyrighted software of Texas Instr.

A detailed catalog is available free. Circle "FREE" on the order form or send a letter or postcard.

ORDER FORM
EXTENDED BASIC MODULE REQUIRED FOR ALL PROGRAMS

CATALOG	FREE		$ FREE
GAMES PAK/I (Froggy, Extended Baseball, Gorfia Pestulitis, Extended Hangman, Tic-Tac-Toe)	$26.95	(C or D)	$
GAMES PAK/II (Artillery, De-Cypher, Puzzle 15, Flip Checkers)	$26.95	(C or D)	$
GAMES PAK/III (Kong, Bouncer, Romeo)	$26.95	(C or D)	$
TYPWRITER (word processor)	$32.00	(C price)	$
	$35.00	(D price)	$
NAME-IT (data base/mail list)	$32.00	(C price)	$
	$35.00	(D price)	$
SCREEN/DUMP (printer required)	$12.00	(C or D)	$
MASTER DISK FILE	$15.00	(D only)	$
INDIVIDUAL GAMES:			
DIABLO	$19.95	(C or D)	$

☐ Froggy ☐ Extended Baseball ☐ Gorfia Pestulitis ☐ Extended Hangman ☐ Bouncer ☐ Kong
☐ Tic-Tac-Toe ☐ Artillery ☐ De-Cypher ☐ Puzzle 15 ☐ Romeo ☐ Flip Checkers

Total Individual Games: (C or D) $

Shipping & Handling (1st Class Mail): Included
Sales Tax: Included
Add $2.00 if C.O.D. (U.S. Mail Only): $
Check or money order or C.O.D. Total: $

PLEASE SPECIFY CASSETTE OR DISK

Send this form or a substitute with check or money order to:
Extended Software Company
11987 Cedarcreek Drive
Cincinnati, Ohio 45240
(513) 825-6645

SATISFACTION GUARANTEED
IF YOU ARE NOT COMPLETELY SATISFIELD, YOU MAY RETURN THE PROGRAMS (and instructions) WITHIN 15 DAYS FOR A FULL REFUND OF YOUR PURCHASE PRICE.

Dealer and programmer inquiries welcome.

Cut Here (may be copied or substituted)

Have No Fear . . . from p. 15

With this prompt, the computer wants to know the last memory address to save. Here type 7FFF, again to be sure you get everything. This means that you are storing the entire contents of the Mini Memory cartridge on tape. When you have typed 7FFF and pressed [ENTER], the usual procedure to save on tape will follow.

To load the contents later on, select EASY BUG and then choose the L command. The computer will then respond with the standard sequence of instructions for loading from cassette.

When you have finished loading the program and the question mark appears, type E (*execute Assembly Language program*), followed by the hexadecimal address where the program begins—in this case >7D00.

The SYM (Symbol Table) Directive

As we have seen, it is very possible to get the message that there are one or more unresolved references when you END a program. If this happens, it is often very difficult to find those labels, especially when the program is long. In these cases, use the SYM assembler directive. It is simple to use: Type a space to skip the label field, and type SYM. Then press [ENTER]. All resolved and unresolved references will be displayed, up to a maximum of 32 unresolved references. Then you can see exactly where the problem lies.

Try this out by typing the program to display text once again, except for the line where you include the text. When you end the program, you will get the message:

```
0001 UNRESOLVED REFERENCES
```

as we have already seen. Press any key except [ENTER] to return to the assembler, and then type a space and SYM. Press [ENTER], and the following will be displayed:

```
RESOLVED REFERENCES
NN-7D10
UNRESOLVED REFERENCES (WORD)
ST-7D06
```

RESOLVED REFERENCES are all the labels which have been given a value, or defined. UNRESOLVED REFERENCES (WORD) are all the labels which are referenced in any instructions except JUMP instructions. (In our case, the ST label is displayed, together with its memory location, indicating that our problem lies there.)
A third possibility is:

```
UNRESOLVED REFERENCE (JUMP)
```

This is any label which has been referenced in a JUMP instruction but has not been given a value or defined. For this situation, add this line:

```
7D12 xxxx     JMP CV
```

and then type SYM again. It will show you the resolved labels, the unresolved word label, and the unresolved jump label (CV-7D12). In this way, it is easy to find out which labels are unreferenced in the program. Using the AORG directive, it is easy to move back to any line and correct it, or add anything missing.

Conclusion

In this part of the series we have learned many new things: how to type in, run and save short programs, assembler instructions, and the assembler directives SYM and END.

Now that you are starting to understand the structure of Assembly Language programs, we are ready to look at a whole list of important Assembly Language instructions so that you can start trying out your own programs. In the next part of this series, we will also start saving programs by name, so we can run them directly.

You should now be able to experiment a bit by writing your own programs, based on what you have learned in this article and prior knowledge. You will see that the dark cloud hanging over Assembly Language is beginning to disappear.

Creative . . . from p. 11

```
 900 GOSUB 1120
     NEXT COUNT
     GOTO 920
     DISPLAY AT(15,15):"CORRECT!
     !"
 910 CALL SAY("THAT IS CORRECT V
     ERY GOOD")
 920 DISPLAY AT(19,15):"TRY AGAI
     N?" :: DISPLAY AT(20,15):"(
     YES OR NO)"
 930 CALL SAY("DO YOU WANT TO TR
     Y AGAIN")
 940 ACCEPT AT(19,25)BEEP VALIDA
     TE("YESNO")SIZE(3):ANS$
 950 IF SEG$(ANS$,1,1)<>"Y" THEN
     1000
 960 FOR E=1 TO 24
 970 CALL HCHAR(E,15,32,16)
 980 NEXT E
 990 GOTO 660
1000 CALL DELSPRITE(ALL)
1010 CALL CLEAR
1020 CALL SAY("GOODBYE")
1030 DISPLAY AT(12,12):"GOODBYE"
1040 RUN "DSK1.LOAD"
1050 STOP
1060 N=B+I
1070 CALL SPRITE(#N,SHAPE,C(I),R
     -10,COL+10)
1080 CALL SOUND(1000,FREQ(I),0)
1090 FOR PAUSE=1 TO 225 :: NEXT
     PAUSE
1100 R=R-10 :: COL=COL+10
1110 RETURN
1120 FOR PAUSE=1 TO 200 :: NEXT
     PAUSE
1130 RETURN
1140 END
```

Tell the Time

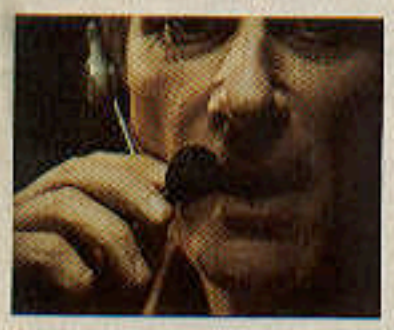
```
 100 REM *********************
 110 REM *  TELL THE TIME  *
 120 REM *********************
 130 REM  BY DAN ROZLER
 140 REM 99'ER VERSION 2.13.1XB
 150 CALL MAGNIFY(2)
 160 DIM CODE(12),ROW(12),COL(12
     ),CDE(12),SHR(12),SHC(12),B
     HR(12),BHC(12)
 170 CALL CHAR(128,"4649494949494
     946")
 180 CALL CHAR(129,"2424242424242
     424")
 190 CALL CHAR(130,"8C9281818284
     8B9F")
 200 CALL CHAR(132,"506A4D4949494
     949")
 210 CALL CHAR(131,"4242427E424242
     42")
 220 CALL CHAR(136,"000000FFFF000
     000):: CALL COLOR(14,7,8)
 230 CALL CHAR(97,"1830606C080000
     00")
 240 CALL CHAR(98,"0103060C183060
     C0")
 250 CALL CHAR(99,"000000FFFF0000
     00")
 260 CALL CHAR(100,"C060301808C06
     030")
 270 CALL CHAR(101,"80C060301808C
     060")
 280 CALL CLEAR
 290 CALL CHAR(102,"1818181818181
     818")
 300 CALL CHAR(103,"0000000010306
     0C1")
 310 CALL CHAR(104,"6030180C06030
     100")
 320 FOR I=1 TO 12
 330 READ CDE(I),SHR(I),SHC(I),B
     HR(I),BHC(I)
 340 DATA 97,8,25,52,192
 350 DATA 98,8,25,52,192
 360 DATA 99,9,25,60,192
 370 DATA 100,10,25,72,192
 380 DATA 101,11,25,80,192
 390 DATA 102,11,24,77,180
 400 DATA 103,10,23,76,168
 410 DATA 98,10,23,72,168
 420 DATA 99,9,23,60,168
 430 DATA 101,8,23,52,168
 440 DATA 104,7,23,44,168
 450 DATA 102,7,24,43,180
 460 NEXT I
 470 FOR I=1 TO 12
 480 READ CODE(I),ROW(I),COL(I)
 490 DATA 49,32,196
 500 DATA 50,44,204
 510 DATA 51,60,204
 520 DATA 52,76,204
 530 DATA 53,92,196
 540 DATA 54,108,180
 550 DATA 55,92,164
 560 DATA 56,76,156
 570 DATA 57,60,156
 580 DATA 128,44,156
 590 DATA 129,28,164
 600 DATA 130,20,180
 610 NEXT I
 620 GOSUB 1300
 630 DISPLAY AT(1,1):"WHAT TIME
     IS IT?"
 640 CALL HCHAR(2,3,136,16)
 650 CALL SAY("WHAT TIME IS IT")
 660 CALL SPRITE(#20,131,7,43,40
     )
 670 CALL SAY("THIS IS THE SMALL
     HAND")
 680 DISPLAY AT(3,3):"HOUR" :: D
     ISPLAY AT(4,3):"HAND"
 690 CALL SPRITE(#21,132,13,18,1
     04)
 700 CALL SAY("THIS IS THE LARGE
     HAND")
 710 DISPLAY AT(6,10):"MINUTE" ::
     DISPLAY AT(7,11):"HAND"
 720 CALL HCHAR(8,3,136,16)
     CALL HCHAR(14,3,136,16)
 730 RANDOMIZE
 740 SH=INT(12*RND+1)
 750 BH=INT(12*RND+1)
 760 IF BH=12 THEN Z=SH :: V=0 ::
     GOTO 800
 770 IF SH=1 AND BH>6 THEN Z=12
     :: V=BH :: GOTO 800
 780 IF BH>6 THEN Z=SH-1 :: V=BH
     :: GOTO 800
 790 V=BH :: Z=SH
 800 TRY=1
 810 DISPLAY AT(9,1):"HOUR HAND
     IS" :: DISPLAY AT(10,1):"ON
     THE "; SH
 820 CALL HCHAR(SHR(SH),SHC(SH),
     CDE(SH)) :: CALL SPRITE(#SH,
     131,7,ROW(SH),COL(SH))
 830 FOR PAUSE=1 TO 300 :: NEXT
     PAUSE
```

Continued on p 78

"TI owners! This microphone is going to revolutionize the way you play video games."

Announcing the MBX™ Expansion System developed by Milton Bradley for your TI Home Computer. This amazing new peripheral actually allows the computer to respond to the sound of your voice!

The complete MBX™ Expansion System:



THIS MICROPHONE lets you control the action on the screen. The process is called voice recognition. It recognizes and responds to *your* voice alone. You talk into the microphone, and the graphics on the screen do what you tell them to do. Whether it's a baseball player catching a fly or a turtle moving along its trail, they will obey your spoken command. It's uncanny the way they recognize your voice and add a whole new dimension to game play.

THIS KEYPAD has 64 positions with overlays which enable you to interact with the video screen. The MBX Expander unit also gives you outstanding speech synthesis. The computer speaks to you in a realistic, well modulated voice. You talk to the computer, and the computer talks to you.

THIS JOYSTICK is a precision engineered pistol-grip, analog control that allows 360° rotation plus front-to-back and left-to-right proportional movements. The result is total maneuverability of screen graphics. It's amazing!

The MBX Joystick was designed in a unique pistol-grip configuration to ensure comfort and eliminate joystick fatigue. It also features auxiliary control buttons and a quick-action trigger button to enhance fast play capability.

Incredible new software, too!

Here are just three of the specially designed cartridges for the MBX System.

CHAMPIONSHIP BASEBALL™ with voice recognition is the most realistic baseball game ever devised. It has just won the coveted Gold Joystick award as the most innovative game of the year.

TERRY TURTLE'S ADVENTURE™ is part of the BRIGHT BEGINNINGS™ series for younger children that combines fun with learning and uses a fascinating game to aid in elementary programming.

I'M HIDING™ is another in the BRIGHT BEGINNINGS™ series which uses voice recognition to help your child learn shapes, colors, and size relationships through game play.

The amazing MBX™ Expansion System

Available now at your TI dealers

Let's Build America

by Domenick C. Flotta
1140 Balthaser St.
Harrisburg, PA 17112

For those of us who learned our American geography on a jigsaw puzzle of the fifty states, *Let's Build America* will strike a familiar chord. In this game you generate a computer graphics map of the United States by typing in the names of the states. Spell in C-A-L-I-F-O-R-N-I-A, and a bright red replica of the Golden State appears on the left side of your screen. Tap in O-R-E-G-O-N, and a slightly distorted representation of the Beaver State appears to the north. Before you dismiss this as an educational exercise for the kids, try naming all fifty states. Are you really quite sure whether it's Vermont or New Hampshire that's further east? If you want even more of a challenge, you can answer [Y] when the prompt CAPITAL NAME? appears on the screen. You can then rack your memory to come up with the (correctly spelled) names of such cities as Montpelier, Cheyenne, and Tallahassee.

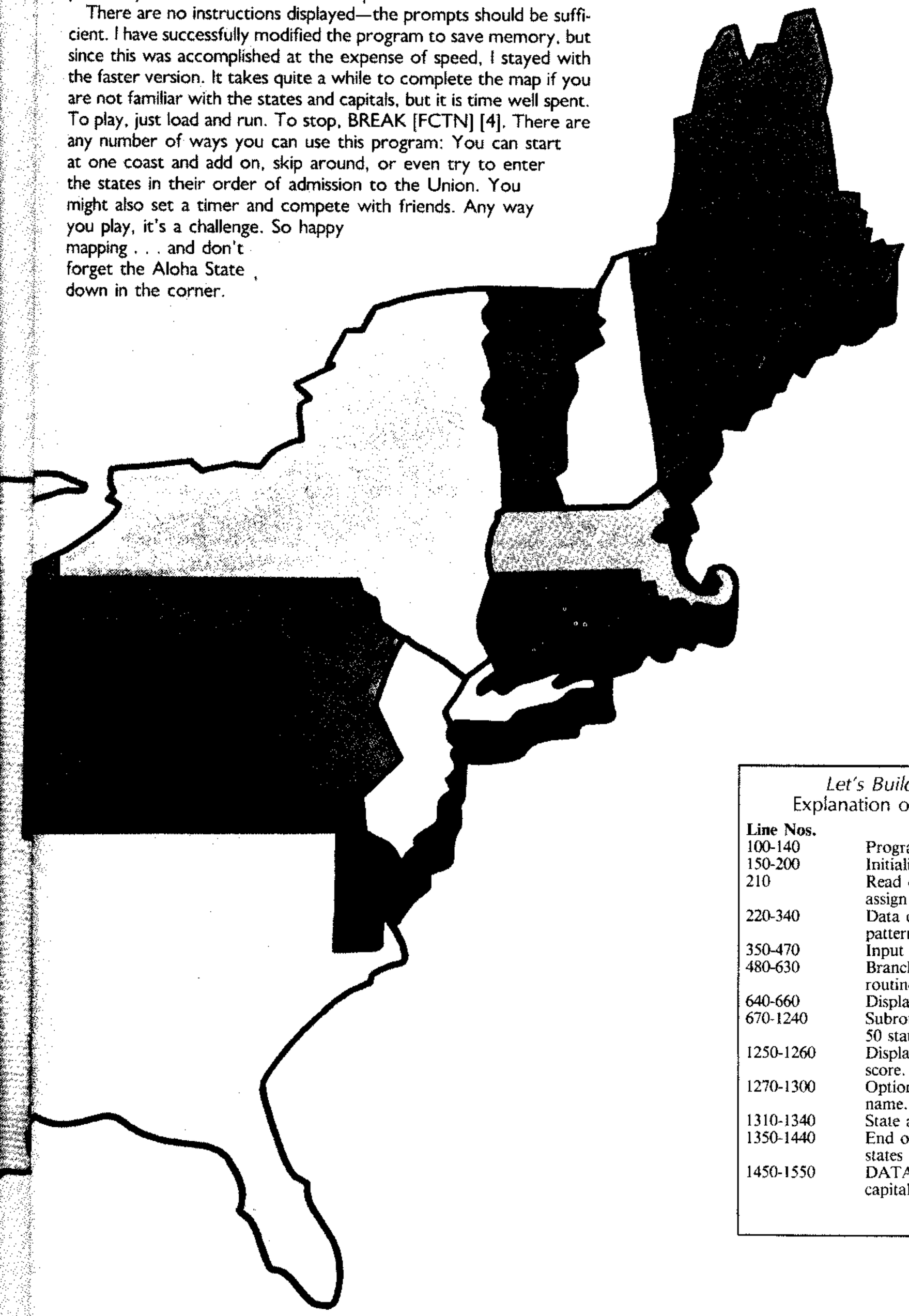
If the city you name is incorrect, it is counted as an attempt, and you will be prompted to try again. You are then asked to enter [Y] or [N] again for the capital re-try. The program will display the score of total attempts and total correct. Each state entered will be displayed on the screen until either the entire U.S.A. is built or the user quits by breaking. The states and capitals are scored separately. For each, the best score possible is 50 attempts and 50 right. When you complete the map correctly, you will see a display of the American flag and hear a bar or two of "God Bless America" as your reward.

The map is not a perfect simulation, but it is close enough for the purposes intended in this program. I have used Dr. Borden D. Dent's method of mapping the interior states ("Color Mapping," Vol. 1, No. 6 of *99'er HCM*), but the border states and Hawaii were developed by using sprites. The flag contains only 49 stars;

I was not able to solve that problem, but I thought it turned out beautifully nonetheless. I had to eliminate some other enhancements because I was running out of memory, but I retained the best and the most necessary.

All errors or duplicate attempts of the same state are indicated by a beep and a display of WRONG NAME or ALREADY USED (followed by the state name). A wrong attempt is added to the total of "wrong" entries, but on duplicate entries, the attempt is not added. After all, it is easy to forget a state which may have been entered previously—we must have some compassion!

There are no instructions displayed—the prompts should be sufficient. I have successfully modified the program to save memory, but since this was accomplished at the expense of speed, I stayed with the faster version. It takes quite a while to complete the map if you are not familiar with the states and capitals, but it is time well spent. To play, just load and run. To stop, BREAK [FCTN] [4]. There are any number of ways you can use this program: You can start at one coast and add on, skip around, or even try to enter the states in their order of admission to the Union. You might also set a timer and compete with friends. Any way you play, it's a challenge. So happy mapping . . . and don't forget the Aloha State down in the corner.



Let's Build America
Explanation of the Program

Line Nos.	
100-140	Program Header.
150-200	Initialize variables.
210	Read character patterns and assign graphics characters.
220-340	Data containing graphics patterns.
350-470	Input state to be displayed.
480-630	Branch to selected state routine.
640-660	Display score.
670-1240	Subroutines to display the 50 states.
1250-1260	Display error message and score.
1270-1300	Option to input capital name.
1310-1340	State already used.
1350-1440	End of game after all 50 states entered.
1450-1550	DATA for all 50 states and capitals.

Continued on p. 52

SQUEEZING the Most Out of TI BASIC

"I experimented to determine just how slow the various operations and subroutines were."

by John T. Dow
6360 Caton
Pittsburgh, PA 15217

Over a year ago I developed a flight simulation program in console BASIC for the TI-99/4A and TI-99/4. One of my primary considerations in writing the program was to make it as realistic as possible. This meant that the display and controls should be as much like those of a typical airplane as I could manage, and that the program should respond reasonably rapidly to the user's control changes. To accomplish this high degree of realism and responsiveness in the program, I had to become very familiar with the features and limitations of the computer and the BASIC language.

The Program's Features

To give you an idea of just how much can be done in BASIC, here is a brief description of the flight simulation program:

1) The display shows the instrument panel of an aircraft with 10 dials, 11 lights, and 6 digits. The resolution on the dials is 9 degrees. For example, each 9 degrees on the altimeter represents a 25 foot increment.
2) The plane is flown primarily by the joystick, but 21 keys (the digits 1-9 and the letters A-L) select options and control the plane, e.g., select fuel tanks, turn the wing leveler on and off, change power, and so forth.
3) Options include take off, landing, short field landing, and instruments approach (with glide slope and localizer needles, and with outer and middle marker lights)—all with the desired degree of wind and turbulence.
4) The simulation is realistic in that the plane's performance is affected by altitude, and control sensitivity is affected by airspeed.
5) When the "airplane" is in sight of the airport, a display digit above the instrument panel shows the distance and direction to the runway touchdown point. (The airport itself is not actually displayed.)
6) Crashes result from landing either too hard, off the runway, or at an unacceptable angle to the runway, from exceeding safe speeds (with or without flaps), from pulling too many Gs, or from flying low enough to hit obstacles.
7) Sound effects include an engine noise that varies with speed and power; stall warning and stall; falling and crash sounds; and sirens after a crash.
8) Performing all of these feats, the program still responds to your controls on the average of once every .75 seconds. And it operates in real time; that is, if the dial shows you climbing at 500 feet/minute, the altimeter will go up 500 feet every actual minute.

The Computer and the Language

The 99/4A and console BASIC clearly have some strong points and some not-so-strong points. For instance, the 99/4A's color and high-resolution graphics are flexible and very easy to use, and its ability to create simultaneous tones and noises with independently controlled intensities is particularly useful in a flight simulation program. I did find, however, that the editing features of BASIC are slow if the program is large. The precision of BASIC's calculations is impressive, but unfortunately, this robs the machine of much of its speed. To have been able to use integer arithmetic might have helped execution time significantly, although I understand that the split between CPU and VDP memory partially accounts for both the low price of the machine and for its lack of speed when VDP memory is used heavily, as it is with BASIC. I would have appreciated better documentation on how best to use BASIC. Both experienced and novice programmers should know, for instance, that it is not necessary to use PEEK and POKE with strange addresses to make things happen. But on balance, both the machine and the BASIC language turned in an impressive performance.

Programming the Display

Console BASIC cannot provide sprites, as Extended BASIC can, nor can it update displays as rapidly as an assembly language program can. However, with console BASIC it is possible to achieve as much detail in a graphics display as the monitor (or TV set) permits. When defining characters, you can turn individual pixels on or off. Each character, 8 bits by 8 bits, can be only two colors, but with some planning you can build a display from characters in such a way that it appears to be a homogeneous whole.

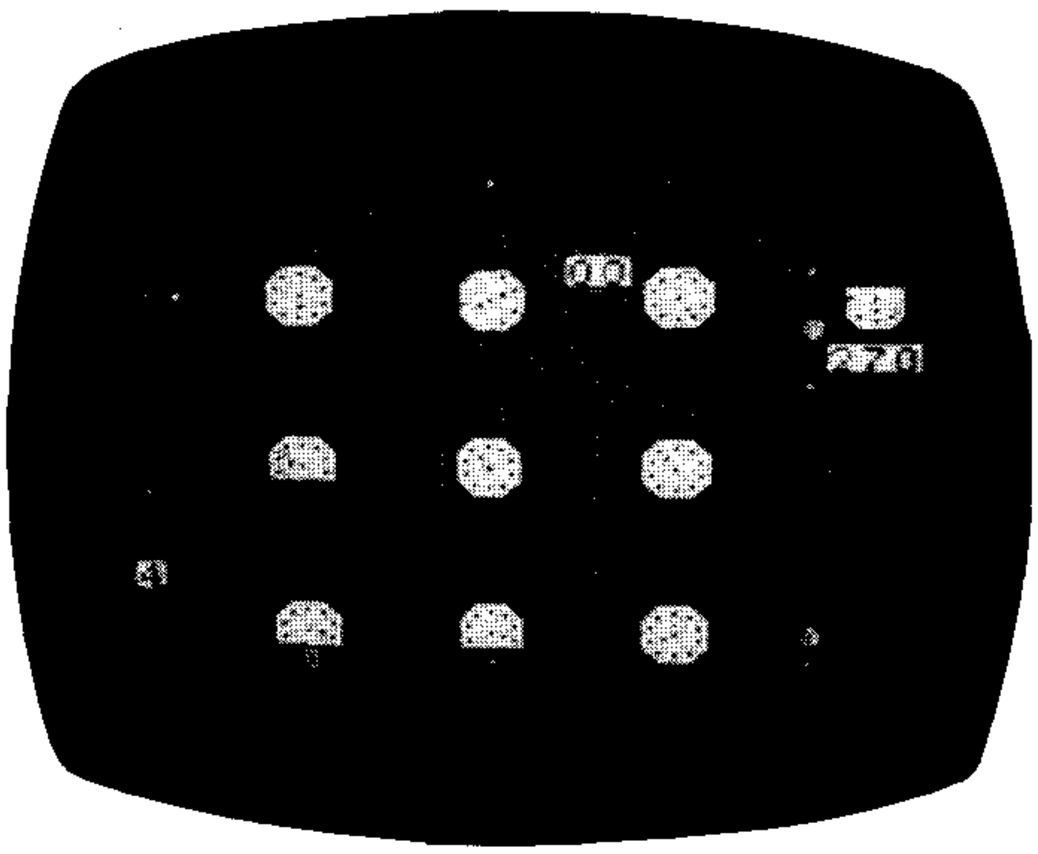


Figure 1: Simulated instrument panel of airplane

Since an aircraft panel is full of controls, one of my first tasks was to simulate a dial. Figure 1 illustrates the solution I came up with. I decided not to try to represent the needle with a line but to use, instead, mere end points. If you look at the tachometer, you can see that there are no solid lines for needles, yet when you fly, you can forget this because all the information is contained in the position of the end points. This solution allows the instruments to be three characters on a side (that is, 24 pixels), with the center character alone representing the needle position. The rounded dial face with the 10 tick marks is actually formed by the 8 fixed characters which surround the dial. A total of 40 characters is substituted into the center position, allowing increments of 9 degrees, since 40 times 9 is 360. Because there is only one character representing the needle to change, the logic of updating the display is reduced, in most cases, to a calculation of the appropriate character and one call to HCHAR. This simplification makes the program both smaller and faster. Displaying lights was a much simpler task: I used a small white circle (in a black field) for an "off" light, and a larger red one for an "on" light.

At first I used HCHAR and VCHAR to initialize the static parts of the display each time the program was run. Even though I used lists to specify the display and looped down the list, this wasted memory and was much too slow. The solution was to use PRINT statements, one for each three lines, for a total of seven statements. Nearly all the characters I used for the static parts of the display were redefined lower case letters and punctuation. For instance, a lower case "a" is redefined to be the upper left corner of a dial. Therefore, I could specify a lower case "a" in the PRINT statement but get part of a dial on the screen. Although the PRINT statements do not make sense without documentation, they are compact and define the instrument panel within a few seconds.

I made the complete display come on the screen almost instantly for several reasons. First, it seemed to me that the effect of the instruments scrolling up the screen would be distracting to the user watching the displays. Also, I had to use a few calls to HCHAR to complete the display: a couple because the characters were in the range of user-defined characters, and several to initialize various dial positions. Therefore, I prepare the whole display before turning on the colors for the various

color sets. This has the effect of turning on the complete display at once.

Sound Effects

I use sound partly to add realism but mostly to provide feedback to the user. For instance, pressing the trim key, [H], has no visual effect that is immediately apparent, but it will soon have an impact on the behavior of the plane. Also, the program does not check the keyboard frequently enough that the user can rely on a quick tap of the key; so I have the program emit a tone when it detects that the [H] key is being held down. The user can release the key at that time, or if he wants still more trim to be "cranked in," he can continue to hold the key down.

Similarly, the noise generator is used to mimic engine noise, but the frequency varies somewhat with airspeed. This provides a helpful and realistic cue to the user.

Real airplanes have an aural stall warning indicator, so I added one too. But unlike in a real airplane, where a stall is a major and easily noticed event (because the airplane starts to drop rapidly!), in the simulation one has only a visual indication on the instruments. To make the stall more noticeable, I added another, higher-pitched, tone to signal dramatically and immediately that a stall has developed.

How Fast Is BASIC?

The BASIC language is, of course, not known for its speed. I experimented to determine just how slow the various operations and subroutines were, planning to avoid or use sparingly the especially slow ones. I wrote a program like this:

```
100 INPUT N
110 FOR J=1 TO N
120 NEXT J
```

I used a watch with a sweep second hand to time the program. If you run this program, enter 1000 for N. When you hit [ENTER], start timing. It should take three seconds. This means that the time for each iteration is three milliseconds.
Now insert a statement as follows:

```
115 X=1
```

Run it again, 1000 times. It should take six seconds. Subtract the three seconds for the loop by itself, and the new statement is responsible for the remaining three seconds. In other words, a simple assignment also takes three milliseconds.

ABS	5	RANDOMIZE	1
ASC	20	READ	1
ATN	110	REM	1
CALL CHAR	120	RESTORE	1
CALL CLEAR	40	RND	9
CALL COLOR	50	SEG$	20
CALL GCHAR	50	SGN	9
CALL HCHAR	50	SIN	130
CALL JOYST	50	SQR	120
CALL KEY	40	STR$	80
CALL SCREEN	35	TAN	280
CALL SOUND	70	VAL	18
CALL VCHAR	50	+	2
CHR$	8	–	2
COS	150	*	3
EXP	170	/	6
FOR loop	3	^	50
GOSUB,RETURN	1	&	14
GOTO	1	<,>,etc	3
IF THEN ELSE	2	numeric assignment	3
INT	17	string assignment	3
LEN	6	subscription	2
LOG	200		
ON GOSUB,RETURN	3		
ON GOTO	3		
POS	40		

Note: It makes no difference in time whether you use integer values or real values as subscripts or as parameters to subroutines.

Table 1: Console BASIC times (in milliseconds)

If different statements are inserted for statement 115, the time for various operations and routines can be calculated. For instance, insert X=INT(1.5), and time it. I got 23 seconds. That is 23 milliseconds total; subtract three for the loop and another

three for the assignment, leaving 17 milliseconds for each call to INT. See Table 1 for a complete listing of times. (Note: I tried slower routines only 100 times, so those times are accurate only to the nearest 10 milliseconds.)

I tried two other variations. First, I altered the length of the variable name in the FOR loop. Each character seems to add only .1 millisecond, so don't be too concerned about the length of variable names for their effect on the speed of the program. However, frequently used long names could deprive you of hundreds of bytes of memory.

Second, I tried the loop by itself, as the first few statements in a large program, and as the last few statements in a large program. When at the front of a large program, it ran only half as fast as when alone or at the end of a large program. From this it is clear that the most frequently executed parts of a large program should be at the end.

Getting The Most Speed

As I worked on this program, I set a one-second processing cycle as my goal—that is, the controls (joystick and keyboard) and at least the most critical instruments should be checked or updated once every second. One second is 1,000 milliseconds, by definition; to a computer that should be an eternity. But look at Table 1 and see how fast that 1,000 gets used up. Updating the sound effect takes 70; reading the keyboard and joystick takes another 90; each character displayed via HCHAR takes 50. If the program changes only five characters and the parameters in a CALL SOUND statement, in addition to reading the controls, that will use up 410 of the 1000 millisecond goal. If the loop includes a SIN and a COS as well, it will use up 690 of the 1000 millisecond target, leaving only 310 milliseconds for all other processing.

I was not able to make my 1,000 millisecond goal, particularly after I added some rather elaborate options to make the program more interesting. The actual time of the processing cycle is closer to 1,500, and I had to be frugal to keep it that fast. For starters, I did not do anything more often than necessary.

For example, although the user can control the plane with any of the digits 1-9 and any of the letters A-L, only one call to KEY is necessary. The variable which is set to the key the user pressed can of course be tested any number of times.

In the same way, JOYST is called only once, even though both X and Y dimensions are used. If a program has to use one of the slower functions, such as SIN or COS, be sure that it is not called more than once with the same argument; that is, don't use SIN(A) more than once, but set TMP=SIN(A) and use TMP more than once. In general, factor out significantly time-consuming parts of formulas when they are repeated.

The user controls the plane with the letters A-L because they are represented by the numbers 65-76 consecutively. This makes it easy to test for a valid response (it must be in the range 65-76) and to branch to the proper section of code for processing, using ON GOTO. It also helps the user remember the choices: A=abort, B=back, C=climb (increase power), D=descend (decrease power), and so forth.

A problem arises in the program when values, such as the direction, "wrap around" rather than growing indefinitely larger or smaller. To simplify the computations of this type of value, I set up a vector with 120 values. It is dimensioned 0-119. The values are 120-159, repeated as a group three times. (These values are the 40 characters that were defined for the various dial positions.) A dial position character can then be selected by scaling the value, adding 40, and using the result as a subscript into the vector. For instance, the rate of turn can range from about −7.5 to about +7.5, so scale it by multiplying by 1.33; then add 40:

```
D=V(R*1.33+40)
```

If R is 0, D is V(40), or 120. This is the "straight up" pointer. If R is somewhat less than 0, D would be V(39), or 159. And if R were a little larger than 0, the character would be 121. This vector is used several times in the program to convert a real value into an integer result while scaling the value and correcting for any wrap around past the top of the dial.

In this example, the use of the vector makes this part of the program two to three times faster. The time required for this single statement is about 10 milliseconds: multiplication (3 milliseconds) + addition (2) + subscription (2) + assignment (3). Compare that to the equivalent set of statements below. Several statements are needed, instead of one, with a total time of 23 to 36 milliseconds, depending on the value of R:

```
10 D=INT(R*1.33)+120
20 IF D>=120 THEN 50
30 D=D+40
40 GOTO 70
50 IF D<=159 THEN 70
60 D=D-40
70 . . .
```

This would be somewhat faster if the INT were omitted. However, a call to HCHAR is costly (50 milliseconds), so testing D against the last value displayed can save some time. The test would not be exactly equal unless the values were exactly the same, i.e., integers. So the use of INT seems necessary. The fact that BASIC automatically converts a value to an integer for use as a subscript faster than INT does is a major reason the vector is so helpful with program speed.

```
70 IF D=LASTD THEN 100
80 LASTD=D
90 CALL HCHAR(row, column, D)
100 . . .
```

As Table 1 shows, the SIN and COS routines are slow. However, they are necessary functions for simulating an airplane's flight. They are needed to compute location in space (over the two dimensional plane of the earth's surface), and to compute the forces and motion resulting from pitch and bank. I used two techniques for faster processing.

First, I do not compute the plane's location over the surface of the earth every iteration, that is, every 1.5 seconds. Instead, I do the complete computation every four iterations, with part of the computation performed on each of the four iterations. This means that the location of the plane is updated every six seconds. In this way, the times of the SIN and COS routines are effectively divided by four.

Second, accuracy can be sacrificed for speed. Define a vector for SIN and COS and load each at the beginning of the program, either by calling SIN and COS or using READ and DATA statements. During execution, scale the angle (if necessary) by multiplying by a constant, then add a constant for a "zero adjust" and use the result as a subscript into a vector. This will take only a few milliseconds. In my program the vectors have 19 values, from 0 to 18, corresponding to −90 to +90 degrees. Thus, the following expression returns the cosine for X:

```
CSV(X+9)
```

where X is already in nine-degree increments and CSV is the vector holding the cosine values.

Memory

Before discussing memory-saving methods, let me mention a way you can tell how much memory you have remaining. In Extended BASIC you have the SIZE command, which displays the number of bytes available. Console BASIC does not have this command, but by now I am sure that many of you have heard of the following technique. Insert these two statements into your program:

```
1 QQ=QQ+8
2 GOSUB 1
```

This has the effect of calling statement 1 again and again, adding 8 to QQ on each call. (If your program already uses QQ, use some other variable name, such as ZZ.) Each call requires another 8 bytes of memory to stack the return address. Thus, this routine eventually uses up memory and stops. At that point, print the value of QQ to learn how many bytes of memory are available. Be aware that when your program executes, it can use more memory than needed just to store the program. Place the statements in the program where they will be executed during the heaviest use of memory.

Continued on p. 58

Jailhouse Romp

A Review of Jail Break

by Judy Sanoian
99'er HCM Staff

Name:	**Jail Break**
Author:	**Mehdi Habibi**
Program Type:	**Cops and Robbers**
Language:	**Assembly Language**
Distributor:	**Bit Byte Bit Software** **P.O Box 565** **Coram, NY 11727**
Price:	**$19.00**

System Requirements:
Joysticks
Mini Memory with 32K expansion
Line-by-Line Assembler
Cassette recorder and cable

	Poor	Fair	Good	Excellent
Performance				
Engrossment				
Documentation				

The alarm sounds, and the searchlights flood the prison yard. It is every warden's nightmare. Riot in cellblock number nine, breakouts in ten, eleven and twelve. The ex-cons have stormed the barricades, and the inmates are swarming through the gates, heading for town to plunder and pillage. That's the scene in *Jail Break*, a fast action game by Bit Byte Bit Software.

As in most computer games, you represent the arm of the law. As the lone sheriff in a town overrun by escaped criminals, your goal is to capture the deranged escapees. You must work fast, for the console cons' sole purpose in life is to sack the local bank. And you cannot rest until *all* the thugs are safely behind bars. If there are any crooks at large, they will help their cohorts break out of the pen.

If you are accustomed to games in BASIC or Extended BASIC, your first reaction to *Jail Break* will be "Wow, nice response!" I felt like I had come down out of my Sherman tank and taken command of a Rolls Royce. Such ease in handling—no more jamming on the joystick while a sluggish character chugged right or left. And the speed! The escaped cons charge about frenetically, smashing into one another like frenzied molecules over a bunsen burner. Your lawman is equally fast on his feet, tearing in and out of banks and stopping on the proverbial dime. Unfortunately there's a price to pay for all this precision and speed. A small investment in equipment (a Mini Memory cartridge plus 32K of expansion memory) and a

computer gaming™

Computer Gaming is a section for all game lovers—players, designers, and programmers of microcomputer games. Regular features include product reviews, letters to the editor, player strategy, a question and answer forum, a Hall of Fame for high scorers, tutorial articles on game design and programming, plus interviews with professionals in the world of computer gaming.

All submissions for *Pros on Programming* are governed by the same conditions and payment rate as manuscripts sent to other departments of *99'er Home Computer Magazine*. Materials submitted for the features shown below are treated the same for Copyright purposes as *Letters to the Editor* in *99'er Home Computer Magazine* (as explained in the Masthead); if chosen for publication, the material (except for *99'er Hall of Fame*) will earn for its author a free computer game (either TI or third-party) and/or a one-year subscription to this magazine.

99'er Hall of Fame candidates with high scores in TI, third-party, or *Computer Gaming* games must completely describe the conditions under which their scores were achieved (i.e., skill level, keyboard or joystick use, screen number, partner participation, appearance of screen, etc.) Candidates may not be directly related to or affiliated with the programmer of the game or the publishing firm. No compensation will be provided to new inductees whose names are chosen to be immortalized—Fame is its own reward.

Game Review Criteria

Game Performance measures how well the game responds to the player's commands, rates the quality and realism of the graphics and animation, and examines how well the sound effects, music or speech are integrated into the game. It also determines whether the game delivers what is promised in its advertisements.

Engrossment focuses on that intangible quality that holds the player on the edge of his seat while the hours tick by unnoticed. The game's *staying power* is also assessed.

Documentation rates the printed matter that comes with the game. It notes whether the instructions are clear, comprehensive and easy to use, whether the machine configuration requirements are spelled out, and looks for such information as how to load the program, use the keyboard, and restart the game.

laborious loading process are necessary to play *Jail Break*.

The game begins at the moment of the jail break with the thugs spilling from the prison gates and speeding helter skelter towards the banks. Your fearless lawman is immediately on their tail. As he makes contact with each escapee, they are instantly transported back to the slammer. The thieves at large are meanwhile busily invading the four local banks, smashing into walls as if drunk with their newly won freedom. When they have stolen all the cash in the banks, your playing time (and the local economy) is wiped out. The goal is to get all the escapees back into their cells at once. . . before all the money is gone.

In planning your strategy, keep in mind an old bit of wisdom from the Depression—never stash all your cash in one bank. If you do, you will find yourself hovering around that one bank entrance, guarding your only remaining money like a nervous Scrooge McDuck. Meanwhile the thieves will have their run of the town. With these hoodlums, your best defense is an early offense. The second the game starts, you should race around, stopping the thugs at all four banks before they get the chance to deplete any of them. For as the game wears on, these two-bit crooks become hardened felons, converging on the banks that have cash like hungry cockroaches. Also, later in the game, you will find that the walls of the banks have crumbled under the constant state of siege.

Laid Back Lawman

If you are disturbed by violent computer game scenarios, you will like the gentle tone of *Jail Break*. Instead of zapping, shooting, or otherwise obliterating the escapees, our harried lawman merely directs them back into the penal system. On the other hand, those who decry the revolving prison door syndrome will find *Jail Break* all too lifelike. No sooner are these thugs slapped into the slammer than they are back out on the street, knocking over the first bank in sight. Another realistic touch is the varied response of the prisoners. Some zip back to prison at the touch of your billy club; others tend to resist arrest. You may have to "bump" these hardcore types several times.

In addition to superior speed and response, *Jail Break* boasts some fine extra features . You can instantly freeze the action by pressing the [ENTER] key. This is especially helpful at the start of the game when the escapees are swarming all over the screen faster than the eye can follow. If you immediately press [ENTER], you can scan the screen to see where you should go to make your first arrests. There is also an effective utilization of sound in this game. There is a pleasant "plink" when you capture a prisoner, and a low-pitched "doink" tells you when the thugs have entered a bank or when a bank's walls are tumbling down. I appreciated this audible -rather than visual- -warning because my eyes were constantly glued to my fast-moving sheriff.

Another fancy extra is *Jail Break's* automatic permanent scoreboard. Your Mini Memory lets you save your scores so you can try to better them the next time you play. A word of warning, however, about this scoreboard: You will need to have your [ALPHA LOCK] key down to enter your name for the score sheet. Immediately after you type in your name, the game will start. If your [ALPHA LOCK] is still down, you will not be able to move. You can lose precious seconds trying to figure out why your lawman is suddenly slow on his feet. The documentation warns you about this, but it is easy to forget in the heat of a prison riot.

Undocumented Crimes

The documentation is the only sour note in *Jail Break*. The directions for loading the program are simply not adequate. The author presumes that if you own a Mini Memory, you already know how to load the *Line-by-Line Assembler*. This may be a fair assumption, but it would still be convenient to have the complete instructions for loading the game in one place. As it is, you must look up how to load the *Line-by-Line Assembler* in the Mini Memory manual, then switch back and find your place in the *Jail Break* loading instructions. Also, the procedure to enter the program name into Mini Memory is very difficult to follow. At one point, there is no direction to press the [ENTER] key when it is required, and you are left to guess at the number of spaces between characters. There is no slash through the 0 character to tell you it is zero rather than a capital O. Loading *Jail Break* for the first time is already a lengthy process, without the frustration of incomplete instructions. It was also not clear to me how much of the loading procedure had to be performed every time you load the game. You must repeat all the procedures —except loading the *Line-by-Line Assembler*— each time you want to play. Of course, you will figure all of this out after you have loaded the program once or twice, but it is quite confusing the first time through.

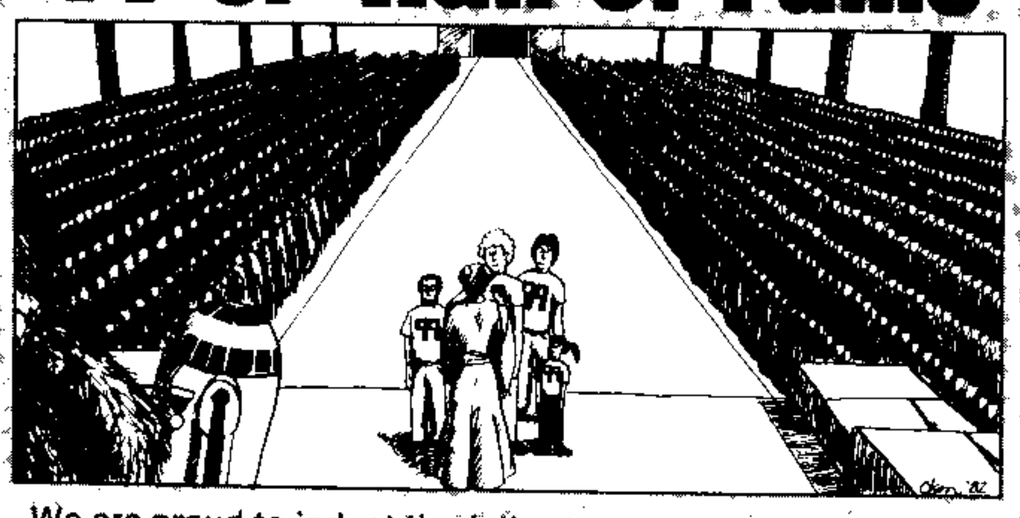
Jail Break features clean, crisp graphics. The prisoners are represented by hyperactive dots that bounce and reverberate against each other in a manner similar to Brownian motion. In sharp contrast, your lawman is a steadfast, stolid stick figure with two feet planted squarely on the ground. A black box with a perpetual open door policy represents the jail, and the banks are square structures with disintegrating walls and easily accessible dollar signs. At specified points in the game the prisoners change color, making for some pleasing visual variety.

Nine levels of difficulty are offered in *Jail Break*. You are also asked to select whether you want to play with eight, sixteen, or twenty-four fugitives on the lam. There is an exceptionally wide range of difficulty among these many levels. Most novices will be able to win on levels zero and one—a nice psychological lift for some of us who are used to bombing on the first round of any joystick-operated game. Once past the first couple of rounds,

however, the game quickly becomes a challenge. The difficulty level automatically adjusts itself when you win a game (trap all the prisoners).

I can guarantee that *Jail Break* is tough enough to challenge even the most tenacious law and order types. With a constant supply of rambunctious thugs breaking, entering, and looting at breakneck speed, you are more likely to burn out from exhaustion than from boredom. This is an edge-of-your-seat type of game—definitely not the thing for unwinding after a hard day at the office. If you've got the guts, determination, and a whole posse of console add-ons, I recommend you take a crack at *Jail Break*. Fine response and killer speed make it a joy to stick it to these cons. And at $25 for each thug you pop, this is a game in which crime can pay quite handsomely.

99'er Hall of Fame



We are proud to induct the following gamesters into the 99'er Hall of Fame: We will be sending each of them one of our new Hall Of Fame Certificates. Congratulations to

Name: **Brian Benton (Sacramento, CA)**
Game: **TI Invaders**
Score: **34,360 (Level 2)**

Name: **Mark Houslet (Oxford, WI)**
Game: **Munch Man**
Score: **386,280 (118th Screen)**

Name: **Ron L. Walters (Brandenburg, KY)**
Game: **Car Wars**
Score: **121,000 (13th Level)**

Name: **Sandy Bohnet (Kalamazoo, MI)**
Game: **Tombstone City**
Score: **366,000 (Level 2)**

Name: **Chris Jordan (Marshall, MI)**
Game: **Jungle Jim**
Score: **2,500 (Level 1)**

Name: **Mark E. Andersen (West Allis, WI)**
Game: **Jungle Jim**
Score: **2,500 (Level 1)**

Send in Your Photos and Anecdotes

Do you have a favorite photograph (color or black and white) featuring an unusual application of your Home Computer? Would you like to share your unusual or amusing anecdotes relevant to Home Computing? *99'er Home Computer Magazine* will pay $25 for items it publishes. Material chosen will be subject to the same copyright treatment as "Letters to the Editor" as set forth on the Masthead page. No submissions can be returned. Send anecdotes and copies of photos to: Potpourri Editor, 99'er Home Computer Magazine, 1500 Valley River Drive, Suite 250, Eugene, Oregon 97401.

Arithmetex

A Review by Erin O'Connor

99'er HCM Staff

Name: Arithmetex
Program Type: Educational Math game
Language: TI BASIC
Distributor: Fantasia '99 Software, 3 Victor Boulevard, St. Catherines, Ontario, Canada L2T 2B2
Price: $11.95 US/$14.95 CAN

System Requirements:
Cassette recorder and cable

	Poor	Fair	Good	Excellent
Performance				
Engrossment				
Documentation				

After an unexceptional title screen, some routine selections of menu options (Addition, Subtraction, Multiplication, Division, or Scrambled), level of difficulty (low or high), number of players (1 to 5), and player(s) name(s), it was gratifying to get to the heart of *Arithmetex*. The display opened to a well-drawn thermometer in which red mercury would rise in 5- to 10-point increments as my score rose to 100 and end-of-game. The arithmetic problems themselves featured red numbers and navy blue operands in horizontal sequence against a lavender background. It was fun, at first, to watch the numerals and operands of a problem be created in segments right before my very eyes. And I liked the personal way the machine addressed me, even when it was to say, tactfully, SORRY, ERIN. THE CORRECT ANSWER IS —.

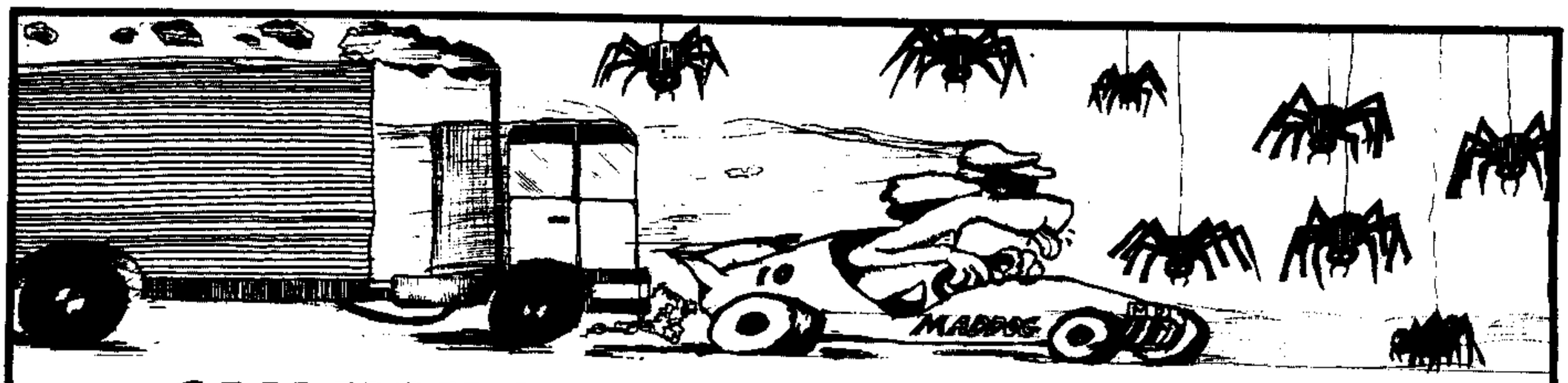
But by the time the program and I had reached the point of CONGRATULATIONS, ERIN. YOU WIN WITH A SCORE OF 105, the honeymoon was over. The trophies arrayed (looking like so many genie lamps) in a row across the screen were unremarkable, and the program's thoughtfulness in noting my average score (speed from 5 to 10) was carried out to a pedantic nine decimal places. Relations were already strained when I realized early on that despite the seeming courtesy of the program's initial questions, it just wasn't going to listen to me. Before each problem the program would considerately inquire READY? If I responded [Y], after an interval the program would write out a problem. If I responded [N], after the same interval, it would write out a problem. I began to feel that the program's friendliness was just a false heartiness. I was not supposed to take the question literally, but as a warning that the next problem would be screened sometime soon.

How Fast Can You Answer?

The time between problems began to seem interminable. The program's three unaltering responses to correct answers, incorrect answers, and winning the game, interspersed with the superfluous READY? made for slow going. When I wearied of the snail's pace of one option, I turned in vain to the documentation for instructions on how to get back to the menu. I had to patiently amass 100 points or resort to BREAK [FCTN][4].

Unlike Oscar Wilde, I might have forgiven the program the sin of tediousness, but its unfairness was the last straw. The game advances the questionable idea (more on that later) that there is merit in

Continued on p. 43



CAN YOU MEET THE CHALLENGE OF:

MAD DOG I:
You're speeding down the highway of the Twilight Zone pursued by a big, Mack truck, when suddenly you encounter giant, mutant spiders. Your mission: spider mash. Beat the timer and crush bonus bugs for extra points. Advancing levels of play.

WYVERN:
('wī vərn), a two-legged dragon with wings and a barbed tail. Can you get the treasures without being fried? Look out for the web-spinning spider, the baby dragons, and especially the fire breathing Wyverns. Multiple rooms, advancing levels of play.

CAVEMAN:
A long time ago on a planet far away. . . Dodge giant sow bugs, hit beasts with thrown rocks and evade the vampire bats. Choose from Level 1 to 7, advancing difficulty. Great intro. song with explosion.

MAD DOG II: (Assembly language)
A souped up version of MAD DOG I. Choose from Level 1 to 4. Quicker response. Extra car. More fun. Totally awesome crash!

LOST PATROL:
Stranded deep in enemy territory, you must make your way back to your forces. Your 5 man patrol must avoid enemy men, treacherous terrain, and forces of nature. Pray that at least 1 man of your patrol can survive. 4 boards, choose from Level 1 to 7. JUNGLE KING, watch out!

BASEBALL TRIVIA:
Over 159 questions of hitters and pitchers. Such as: "Who struck out 21 men in 1 game?" Multiple choice. Each game is different. Intro. song is "Take Me Out to the Ball Game". Not to be missed! Hours of fun.

MasterCard

CALL TOLL FREE

VISA

Mastercard & VISA Card Holders

DATA/WARE DEVELOPMENT, INC.

1-800-382-3282 **1-800-882-3282** (CA only)

All games feature professional arcade-like color graphics and sound effects.

All orders shipped within 2 business days!
10% discount (2 or more games)

1-619-453-7660 (AK, HI, Canada)

Dealers' and Programmers' inquiries invited.

FREE catalog upon written request.

ORDER FORM

Extended BASIC Module Required

QTY.		QTY.	
☐	Wyvern (cassette) $14.95	☐	Wyvern (disk) $16.95
☐	Mad Dog I (cassette) $13.95	☐	Mad Dog I (disk) $15.95
☐	Lost Patrol (cassette) $13.95	☐	Lost Patrol (disk) $15.95
☐	Caveman (cassette) $13.95	☐	Caveman (disk) $15.95
☐	Baseball Trivia (cassette) $8.95	☐	Baseball Trivia (disk) $10.95

Mini Memory Module Required

☐	Mad Dog II (cassette) $15.95	☐	Mad Dog II (disk) $17.95

☐ Check ☐ Money Order ☐ MasterCard ☐ VISA

☐ Card No. _____ Exp. _____

Name _____ Signature _____

Street _____

City/State/Zip _____ Phone _____

Send this form or a copy with your check, money order or credit card information to:

Data/Ware Development, Inc.
4204 Sorrento Valley Blvd.
San Diego, CA 92121

ATTN: Games

SUBTOTAL: _____

10% discount (2 or more games): ( ) _____

CA res. sales tax: _____

Shipping & Handling: $2.25 _____

TOTAL ENCLOSED: _____

Taco Man

by Robert S. Kemmetmueller
19420 133rd Ave. No.
Rogers, MN 55374

tle your tortilla down the halls, remember: This race is not to the swift, but to the *gordo*.

We've all heard of *MunchMan* and games of the *Burger Builder* variety. There have even been rumors of a *Nacho Man* game, featuring a machismo tortilla guzzler. Well, now get ready to play the latest addition to the fast-food chowhound game collection—*Taco Man*! In this game you get five tries (taco shells) to gobble all the taco ingredients before the ravenous tacovores devour you.

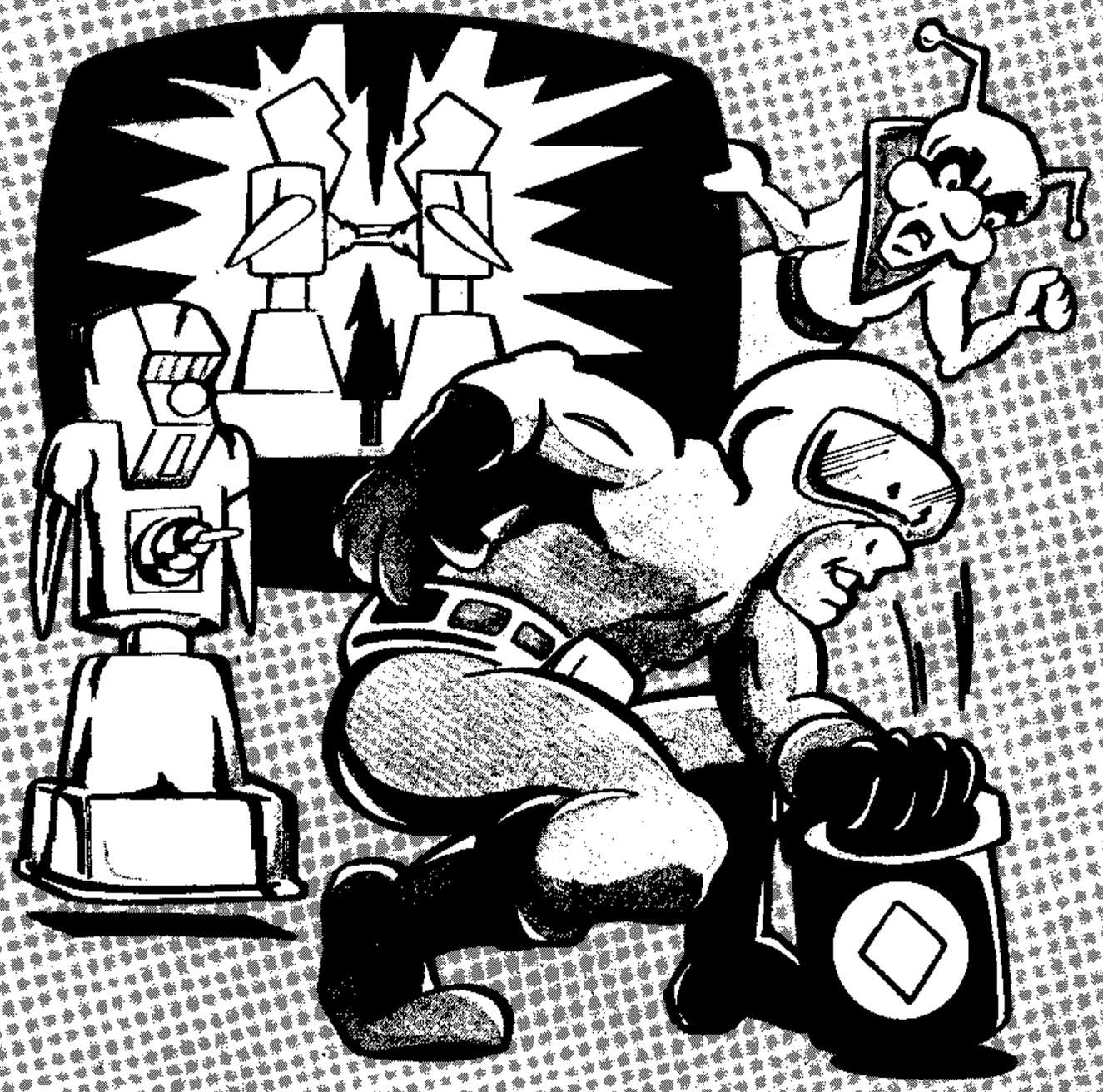
You move your taco man using either the E, X, S and D keys or the joystick. Wandering through the corridors of the Taco Heaven kitchen, your tireless tortilla gobbles up huge quantities of hamburger, cheese and tomato slices. You rack up points for every tidbit you eat, as long as you avoid the jaws of the pumpkinesque tacovores. An encounter with these fiends will put an end to your feeding frenzy. Your taco is not, however, totally defenseless. These tacovores are real *gringos* when it comes to jalapenos. One bite of a chili pepper and you can clear them out of the room. You are then free to chow down to your heart's content until their return (signalled by a flashing screen). As you hus-

EXTENDED BASIC

Taco Man
Explanation of the Program

Line Nos.	
100-140	Progam Header.
150-180	Check for memory expansion.
190-250	Initialize graphics characters.
260-280	Initialize colors.
290-510	Data for playing screen format.
520-560	Display playing screen.
570-640	Initialize new game and sprites.
650-720	Change sprite colors, beginning of control loop.
730	Check for a collision.
740-790	Scan joystick and keyboard.
800-880	Move taco man, and update score.
890-920	Display score. Check to see if all of the dots have been eaten.
930-1030	Taco man gets killed.
1040-1100	Advance to the next screen.
1110-1200	End of the game.
1210-1600	Display instructions.

Continued on p. 39



Robochase

by Greg Vaughan
211 Frederick St.
Santa Cruz, CA 95062

While on a routine patrol in sector 47, your spaceship was attacked, and you have been taken prisoner by a mad scientist. He has taken you to his asteroid and sentenced you to death for intruding upon his territory.

You method of execution is a bizarre one. You are thrown into a room full of deadly robots and electrified barriers. As the robots begin to move straight towards you, you realize that your only hope is to maneuver the robots into the electrified barriers and into each other. If you are surrounded, you can use your teleportation belt to escape, but you should use it only as a last resort because it has only one charge left on it. The diamond-shaped recharger packs located on the ground are your only hope, so you desperately head for them. Finally you destroy all of the robots in the room, and have even accumulated several extra teleport charges. But before you can congratulate yourself, you are teleported into another room with another ten robots. This time there are fewer barriers and teleport rechargers. To make things worse, half the charges in your teleport belt have been drained!

The Game

Robochase is a fight to the death in a prison somewhere in the future. You can move up, down, left, or right by means of either the Q, A, P, and / (slash) keys or the joystick. Diagonal movement is not allowed. You can teleport by using either the space bar or the fire button on the joystick.

Each level opens with ten robots. The robots move directly towards you, either horizontally, vertically, or diagonally. (The chance that they will move diagonally increases as you go to higher levels). If the robots run into barriers, they are destroyed. If they run into each other on the first and second levels, both robots are destroyed and a junk pile is formed. The junk pile acts as a new barrier. On the third and fourth levels, colliding robots are destroyed, but no helpful junk pile is left behind. On the fifth level and beyond, only one robot is destroyed. The other remains to menace you.

In the upper right-hand corner of the screen, the tally of your teleports is displayed. You start out the game with one. To gain additional teleports, you run across the white diamonds on the screen. Every time you go up a level, the number of teleports you have left is divided by two (rounded down).

There are four ways you can die: You can be captured by a robot, collide with a barrier, run into a junk pile, or bash into one of the outside walls. If all the robots on a level die, the screen will flash, your score will be displayed, and you will proceed to the next level.

Starting on the fourth level, Spunky the Martian will appear. Spunky moves in a billiard ball pattern, bouncing off each of the four walls. He will destroy barriers, robots and teleport rechargers. If you eliminate Spunky, you get a bonus teleport.

Scoring for the game is as follows: You get 25 points for each robot killed, 75 points for each diamond collected, 100 points for getting Spunky and 125 points for going up a level. You lose 50 points each time you teleport.

Note: When typing the program in, if you don't want to bother having instructions in the program, type in REM in line 920 and delete lines 3340-3720. Also remove the second part (" < OR ! FOR INSTRUCTIONS > ") of the PRINT statement in line 540.

BASIC

Robochase
Explanation of the Program

Line Nos.	
100-460	Initialize variables for title.
470-680	Display title screen.
690-920	Initialize variables for game.
930-1290	Set up game board.
1300-1410	Get input and do joystick movement.
1420-1530	Do keyboard movement.
1540-1620	Check for person running into something.
1630-1820	Move robots.
1830-1950	Check for robots running into things.
1960-2080	Display death messages.
2090-2350	Routine for going up a level.
2360-2470	Routine for teleporting.
2480-2520	Routine for getting a diamond.
2530-2640	Subroutine for flashing person.
2650-2770	Display score and ask to play again.
2780-2900	Death subroutine.
2910-3030	Check for robot collision and create junkpiles.
3040-3170	Move Spunky and check for collision.
3180-3260	Routine for capturing Spunky.
3270-3330	Routine for Spunky capturing a robot.
3340-3720	Instructions.

Continued on p. 35

Make Someone Really Happy on Christmas Morning

As the holiday season approaches, you're sure to be wondering what to get for those 99/4 and 99/4A owners on your shopping list. If you know (or are) a 99'er who appreciates high-quality software packages, then **REPUBLIC SOFTWARE** has the solution to your gift-giving problems.

All **REPUBLIC SOFTWARE** products have the unique ability to adapt themselves to your computer system's capabilities as your system grows. Each package will execute on a cassette-based console BASIC system (RING DESTROYER requires Extended BASIC), but each program also includes versions or options to take advantage of the greater speed or capability of such additional equipment as Joysticks, Assembly Language, Extended BASIC, disk drives, the Expansion RAM, the speech synthesizer, or the RS232 interface and compatible printer. For trouble-free loading, **REPUBLIC SOFTWARE** programs are recorded on high-quality cassettes or disks produced by such manufacturers as TDK, Scotch, and Wabash. And each **REPUBLIC SOFTWARE** program is accompanied by a clearly-written, detailed instruction manual printed (not photocopied) on materials that will last.

So, for the best in software value, see your dealer or order directly from **REPUBLIC**. RING DESTROYER, UTILITIES 1, and INTERCEPTOR are each available for $19.95. Please specify disk or cassette.

Order Now for Christmas!

Phone (202) 978-3554 to place your order and we will reduce the price by $2.00 to pay for the call. VISA and MasterCard welcome. To better serve our customers in all time zones, our hours have been changed to 1 to 8 PM EST, Monday through Friday.

Dealers—ask about our special Holiday Pak.

PO Box 23042 • L'Enfant Plaza
Washington, DC 20024

REPUBLIC SOFTWARE

MasterCard VISA

Robochase... from p. 33

```
100 REM  *************
110 REM  * ROBOCHASE *
120 REM  *************
130 REM  BY GREG VAUGHAN
140 REM  99'ER VERSION 2.13.1
150 REM  DISPLAY TITLE SCREEN
160 RANDOMIZE
170 CALL SCREEN(16)
180 CALL CLEAR
190 DIM Z$(4)
200 DATA FFFFFFFFFFFFFFFF,01030
70F1F3F7FFF,FF7F3F1F0F07030
1,80C0E0F0F8FCFEFF,FFFEFCF8
F0E0C080
210 DATA FCFCFCFCFCFCFCFC,3F3F3
F3F3F3F3F3F
220 DATA E0F0F8F0E0F0F8,F8F8D8D
8D8F8F8,F0D8D8F0D8D8F0,78F0
E0E0E0F078,8888D8F8D88888
230 DATA 2070D8D8F8D888,70D8C07
018D870,F8F0E0F8E0F0F8
240 DATA FCFCCCCCCCCCFCFC,38381
818181818,FCFC0CFCFCC0FCF
C,FCFC0CFCFC0CFCFC,CCCCCCFC
FC0C0C0C
250 DATA FCFCC0FCFC0CFCFC,FCFCC
0FCFCCCFCFC,FCFC0C0C0C0C0C0
C,FCFCCCFCFCCCFCFC,FCFCCCFC
FC0C0C0C
260 DATA 8040,2010,0804,0201
270 FOR X=33 TO 39
280 READ A$
290 CALL CHAR(X,A$)
300 NEXT X
310 FOR X=104 TO 111
320 READ A$
330 CALL CHAR(X,A$)
340 NEXT X
350 FOR X=48 TO 57
360 READ A$
370 CALL CHAR(X,A$)
380 NEXT X
390 READ A$,B$,C$,D$
400 Z$(1)=A$&B$&C$&D$
410 Z$(2)=D$&A$&B$&C$
420 Z$(3)=C$&D$&A$&B$
430 Z$(4)=B$&C$&D$&A$
440 CALL CHAR(97,Z$(1))
450 CALL CHAR(98,Z$(4))
460 CALL COLOR(9,9,16)
470 PRINT "aaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaa"
480 A$=CHR$(34)
490 PRINT "a!$  !!  !$  !!  !%  &'  "
;A$;"$  ";A$;"$  !%b"
500 PRINT "a!%  &'  !%  &'  !  !!  &
'  !$  !$b"
510 PRINT "a!$  &'  !$  &'  !  !!  !
!  #!  !%b"
520 PRINT "a&'  !!  !%  !!  !$  &'  %
#  #%  !$b"
530 PRINT "bbbbbbbbbbbbbbbbbbbb
bbbbbbbbbbbbbbbbbbbbbbbbbbbb
bbbbbbbbbb"
540 PRINT ::::"<PRESS ANY KEY T
O CONTINUE>":"  <OR I FOR I
NSTRUCTIONS>":::::
550 X=3
560 Y=1
570 CALL COLOR(1,X,1)
580 CALL CHAR(97,Z$(Y))
590 CALL CHAR(98,Z$(5-Y))
600 CALL KEY(3,K,S)
610 IF S<>0 THEN 700
620 X=X+1
630 Y=Y+1
640 IF Y<5 THEN 660
650 Y=1
660 IF X<13 THEN 570
670 X=3
680 GOTO 570
690 REM  INITIALIZE VARIABLES
700 R=10
710 D=1
720 S=0
730 LE=1
740 BA=24
750 DI=6
760 CALL CLEAR
770 CALL CHAR(126,"000008181C3C
3E7E")
780 CALL CHAR(119,"81423C247E24
3CFF")
790 CALL CHAR(128,"00107C383828
2800")
800 CALL CHAR(136,"38107C7C7C6C
6C6C")
810 CALL CHAR(144,"007E7E7E7E7E
7E00")
820 CALL CHAR(152,"10387CFE7C38
1")
830 CALL SCREEN(15)
840 CALL COLOR(1,2,1)
850 CALL COLOR(13,3,16)
860 CALL COLOR(14,5,15)
870 CALL COLOR(15,7,16)
880 CALL COLOR(16,16,15)
890 CALL COLOR(11,3,1)
900 DIM A(10)
910 DIM B(10)
920 IF K=73 THEN 3350
930 PRINT "          hijklmno"
940 PRINT "    !!!!!!!!!!!!!!!!!!!!
!!!!"
950 FOR X=1 TO 20
960 PRINT "    !
    !"
970 NEXT X
980 PRINT "    !!!!!!!!!!!!!!!!!!!!
!!!!"
990 FOR X=1 TO BA
1000 CALL HCHAR(INT(RND*20+3),IN
T(RND*20+6),144)
1010 NEXT X
1020 FOR X=1 TO DI
1030 CALL HCHAR(INT(RND*20+3),IN
T(RND*20+6),152)
1040 NEXT X
1050 FOR X=1 TO 10
1060 A(X)=INT(RND*20+6)
1070 B(X)=INT(RND*20+3)
1080 CALL GCHAR(B(X),A(X),CH)
1090 IF CH<>32 THEN 1060
1100 CALL HCHAR(B(X),A(X),136)
1110 NEXT X
```

Robochase

```
110 BU=0
120 IF LE<4 THEN 1220
130 BU=1
140 NX=1
150 NY=1
160 BX=INT(RND*19+6)
170 BY=INT(RND*19+3)
180 CALL GCHAR(BY,BX,CH)
190 IF CH<>32 THEN 1170
200 CALL HCHAR(BY,BX,119)
210 Q=INT(RND*20+6)
220 W=INT(RND*20+3)
230 CALL GCHAR(W,Q,CH)
240 IF CH<>32 THEN 1220
250 RT=0
260 CALL HCHAR(1,29,48+D)
270 GOSUB 2530
280 CALL HCHAR(W,Q,32)
290 REM MOVE PERSON
300 CALL KEY(3,I,T)
310 CALL KEY(1,X,Y)
320 IF (X<>18)+(I=81) THEN 1360
330 I=32
340 GOTO 1420
350 CALL JOYST(1,X,Y)
360 IF (X=0)*(Y=0) THEN 1420
370 IF X=4 THEN 1440
380 IF X=-4 THEN 1470
390 IF Y=4 THEN 1500
400 IF Y=-4 THEN 1530
410 IF (I=32)*(D>0) THEN 2370
420 IF I<>47 THEN 1460
430 Q=Q+1
440 GOTO 1540
450 IF I<>80 THEN 1490
460 Q=Q-1
470 GOTO 1540
480 IF I<>81 THEN 1520
490 W=W-1
500 GOTO 1540
510 IF I<>65 THEN 1540
520 W=W+1
530 CALL GCHAR(W,Q,CH)
540 IF CH=136 THEN 2010
550 IF CH=144 THEN 2050
560 IF CH=152 THEN 2480
570 IF CH=33 THEN 2050
580 IF CH=126 THEN 1970
590 IF CH=119 THEN 3180
600 CALL HCHAR(W,Q,128)
610 CALL SOUND(50,300,0,600,0,1
    200,0)
620 REM MOVE ROBOTS
630 IF BU=1 THEN 3040
640 FOR X=1 TO 10
650 IF A(X)=0 THEN 1940
660 CALL HCHAR(B(X),A(X),32)
670 CH=.24+(LE*.03)
680 IF CH<=.43 THEN 1710
690 CH=.43
700 IF RND>.5+(CH/2) THEN 1780
710 IF A(X)=Q THEN 1780
720 IF A(X)<Q THEN 1760
730 A(X)=A(X)-1
740 GOTO 1770
750 A(X)=A(X)+1
760 IF (RND>CH/(.5+(CH/2)))+(B(
    X)=W) THEN 1830
770 IF B(X)=W THEN 1730
780 IF B(X)<W THEN 1820
790 B(X)=B(X)-1
800 GOTO 1830
810 B(X)=B(X)+1
820 CALL GCHAR(B(X),A(X),CH)
830 IF CH=128 THEN 2010
840 IF (CH=136)*(LE<5) THEN 2920
850 IF CH=32 THEN 1930
860 A(X)=0
870 S=S+25
880 CALL SOUND(50,-5,0)
890 R=R-1
900 IF R=0 THEN 2100
910 GOTO 1940
920 CALL HCHAR(B(X),A(X),136)
930 NEXT X
940 GOTO 1290
950 REM DEATH MESSAGES
960 GOSUB 2780
970 CALL CLEAR
980 PRINT "YOU RAN INTO A JUNK
    PILE":"ON LEVEL";LE::
990 GOTO 2660
1000 GOSUB 2780
1010 CALL CLEAR
1020 PRINT "YOU HAVE BEEN CAPTUR
    ED BY":"A ROBOT ON LEVEL";L
    E::
1030 GOTO 2660
1040 GOSUB 2780
1050 CALL CLEAR
1060 PRINT "YOU HAVE BEEN ELECTR
    IFIED":"ON LEVEL";LE::
1070 GOTO 2660
1080 REM GOING UP A LEVEL
1090 CALL COLOR(15,7,15)
1100 S=S+125
1110 FOR X=1 TO 5
1120 CALL SCREEN(7)
1130 CALL SOUND(-1000,110,0,220,
     0,400,0,-7,0)
1140 CALL SCREEN(15)
1150 CALL SOUND(-1000,300,0,600,
     0,800,0,-7,0)
1160 NEXT X
1170 CALL SOUND(-1,400000,30)
1180 R=10
1190 LE=LE+1
1200 IF BA<=12 THEN 2230
1210 BA=BA-4
1220 IF DI<=3 THEN 2250
1230 DI=DI-1
1240 D=INT(D/2)
1250 CALL HCHAR(1,29,48+D)
1260 GOSUB 2530
1270 CALL CLEAR
1280 CALL COLOR(15,7,16)
1290 PRINT "ENTERING LEVEL";LE::
1300 PRINT "CURRENT SCORE:";S::
1310 FOR II=1 TO 1000
1320 NEXT II
1330 CALL CLEAR
1340 GOTO 930
1350 REM TELEPORTING
```

Robochase

```
2370 GOSUB 2530
2380 D=D-1
2390 S=S-50
2400 CALL HCHAR(1,29,48+D)
2410 CALL HCHAR(W,Q,32)
2420 Q=INT(RND*20+6)
2430 W=INT(RND*20+3)
2440 CALL GCHAR(W,Q,CH)
2450 IF CH<>32 THEN 2420
2460 GOSUB 2530
2470 GOTO 1290
2480 D=D+1
2490 CALL HCHAR(1,29,48+D)
2500 S=S+75
2510 CALL SOUND(50,-3,0,700,0)
2520 GOTO 1610
2530 CALL HCHAR(W,Q,128)
2540 FOR X=1 TO 5
2550 FOR Y=1 TO INT(RND*20+2)
2560 NEXT Y
2570 CALL COLOR(13,16,16)
2580 CALL SOUND(10,500,0)
2590 FOR Y=1 TO INT(RND*20+2)
2600 NEXT Y
2610 CALL COLOR(13,6,16)
2620 CALL SOUND(10,200,0)
2630 NEXT X
2640 RETURN
2650 REM END OF GAME
2660 PRINT "YOUR SCORE IS";S::
2670 PRINT "PLAY AGAIN (Y/N)?"
2680 T=0
2690 CALL KEY(3,K,S)
2700 IF K=89 THEN 700
2710 IF K=78 THEN 2770
2720 T=T+1
2730 IF T<250 THEN 2690
2740 CALL CLEAR
2750 CALL SCREEN(16)
2760 GOTO 440
2770 END
2780 CALL HCHAR(W,Q,128)
2790 FOR X=1 TO 5
2800 FOR Y=1 TO 10
2810 NEXT Y
2820 CALL COLOR(13,16,7)
2830 CALL SOUND(-1000,110,0,120,
     0,130,0,-6,0)
2840 FOR Y=1 TO 20
2850 NEXT Y
2860 CALL COLOR(13,6,16)
2870 CALL SOUND(-1000,220,0,240,
     0,260,0,-7,0)
2880 NEXT X
2890 CALL SOUND(-8,40000,0)
2900 RETURN
2910 REM ROBOT ROUTINES
2920 FOR Y=1 TO 10
2930 IF (A(X)=A(Y))*(B(X)=B(Y))*
     (X<>Y) THEN 2950
2940 NEXT Y
2950 S=S+50
2960 R=R-2
2970 A(Y)=0
2980 CALL HCHAR(B(X),A(X),32)
2990 IF LE>2 THEN 3010
3000 CALL HCHAR(B(X),A(X),126)
3010 A(X)=0
3020 CALL SOUND(300,300,0,-3,0)
3030 GOTO 1910
3040 CALL HCHAR(BY,BX,32)
3050 BX=BX+NX
3060 IF (BX<>25)*(BX<>6) THEN 308
     0
3070 NX=-NX
3080 BY=BY+NY
3090 IF (BY<>22)*(BY<>3) THEN 311
     0
3100 NY=-NY
3110 CALL GCHAR(BY,BX,CH)
3120 IF CH=32 THEN 3160
3130 IF CH=136 THEN 3270
3140 IF CH=128 THEN 3180
3150 CALL SOUND(200,440,0,880,0,
     523,0)
3160 CALL HCHAR(BY,BX,119)
3170 GOTO 1650
3180 S=S+100
3190 D=D+1
3200 CALL HCHAR(1,29,48+D)
3210 CALL HCHAR(W,Q,128)
3220 CALL SOUND(80,-397,0)
3230 CALL SOUND(80,-319,0)
3240 CALL SOUND(80,-397,0)
3250 BU=0
3260 GOTO 1650
3270 FOR X=1 TO 10
3280 IF (A(X)=BX)*(B(X)=BY) THEN
     3300
3290 NEXT X
3300 A(X)=0
```

Robochase

```
3310 R=R-1
3320 IF R=0 THEN 2100
3330 GOTO 3150
3340 REM  INSTRUCTIONS
3350 CALL CLEAR
3360 CALL SOUND(200,440,0,880,0,
     523,0)
3370 PRINT "       hijiklmno"
3380 PRINT ::
3390 PRINT " YOU HAVE BEEN IMPRI
     SONED"
3400 PRINT :"IN A ROOM BY A MAD
     SCIENTIST"::"  YOU MUST AVO
     ID THE BLUE"::"  ROBOTS AND
      CAUSE THEM TO"
3410 PRINT :"CRASH INTO THE RED
     BARRIERS.":::
3420 GOSUB 3670
3430 PRINT "       hijiklmno"::
     :
3440 PRINT "     ";CHR$(128);"--
     YOU"::"     ";CHR$(136);"--
     A ROBOT"::"  ";CHR$(144);
     "-- A BARRIER"::
3450 PRINT "     ";CHR$(152);"--
     A TELEPORT RECHARGER"::"
     w-- SPUNKY THE MARTIAN"::
3460 GOSUB 3670
3470 PRINT "       hijiklmno"::
     :
3480 PRINT "    YOU CAN CONTROL
     YOUR"::"MOVEMENT BY THE KEY
     BOARD OR"::"      THE JOYS
     TICK."::
3490 PRINT "    Q - UP      P - L
     EFT"::"   A - DOWN    / - RI
     GHT"::
3500 PRINT " USE APPROPRIATE DIR
     ECTIONS"::"    ON THE JOY
     STICK."::
3510 GOSUB 3670
3520 PRINT "       hijiklmno"::
     :
3530 PRINT "YOU CAN TELEPORT TO
     ANOTHER"::"  PLACE ON THE
     BOARD BY"::" PRESSING THE S
     PACE BAR OR"::
3540 PRINT "       THE FIRE BUTTO
     N"::"  HOWEVER, YOU ONLY HA
     VE A"::"CERTAIN NUMBER OF T
     ELEPORTS,"::
3550 PRINT "INDICATED BY A NUMBE
     R AT THE"::"TOP RIGHT HAND
      SIDE OF THE"::"
     SCREEN"
3560 GOSUB 3670
3570 PRINT "       hijiklmno"::
     :
3580 PRINT "  YOU CAN GAIN ADDIT
     IONAL"::"  TELEPORTS BY RUNN
     ING OVER"::"      DIAMON
     DS"::
3590 PRINT "  GETTING SPUNKY WIL
     L ALSO"::" GIVE YOU ANOTHER
      TELEPORT"::
3600 GOSUB 3670
3610 PRINT "       hijiklmno"::
     :
3620 PRINT " EVERY TIME ALL TEN
     ROBOTS"::" ON A LEVEL DIE,
     YOU GO TO"::"   THE NEXT
      LEVEL."::
3630 PRINT "SPUNKY DOES NOT APPE
     AR UNTIL"::"   THE FOURTH
      LEVEL.":::"    GOOD LUCK
     ------"::
3640 GOSUB 3670
3650 CALL CLEAR
3660 GOTO 930
3670 PRINT ::" <HIT ANY KEY TO C
     ONTINUE>"
3680 CALL KEY(3,K,ST)
3690 IF ST=0 THEN 3680
3700 CALL CLEAR
3710 CALL SOUND(200,440,0,880,0,
     523,0)
3720 RETURN
```

Taco Man. . . from p. 32

```
100 REM ************
110 REM * TACO MAN *
120 REM ************
130 REM BY R.S. KEMMETMUELLER
140 REM 99'ER VERSION 2.13.1XB
150 ON ERROR 180
160 CALL INIT :: CALL LOAD(-31878,11)
170 GOTO 190
180 CALL ERR(FF,FF)
190 CALL CLEAR :: CALL SCREEN(5)
200 CALL MAGNIFY(3)
210 CALL CHAR(96,"8181818142422418"&RPT$("0",48),100,"0F3040808040300F"&RPT$("0",48))
220 CALL CHAR(104,"1824424281818181"&RPT$("0",48),108,"F00C020101020CF0"&RPT$("0",48))
230 CALL CHAR(33,RPT$("F",16))
240 CALL CHAR(116,"031F3F7F7F71FFFFDFCB4160703A1F03C0F8FCFEFE8EFFFFBCB4206165CF8C0")
250 CALL CHAR(120,"081030507 87C7C38",81,"00006C783E740000",90,"0000003C2C180000",40,"0010045000822080 0")
260 GOSUB 1210
270 CALL COLOR(1,6,1,7,7,1,8,9,1,9,12,1,10,12,1,12,10,1)
280 FOR CO=3 TO 6 :: CALL COLOR(CO,7,1):: NEXT CO
290 DISPLAY AT(1,11):"TACO MAN"
300 DATA "!!!!!!!!!!!!!!!!!!!!!!!!!!!!!"
310 DATA "!!!!QZ(!!!(Z(Q(ZQ(Z!!!((Z!!!"
320 DATA "!!!!(!Q!!!Q!!!Z!!!(!!!Q!Z!!!"
330 DATA "!!!!ZQ((QZ(ZQZQ(Q(Z(ZQZ(Q!!!"
340 DATA "!!!!!Z!(!!!Q!Z!Z!(!!!(!Z!!!!"
350 DATA "!!!(ZQ!Z!!!(!Q!(!Q!!!Q!(QZ!!"
360 DATA "!!!(!Z!Q!!!Z!(!Q!(!!!Z!Q!(!!"
370 DATA "!!!ZQ(QZ(QZQ(Z(Q(Q(QZ(QZ((!!"
380 DATA "!!!!Z!(!Q!!!(!Q!Z!!!Q!Z!(!!!"
390 DATA "!!!!Q!Z!(!!!Q!Z!(!!!Z!(!(!!!"
400 DATA "!!!!(!Q!Z!!!Z!(!(!!!Z!Z!Z!!!"
410 DATA "!!!!Z(Q(Z(QZQ((Z(QZ(ZQ(QZ!!!"
420 DATA "!!!!!Z!(!Q!!!Q!(!!!Z!(!(!!!!"
430 DATA "!!!!!(!Z!(!!!Z!Q!!!(!Q!Z!!!!"
440 DATA "!!!!!Q!(!(!!!(!Z!!!Q!Z!Q!!!!"
450 DATA "!!!!(ZQZ(ZQZ(Q(ZQ(ZQ(ZZQ(!!!"
460 DATA "!!Z(Q!Z!(!Z!!!Z!!!(!Q!(!QZQ!"
470 DATA "!!(!Z!(!Q!(!!!Q!!!Z!(!Q!Z!(!"
480 DATA "!!Q(Z!Q!Z!Q!!!Z!!!(!Q!Z!(QZ!"
490 DATA "!!!!Z(Q(ZQZ(QZ(Q(Z(Q(QZZQ!!!"
500 DATA "!!!!!!!!!!!!!!!!!!!!!!!!!!!!!"
510 DATA "                             "
520 RESTORE
530 D=247
540 FOR A1=1 TO 22 :: READ A2$
550 DISPLAY AT(A1+1,1):A2$
560 NEXT A1
570 SCREEN=1 :: MEN=4 :: SCORE=0 :: DISPLAY AT(23,1):RPT$(CHR$(100),MEN):: DISPLAY AT(24,1):"SCORE:" :: DISPLAY AT(24,7):SCORE
580 SP=1 :: PP=0
590 CALL HCHAR(21,17,120)
600 CALL HCHAR(4,7,120):: CALL HCHAR(4,27,120):: CALL HCHAR(8,6,120):: CALL HCHAR(8,28,120):: CALL HCHAR(19,5,120):: CALL HCHAR(19,29,120)
610 FOR RR=2 TO 20 STEP 2 :: SP=SP+1
620 CO=INT(RND*4)+7 :: IF CO=8 THEN 620 ELSE CALL SPRITE(#SP,116,CO,RR*8-7,256,0,(-1)^(INT(RND*2))*(INT(RND*8)+SCREEN))
630 NEXT RR
640 R=11 :: C=17 :: RV,RV1=-1 :: CV,CV1=0 :: CH,CH1=96 :: CALL SPRITE(#1,CH,12,R*8-7,C*8-7)
650 PP=PP-1 :: IF PP<0 THEN 720
660 IF PP THEN 710
670 CALL COLOR(1,6,1)
680 FOR SP=2 TO 11
690 CO=INT(RND*4)+7 :: IF CO=8 THEN 690 ELSE CALL COLOR(#SP,CO)
700 NEXT SP
710 IF PP=5 THEN CALL COLOR(1,14,1)
720 IF PP>0 THEN 740
730 CALL COINC(ALL,UG):: IF UG=-1 THEN 930
740 CALL JOYST(1,X,Y):: CALL KEY(1,K,S)
750 IF ((X=0 AND Y=0)OR(X<>0 AND Y<>0))AND S=0 THEN 800
760 IF Y=4 OR K=5 THEN CH1=96 :: RV1=-1 :: CV1=0
770 IF X=4 OR K=3 THEN CH1=100 :: RV1=0 :: CV1=1
780 IF Y=-4 OR K+1=1 THEN CH1=104 :: RV1=1 :: CV1=0
790 IF X=-4 OR K=2 THEN CH1=108 :: RV1=0 :: CV1=-1
```

Continued on p. 42

MORE FUN FOR THE TI 99/4A™

Funware®

ST. NICK.™

Wicked witches, banished from Halloween, have swooped down on poor old St. Nick for Christmas. And cranky, as they usually are, they've cast their evil spell on the elves, who are scattering toys everywhere with wild abandon. Christmas itself is in dire jeopardy! Help St. Nick ward off the witches and get his workshop back in order. If this sounds like jolly good fun, just try the challenge of 42 screen levels!

(CAUTION: Don't smile while playing. The elves may scatter your teeth!)

A SUBSIDIARY OF:

C R E A T I V E

HERE'S $5.00 FOR HELPING ST. NICK!™

Your TI 99/4A will be programmed for Christmas excitement with St. Nick. And to really get that fun flowing, each St. Nick game will come with a $5 savings coupon towards your next FUNWARE™ purchase. So check out our hot new games. Fast-moving, challenging, multi-level games that turn your TI into a spectacular entertainment machine. Get into Funware,™ and get the fun you've been missing!

RABBIT TRAIL.
All you want is a few carrots to nibble on... and the whole world's trying to stop you. Run through a maze of rabbit holes as you dodge weasels, hawks, traps and cars. Seven different screens provide an unending challenge that'll leave you hopping mad!

RABBIT TRAIL

AMBULANCE.
Helpless citizens are dropping like flies... somehow you've got to get them all to the hospital before they expire! Speed through city streets, dodging runaway cars, trains, and traffic in a fast-paced strategic adventure guaranteed to get your pulse racing!

AMBULANCE

PIPES.
A simple problem becomes an incredibly complex strategic challenge: Help Arlo, the Absent-Minded Plumber, create a water-supply system for the whole neighborhood. Select the right pipes from the factory, plan your delivery system, hook up the joints and pray that poor Arlo hasn't left the valves open!

PIPES

DRIVING DEMON.
Arcade-style auto racing for the TI 99/4A, in a desperate race against the clock and the competition. Weave through the cars, skid through the oil slicks, then floor it down the straightaway. Your long-range scanner warns you of obstacle cars in your path, but you'll need lightning reflexes to react in time. Mile-a-minute thrills!

DRIVING DEMON

S O F T W A R E

Taco Man . . . from p. 39

```
800 CALL GCHAR(R+RV1,C+CV1,CHK)
    :: IF CHK<>33 THEN RV=RV1 :
    : CV=CV1 :: CH=CH1 ELSE CAL
    L GCHAR(R+RV,C+CV,CHK):: IF
     CHK=33 THEN 650
810 R=R+RV :: C=C+CV :: CALL PA
    TTERN(#1,CH):: CALL LOCATE(
    #1,R*8-7,C*8-7)
820 IF CHK=32 THEN 650
830 IF CHK<>120 THEN 860
840 PP=26-SCREEN :: CALL COLOR(
    1,9,1)
850 FOR SP=2 TO 11 :: CALL COLO
    R(#SP,1):: CALL SOUND(-1,55
    *SP,0):: NEXT SP
860 IF CHK=40 THEN SCORE=SCORE+
    5
870 IF CHK=90 THEN SCORE=SCORE+
    10
880 IF CHK=81 THEN SCORE=SCORE+
    15
890 DISPLAY AT(24,7):SCORE :: C
    ALL SOUND(-1,-6,0)
900 CALL HCHAR(R,C,32)
910 D=D-1 :: IF D=0 THEN 1040
920 GOTO 650
930 FOR DEL=27 TO 1 STEP -4
940 FOR CH=96 TO 108 STEP 4
950 CALL PATTERN(#1,CH)
960 CALL SOUND(-1,110+DEL+CH/4,
    0)
970 NEXT CH
980 NEXT DEL
990 CALL SOUND(-100,-2,4):: CAL
    L SOUND(-100,-3,3):: CALL S
    OUND(-100,-2,2):: CALL SOUN
    D(-100,-1,1):: CALL SOUND(-
    1000,-1,0)
1000 MEN=MEN-1 :: IF MEN=-1 THEN
     1110 ELSE DISPLAY AT(23,1)
    :RPT$(CHR$(100)),MEN)
1010 SP=1
1020 FOR RR=3 TO 21 STEP 2 :: SP
    =SP+1 :: CALL LOCATE(#SP,RR
    *8-7,256):: NEXT RR
1030 GOTO 640
1040 FOR SP=2 TO 11 :: CALL DELS
    PRITE(#SP):: NEXT SP
1050 CALL COLOR(1,6,1)
1060 FOR DEL=1 TO 20
1070 CALL SOUND(INT(RND*200)+1,1
    10+INT(RND*220),INT(RND*3))
1080 NEXT DEL
1090 SCREEN=SCREEN+1
1100 RESTORE :: GOTO 530
1110 CALL DELSPRITE(#1)
1120 FOR DEL=1 TO 5
1130 DISPLAY AT(12,10)SIZE(9)BEE
    P:RPT$(CHR$(100),4);" ";RPT
    $(CHR$(108),4)
1140 DISPLAY AT(12,10)SIZE(9)BEE
    P:"GAME OVER"
1150 NEXT DEL
1160 DISPLAY AT(15,1):"DO YOU WA
    NT TO PLAY AGAIN?Y/N"
1170 CALL KEY(0,K,S):: IF S=0 TH
    EN 1170
1180 IF K=78 OR K=110 THEN CALL
    CLEAR :: END
1190 IF K<>89 AND K<>121 THEN 11
    70
1200 CALL CLEAR :: GOTO 290
1210 FOR CO=1 TO 14 :: CALL COLO
    R(CO,2,1):: NEXT CO
1220 DISPLAY AT(2,11):"TACO MAN"
1230 DISPLAY AT(6,2):"PLEASE REL
    EASE ALPHA LOCK."
1240 DISPLAY AT(8,1):"USE JOYSTI
    CK 1 OR THE E, X,"
1250 DISPLAY AT(9,1):"S, AND D K
    EYS."
1260 DISPLAY AT(11,2):CHR$(100);
    " TACO MAN"
1270 DISPLAY AT(12,2):CHR$(81)&"
     HAMBURGER 15 POINTS"
1280 DISPLAY AT(13,2):CHR$(90)&"
     TOMATO SLICE 10 PTS."
1290 DISPLAY AT(14,2):CHR$(40)&"
     SHREDDED CHEESE 5 PTS."
1300 DISPLAY AT(15,2):CHR$(120)&
    " CHILI PEPPER"
1310 DISPLAY AT(16,2):CHR$(116)&
    CHR$(118)
1320 DISPLAY AT(17,2):CHR$(117)&
    CHR$(119)&" TACOVORE"
1330 DISPLAY AT(20,1):"WANT FURT
    HER DIRECTIONS?Y/N"
1340 CALL KEY(0,K,S):: IF S<1 TH
    EN 1340
1350 IF K=78 OR K=110 THEN CALL
    CLEAR :: CALL COLOR(2,11,1)
    :: RETURN
1360 IF K<>89 AND K<>121 THEN 13
    40
1370 CALL CLEAR
1380 PRINT " THE OBJECT OF TACO
     MAN:"
1390 PRINT "RUN YOUR TACO SHELL
    THROUGH"
1400 PRINT "THE CORRIDORS AND GO
    BBLE UP"
1410 PRINT "AS MUCH HAMBURGER, T
    OMATO"
1420 PRINT "SLICES, AND SHREDDED
     CHEESE"
1430 PRINT "AS YOU CAN BEFORE BE
    ING EAT-"
1440 PRINT "EN BY ONE OF THE TAC
    OVORES."
1450 PRINT :" A LITTLE BACKGROU
    ND ON"
1460 PRINT "THIS PARTICULAR GROU
    P OF"
1470 PRINT "TACOVORES: THEY DREA
    D CHILI"
1480 PRINT "PEPPERS AND IF YOU H
    AVE ONE"
1490 PRINT "IN YOUR TACO SHELL T
    HEY WILL"
1500 PRINT "RUN OFF AND HIDE, BU
    T THEIR"
1510 PRINT "APPETITE DRIVES THEM
     ON AND"
1520 PRINT "IT WON'T BE LONG BEF
    ORE"
```

Taco Man

```
1530 PRINT "THEY'RE BACK ON YOUR
     TRAIL."
1540 PRINT :" PRESS ANY KEY TO
     CONTINUE."
1550 CALL KEY(0,K,S):: IF S<1 TH
     EN 1550 ELSE CALL CLEAR
1560 PRINT " THE GAME STARTS WI
     TH 5"
1570 PRINT "TACOS AND ENDS WHEN
     THEY"
1580 PRINT "HAVE ALL BEEN DEVOUR
     ED."
1590 PRINT :" PRESS ANY KEY TO
     START."
1600 CALL KEY(0,K,S):: IF S<1 TH
     EN 1600 ELSE CALL CLEAR :: C
     ALL COLOR(2,11,1):: RETURN
```

Enter

Arithmetex . . . from p. 30

a speedy response by offering 5 points for a merely correct answer and anywhere from 6 to 10 points for a fast correct answer. But the game won't register your speedy answer until after it has laboriously completed its segment-by-segment drawing of the problem, including the equal sign. It is fairly easy to get 10 points on a one-digit answer. You just press the correct key and hold it down while the program leisurely completes its graphics. Then the program is likely to give you ten points for being there with your answer.

But if the problem requires a two-digit answer, don't be too hasty with your speedy response. The game will register the first digit of your answer, but if you've pressed the two digits in rapid succession, the second will be ignored, and your speed will have been for naught. You must hover over the second key until all of the problem and the first digit of your answer have been traced in slow motion. Whatever its intentions, the program rewards patience more than it does speed. Your speedy answer is unfairly delayed by the program's own slow processes.

In the old way of teaching Math it was considered a good idea that children recite their times tables up to twelve off-by-heart and rapid-fire. Even granting this debatable notion (gaining a little lost ground of late), it is still difficult to see the virtue of a speedy response to 37 plus 52, or 9 times 17. *Arithmetex* also requires the responding child to enter the left-hand digit first, even though the right-hand digit will be the first known unless the child has memorized the answer to 37 plus 52 or 9 times 17.

What Do You Learn?

In addition to the ill-advised and unfairly administered speed requirement, *Arithmetex* poses its problems with a lack of generosity we don't like to see in any pedagogue. There is no provision for self-correction if you hit the wrong key or reconsider an answer. Furthermore, if the answer the program is looking for is a one-digit number, it won't accept a two-digit answer. If the correct answer is a two-digit number, the machine waits for a second digit even if your one-digit response, albeit incorrect, is your absolute intent. The machine won't move on until you compound your error with any old second digit. What has been learned?

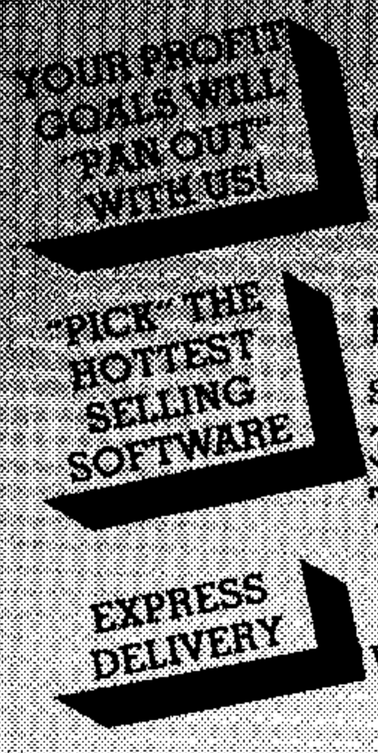
Arithmetex offers two levels of difficulty for children ages 6 to 12. But if you switch from low level to high level, you won't find more advanced material. You'll merely multiply more numbers on the order of 9 times 17 and fewer like 5 times 3. A twelve-year-old would be bored by the difficult level. At that age a child has already studied long division, fractions, decimals, ratios, some Algebra, and even probability. The Multiplication option on the high level of difficulty in *Arithmetex* offers questions as easy as 10 times 2 to as "hard" as 5 times 17, and seems to strike a middle ground of difficulty with 11 times 3. In Dorothy Parker's famous words, it runs the gamut from A to B.

Continued on p. 68

EL DORADO TRADING GROUP

Your Western U.S. Exclusive Texas Instruments High Technology Distributor.

With over 1 million 99/4A owners wanting software and accessories, now's the time to jump on our wagon!



Our support services will help your company reach its profit goals by keeping you abreast of the newest and fastest turning high technology items in the line.

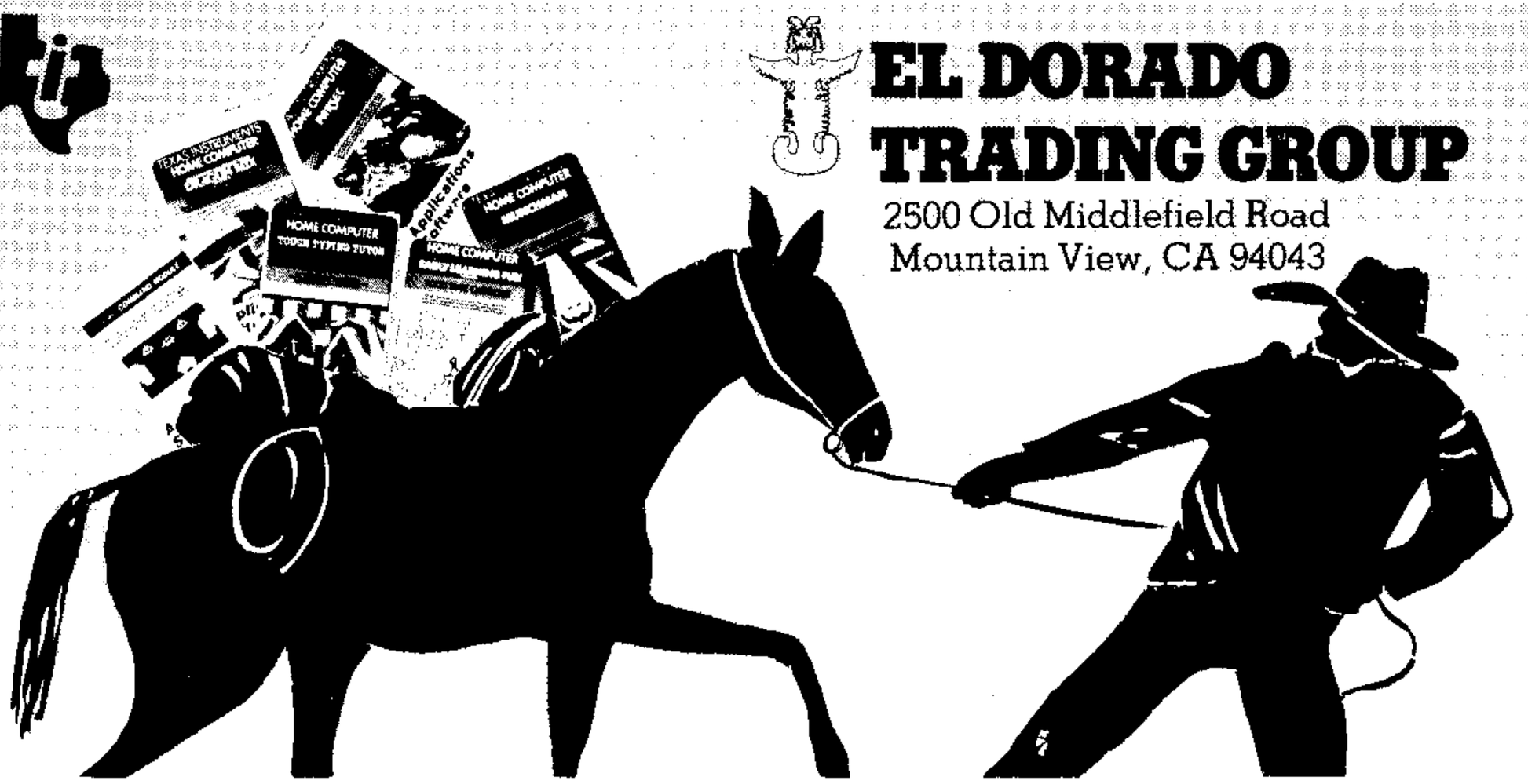
Our trained sales staff will assist you in ordering the "fastest turning" software by suggesting new packages and proven best sellers such as Parsec (PHM 3112), Early Learning Fun (PHM 3002), Dragon Mix (PHM 3117), TI Invaders (PHM 3053), Touch Typing Tutor (PHM 3064) and Munch Man (PHM 3051).

Our extensive inventory allows your order to be processed within a 24 hour time frame.

DON'T MISS OUT ON THE RUSH!

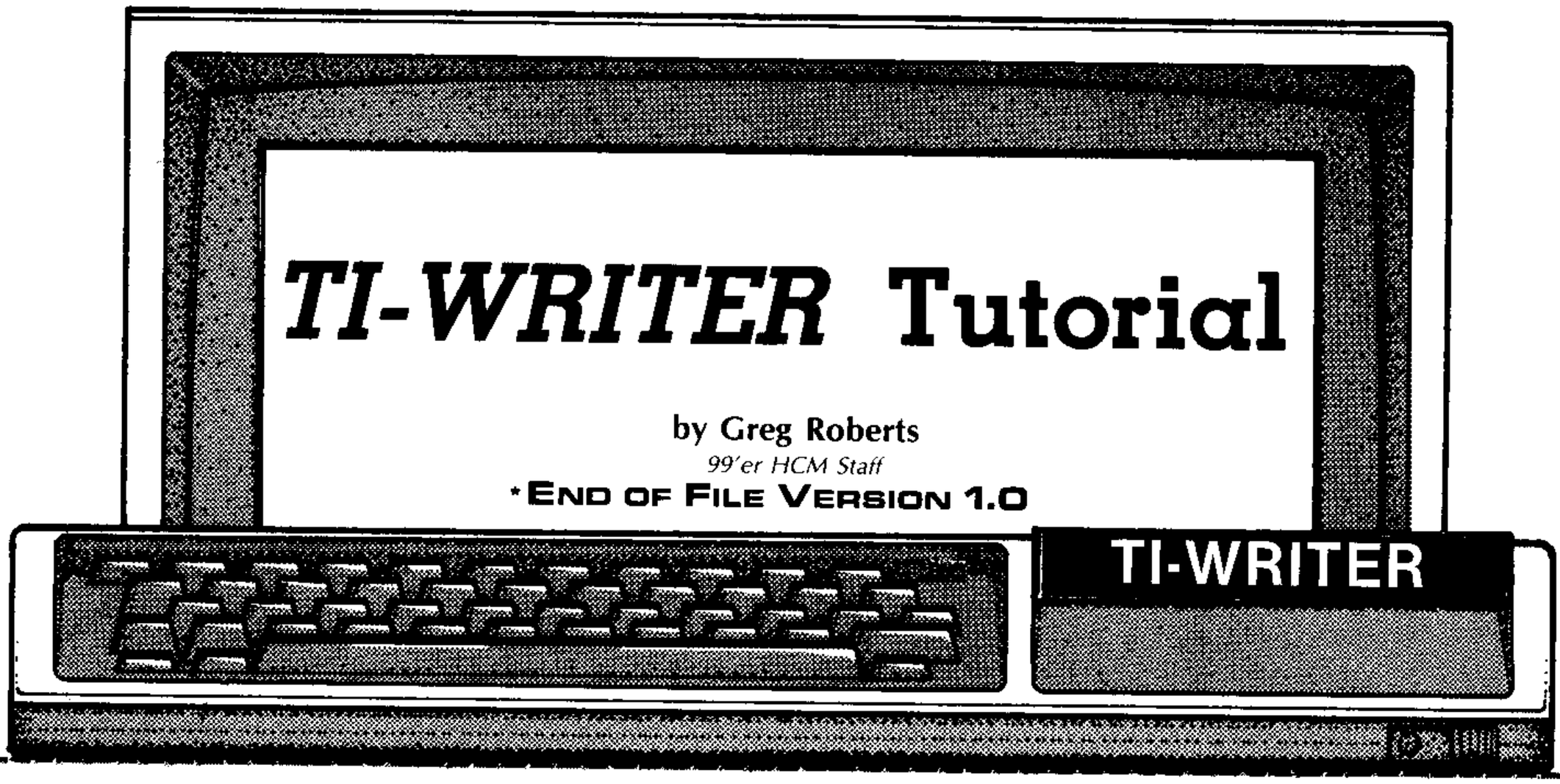
Call us today, **TOLL FREE 800-227-8292**, in California **800-248-2224** except 415 area code call **969-6600** for a confidential "High Technology" dealers price catalog.

All dealers with an assigned customer number and a retail storefront will want to act NOW to ensure delivery during the primary 4th quarter selling season.



EL DORADO TRADING GROUP

2500 Old Middlefield Road
Mountain View, CA 94043



TI-WRITER Tutorial

by Greg Roberts
99'er HCM Staff

*END OF FILE VERSION 1.0

Last month we showed how to edit simple paragraphs with the Insert and Delete commands, and how to SaveFiles on diskette. Some writers may need no more; those basics will probably cover most everyday writing. Even your business letters can be composed within that narrow framework, as long as you write entirely from the left margin without tabs.

Those who like more variety in the format of their writing can use the *TI-WRITER* tab function. As convenient as a typewriter tab, it is even easier to set. Assuming your cursor is blinking at the extreme left-hand side of the screen, press [CTRL][I] or [FCTN][7] (whichever combination best fits your phalanges), and the cursor will automatically move one tab designation to the right. Keep holding down these keys, and it will move across the screen, tab by tab. You don't, of course, have to stick with the tab spacings automatically set by the *TI-WRITER* program (known as the "default" settings). Simply press [FCTN] [9], then T for tabs. Now you can make any changes you like by moving or deleting the T designations and pressing [ENTER].

Mode Moves

To edit blocks of type set off by tabs, you will need to put a carriage return [CTRL] [8] after every line in order to keep your work from completely reformating itself. This can be a tedious process if you have lots of indentations, but there is an alternative. *TI-WRITER* features a special mode for making tables: Fixed mode. So far, we have been writing in Word Wrap mode, the logical choice for editing and reformating large, uniform blocks of type. For setting up tables using a variety of tab settings, the Fixed mode works better: It does not "wrap" your lines, nor reformat them, and carriage return marks are not needed in this mode.

To change from one mode to the other, press [CTRL] [0]. Word Wrap mode is identified by a solid flashing rectangle, whereas the Fixed mode's rectangle is hollow. Now try using this mode to make a table, say, of frequently-used telephone numbers. Use the tabs set by the *TI-WRITER* program or choose your own. Once you've completed a line in Fixed mode, just press [ENTER] and your line will stay in place.

And how do you edit a line in Fixed mode? To insert a character or word, press [FCTN] [2]. As you add to the existing line, notice that it moves to the right. Be careful. If you try to crowd too much onto the line, it will run right off the screen. To add the room you need, you have to put in a space between lines by pressing [CTRL] [O]. Incidentally, be careful not to confuse the 0 (zero) with the O (capital O).

Text Transplants

Writing in Word Wrap mode again, you may find it useful to move an entire paragraph to another location in the manuscript. You may laugh now, all those who write good outlines and are otherwise blessed with organized minds. The rest of us may take comfort in the *TI-WRITER* Move command. After pressing [FCTN] [9], type [M] plus [ENTER]. To see how this process works, fill the screen with some of your favorite limericks or other drivel. The command line will then let you type in three line numbers—for the beginning and end lines of the paragraph you want moved, and for the line to which you are moving. Make sure there is a single space between each of the three line numbers, then press [ENTER]. That's all there is to it:

MOVE start line, stop line, after line:

23 67 12

Lines 23-67 will be inserted after line 12. (Note spaces between numbers.)

Sometimes in your writing you may find that a word used many times throughout the text will have to be changed. For instance, Ti-Writer to *TI-WRITER*, hog to hawg, or Russia to Soviet Union. You can make a single correction that extends to every occurrence of the word, if desired, by using the ReplaceString command.

Press [FCTN] [9] to return to the command line, then [RS] and [Enter]. The computer will display a "prompt line":

REPLACE ENTER/OLD STRING/NEW STRING/ :

To replace a word or phrase with another word or phrase, type a slash [/], the word to be replaced, another slash, the new phrase, and a closing slash. Then press [ENTER]. Example:

/LEROY/FREDDY/

A note of caution: When using ReplaceString to edit words or phrases written in Fixed mode, you must be in Fixed mode when you make your corrections; otherwise,the replacement will delete your tabs.

A close relative of the ReplaceString is the FindString command. Once you get back to the command line, type [FS] and press [ENTER]. The program searches for the first instance of the specific word or phrase you have entered between the slash marks:

/HAWG HEAVEN/

***TI-WRITER* . . . from p. 45**

Upon finding the first occurrence of the word, the computer marks it with the cursor, so that you can quickly find and edit a specific line. In some instances the program will not be able to find the string and will simply return to Edit mode, displaying the last line of your file at the top of the screen. What went wrong? Most likely, one part of the string was on one line and the rest was on the next line. Remember, FindString looks for the string exactly as it is typed into the prompt line.

Features such as these would be useful only within a fairly long block of text, naturally, and that suggests two more useful *TI-WRITER* commands: Roll Up and Roll Down. These commands are executed respectively, as [FCTN] [6] and [FCTN] [4]. They let you move a whole screen up or down instantly, instead of a line at a time.

Next Paragraph, is much like the "roll" commands except for this distinction: The "roll" commands always move your cursor 24 lines at a time (one screen), whereas Next Paragraph moves down to the first word in each paragraph, no matter what its length may be. Press [CTRL] [4] to try it.

These three commands may not be absolutely necessary to word processing; rather they are among the "little extras" that the programmers built into this product to make it competitive in that canine-cannibal world out there.

TI-WRITER: At Home in the Office

by Jerry Shepherd
P.O. Box 941330
Schaumburg, IL 60194

As an independent manufacturers representative, I find myself constantly having to write to customers of the four companies represented by my agency. I could hire a typist, I suppose, but I find that cleaning up a document for typing is almost as much work as the typing itself.

Almost two years ago I committed myself to using the 99/4A as my business computer. I had my accounting all set up on *Multiplan* and had templates designed to do quotation calculations and sales commissions, and to keep auto, travel and entertainment expenses for two salespeople and me. I purchased *TI-WRITER* to replace my old worn-out electric typewriter.

The Epson MX80FT/III does a good job of printing out my reports from *Multiplan*, and I have played around with *TI-WRITER* enough to know that it has the potential to do every last bit of correspondence. I now have everything I need.

For my business letters I use the TL (TransLiterate) command of *TI-WRITER* to let me input the Epson print codes for emphasized characters. The results are not letter quality, of course, but the printing is fast and clear, and the dot-matrix image is acceptable in this kind of correspondence.

Customers on Diskettes

Our order forms are set up as files for each manufacturer we represent. Following the instructions for form letters in the *TI-WRITER* manual, all we have to do is incorporate our handwritten notes into a file by letting the program prompt us for the information needed to complete an order. *TI-WRITER* and the Epson will crank out a beautiful typewritten order on three-part, self-carbon, tractor-feed paper (available at almost any computer supply store). We keep one copy, and mail one to the manufacturer and another to the customer.

Again, using the form letter method in the manual, we have set up data files for our entire customer list and have printed up self-adhesive labels for mailing to customers and manufacturers. On occasion, when we need an envelope for someone not on our mailing label file, we use a file called ENV which is formated to type on standard business envelopes. We simply call up that file, type in the needed name and address and tell *TI-WRITER* to print.

We provide a lot of price quotations for customers, and we find there are two ways to do this effectively. For a long quote (over ten lines) we print out the quote with *Multiplan*. For shorter quotes, we simply print out the quotation from *Multiplan* to a disk file and then call it as a portion of the letter to the customer on *TI-WRITER*. In either case, we have the option of using *TI-WRITER* to edit the quote prior to mailing.

"I have played around with TI-WRITER enough to know that it has the potential to do every last bit of correspondence."

With *TI-WRITER*, you can use the data-file as a mailing list. At some point, almost any business will need to make mass mailings of price changes or other special announcements calling for a form letter. Using the data-file created by *TI-WRITER*, we can make a cover letter seem personal by inserting variable inputs such as name, address, city, state and Zip Code.

Magnetized Mail

We keep disk copies of all correspondence. For large customers with whom we carry on a heavy correspondence, we assign an entry with the name of the customer for the disk name. Smaller customers can be filed several to a disk. This saves us a lot of space and makes it easy to go back and research a customer file. On these same disks we keep files of notes (indexed by date) and memos for our internal use.

TI-WRITER can be used with some mailing list management programs such as *NAME-IT* from Extended Software. One thing *TI-WRITER* can't do is sort data-files by alphabetical or numerical codes. So far, this hasn't been a problem for us, but we can foresee the day when it might be.

TI-WRITER does a great job for our small business, but it does have a few weak spots. By far, the most frustrating weakness is that it always sends an automatic form feed before it starts printing—always, whether you want one or not. [For readers who also find this to be a problem, we have a solution in "Letters to the Editor"—*Ed.*] We have learned to get around this by letting the program send the form feed to the Epson buffer and then momentarily shutting the printer off to cancel the form feed. The program does have an option to stop the printing at the end of every page to change paper when yor are using single sheets such as letterhead stationery.

Another initial irritation is the lack of any HELP screens. Nevertheless, the reference card is good so that most users may not need the HELP screen anyway. Overall, the program is very user-friendly.

The 40-column screen can be annoying to use. The program is designed in such a way that you can see up to 80 columns of text, but only in three "windows" that shift back and forth to show one section of the document at a time. There is a way around this though: simply type your text in a 40-column setup initially, and then, prior to printing, go back and reformat before saving to disk and printing.

TI-WRITER has proven to be a real time- and energy-saver for our small company. It does require some practice before you feel comfortable with it, but after a short time, you can make it "dance." I heartily recommend it in conjunction with *Multiplan* for any small- to medium-sized business.

Letters . . . from p. 7

```
310 PRINT #2:LN$(L+5)&CHR$(132)&"S@"&CHR$(190)&CHR$(200)&CHR$(1)&"0"&CHR$(176)&CHR$(201)&LN$(L+4)&CHR$(0)
320 FOR I=65 TO L+64 :: PRINT #2:LN$(L+I-59)&IF$(I)&CHR$(0):: NEXT I
330 PRINT #2:LN$(2*L+6)&CHR$(132)&"K@"&CHR$(190)&CHR$(200)&CHR$(1)&"7"&CHR$(176)&CHR$(157)&CHR$(200)&CHR$(5)&CL$&CHR$(130)&CHR$(139)&CHR$(0)
340 PRINT #2:LN$(2*L+7)&CHR$(134)&CHR$(201)&LN$(L+4)&CHR$(0):CHR$(255)&CHR$(255)::CLOSE #2 :: DISPLAY AT(23,21)BEEP:"COMPLETE" :: END
```

If you merely want to catalog a diskette without having to remove the cartridge you may be using, the following BASIC program (from a recent issue of TI's own user newsletter) will print the catalog on the screen:

```
100 CALL CLEAR
110 DIM TYPE$(5)
120 TYPE$(1)="DIS/FIX"
130 TYPE$(2)="DIS/VAR"
140 TYPE$(3)="INT/FIX"
150 TYPE$(4)="INT/VAR"
160 TYPE$(5)="PROGRAM"
170 INPUT "MASTER DISK (1-3)?":I
180 I=INT(I)
190 IF (I<1)+(I>3)=-1 THEN 170
200 OPEN #1:"DSK"&STR$(I)&".",INPUT ,RELATIVE,INTERNAL
210 INPUT #1:A$,J,J,K
220 DISPLAY "DSK";STR$(I);" -DISKNAME= ";A$:"AVAILABLE= ";K;"USED= ";J-K
230 DISPLAY :"FILENAME    SIZE    TYPE    P":"------------------------";
240 INPUT #1:A$,I,J,K
250 IF LEN(A$)=0 THEN 330
260 DISPLAY :A$;TAB(12);J;TAB(17);TYPE$(ABS(I));
270 IF ABS(I)=5 THEN 300
280 B$=" "&STR$(K)
290 DISPLAY SEG$(B$,LEN(B$)-2,3);
300 IF I>0 THEN 240
310 DISPLAY TAB(28);"Y";
320 GOTO 240
330 CLOSE #1
```

If you save this short program on every diskette, you'll be able to run this program and catalog each diskette when you're using it.

The procedure to suppress the automatic first page advance when using TI-WRITER's Text Formatter is cumbersome and really only worthwhile if you print lots of short documents. The following procedure (also from a recent issue of TI's user newsletter) prevents the extra page from being printed:

Leave your printer turned off when you start formating. Enter all the information as usual until you reach the question, PAUSE AT END OF PAGE? The default is N for "No." Change this to Y (for "Yes"), and press [ENTER]. WORKING. . . will appear at the bottom of your screen. If you now turn your printer on, PRESS ENTER TO CONTINUE followed by a flashing cursor will appear on the screen. Now press [ENTER]; your document will begin printing without the initial form feed. The disadvantage of the procedure, and the reason it's only useful with short documents, is the necessity to press [ENTER] for each subsequent page.

Academic Questions

Once again let me repeat (this is my third letter to you) what a great magazine you are producing. Please keep up the good work.

I have some questions:

1. Why do you present programs for disk without including similar appropriate entry lines for cassette (e.g., *Tex-cipher* and *Verbose*). One article stated "With a few simple modifications this program will work on cassette." If these modifications are so simple, why not include them in an addendum to a program presentation? This would broaden the number of readers who could use the program as you present it.
2. Professsor Holl's Pocket Programs are great. But, the Professor is just like the professors I had in college. They give you a problem but not the answers. How about presenting the solution to the problems presented in "A Cure for the Listless," "Pocket Battleship," and "Tower of Hanoi." It's a challenge to try to work out the problems, but it would be a delight if the answers could be provided by the Professor.
3. The article, "Never Out of Sorts," left me out of sorts because I didn't know how to get the sorts to sort words into alpha order. Could you supply a transition from number sorting to alpha sorting? PLEASE!!

Please respond if you have time, or better yet, publish these items so all of your readers can benefit.

Robert B. Richards
Ridgewood, NJ

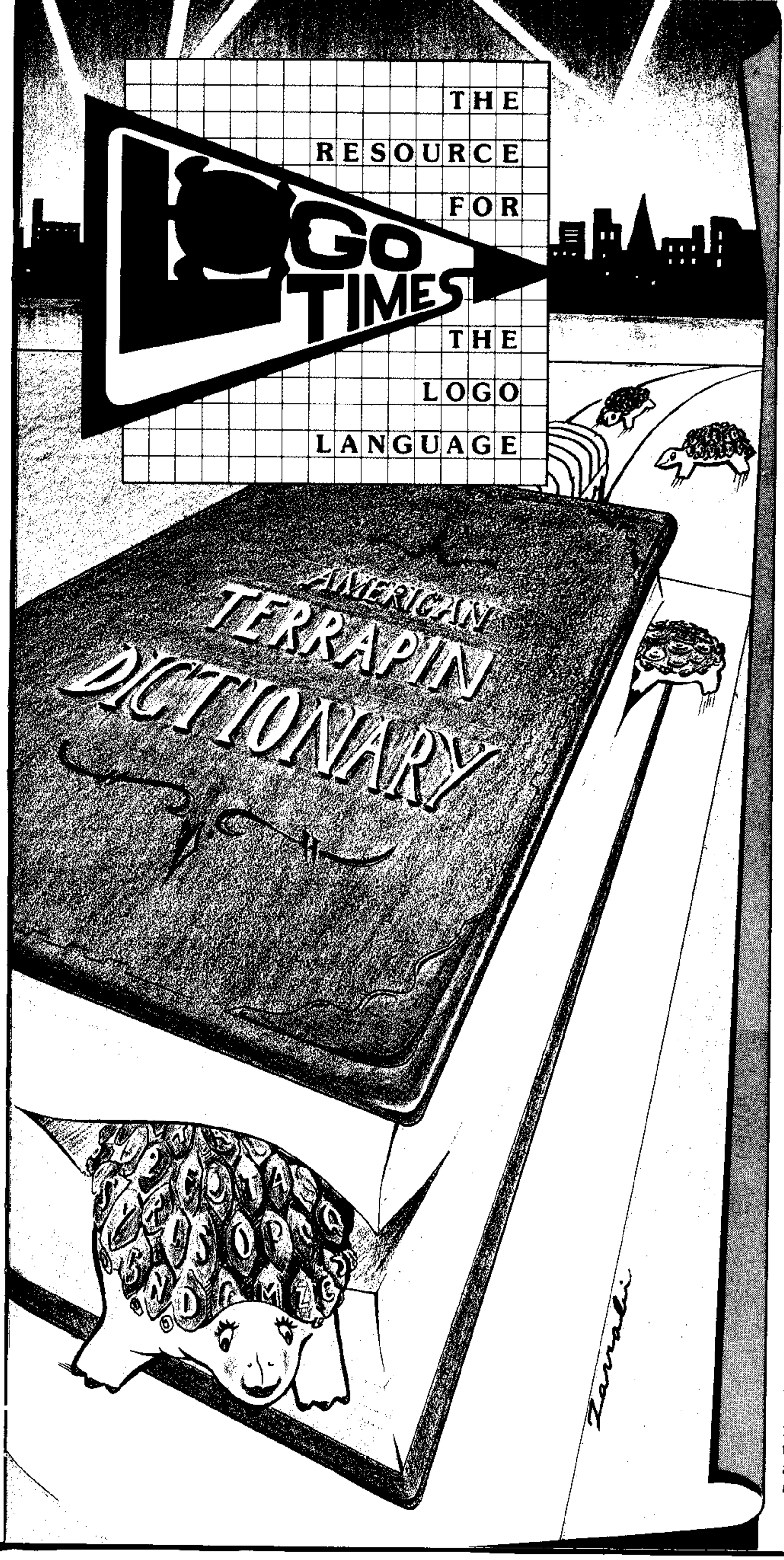
You'll notice that in recent issues, we have provided programming alternatives for a variety of conditions. In cases where programs can be readily modified to use devices other than those contained in the program, we will include the necessary changes—time and space permitting.

As for Professor Holl's problems, you're right. It is just like college. And the professors didn't always have the best answers—as most will quickly admit. Often the students had ingenious solutions, and we suspect that's the case here as well. Keep your eyes peeled for a future notice about readers sending us solutions for publication.

To convert the sorting routines in "Never Out of Sorts" from numeric to alphabetic, you have to change the variable type of all the variables to be sorted from numeric to alphabetic. For instance, in the selection sort, all references to the numeric array A( ) need to become references to the string array A$( ). All variables that hold a temporary value from the array to be sorted also need to be string variables. For instance, in the heap sort, line 260, S is set equal to A(L). For alphabetic sorts, that line would have to set S$ equal to A$(L). Similarly, any variable that is set equal to, or compared with, a member of the array to be sorted must become a string variable.

CAN'T FIND IT?

Our computer literacy ace is still polishing his prose for our promised *99'er Computer Literacy Course* which will debut next month. The portable computer packed up and got out of town this month for vacation, but will come back tan and fit in next month's *Portable Computing Magazine* feature.



L·O·G·O LEXICON

by Roger B. Kirchner
Contributing Editor

Spelling and vocabulary drills are not generally very amusing. But they *can* be fun if we use LOGO to program them. How, you may ask, can LOGO conduct a spelling or vocabulary drill if it can't ask us the word? LOGO can display a word for just an instant and then erase it. The instant should be just long enough to see the word. Then we can try to type it from memory. If we get it right, LOGO can give us another word. If we get it wrong, LOGO can give the same word again. Besides practice in spelling, the method of presenting a word for an instant might also be helpful in speed reading. It would give us practice in recognizing words seen for a very short time.

Let's put our idea into practice and define a procedure to present a list of words to be spelled:

```
TO SPELL :LIST
CS
IF :LIST = [ ] THEN STOP
MAKE "W FIRST :LIST
TYPE "SPELL PC 32
WAIT 180
PRINT :W
WAIT 2
CS
MAKE "ANS FIRST RL
TEST :ANS = :W
IFT PRINT [GOOD ] MAKE "LIST B
F :LIST
IFF PRINT [TRY AGAIN ]
WAIT 180
SPELL :LIST
END
```

SPELL will continue as long as the input list is not empty. LOGO prepares us to see a word by showing SPELL: for three (180 sixtieths) seconds. Then the first word of the list is presented for only 1/30 of a second. RL causes a pause to input a line from the keyboard, and LOGO stores the first word of our response as the value of ANS. This is then compared with the word presented. If we are right, LOGO tells us GOOD and removes the word from the list. If we are wrong, LOGO tells us TRY AGAIN. After a two second delay, SPELL is activated again. If the input list is empty, the procedure stops; otherwise, we are asked to spell the first remaining word of the list.

To try out the spelling procedure, enter SPELL [HOW ARE YOU]. To make it easier to start a drill, we can use a word to store a list of words. For example, entering MAKE "WORDS RL and then entering a

Introduction

LOGO Times is an information resource for anyone interested in participating in the creation of their own *personal* language—one that will easily allow them to communicate with a computer in a totally new audiovisual realm of applied imagination, exploration, and self-discovery. The articles on these pages concern the use of the new TI LOGO language, but readers do *not* need any additional software or equipment (or even a computer) to understand and learn from the material presented here.

If readers want to actually *experience* a TI LOGO environment, they will need either a TI-99/4 or TI-99/4A computer, the Expansion Memory peripheral, and TI LOGO Command Cartridge. A disk drive, although convenient to have, is *not* required; a user's work may alternately be saved on cassette tape, printed out on the TI Thermal Printer, or hand copied into a notebook (for later re-keyboarding).

In each issue, one or more of the articles may reference or build upon the topics discussed in a previous article. It is therefore recommended that for maximum benefit and understanding, new readers obtain the appropriate back issues of *99'er Home Computer Magazine* containing *LOGO Times* articles.

NOTICE

LOGO Times is actively soliciting articles. Manuscripts should be typed double-spaced, and accompanied by a cassette tape or disk if containing any lengthy procedures or graphics.

Send all materials to:

LOGO Times Editorial Dept.
99'er Home Computer Magazine
1500 Valley River Dr., Suite 250
Eugene, OR 97401

All mail directed to the Letters-to-the-Editor column (*Letters on LOGO*) will be published in accordance with the conditions set forth on *99'er Home Computer Magazine's* Masthead page.

Our Contributing Editors

Henry Gorman, Jr.
Department of Psychology
Austin College
Box 1584
Sherman, TX 75090

Roger B. Kirchner
Department of Mathematics
Carleton College
Northfield, MN 55057

William M. Goodman
7 Vanier Drive, No. 502
Guelph, Ontario
Canada N1G 2L1

sequence of words will store the words in a list as the value of WORDS. To present these words to yourself, enter SPELL :WORDS. You can also make up words for a friend, and have her do the same for you. Be sure to give the lists different names. Don't forget that when you save procedures, your entire workspace is saved. When the procedures are recalled, the values of all the words, as well as the procedure definitions, are recalled. This means that your lists can be saved.

Our method for presenting words can also be used for vocabulary drills. We can have LOGO check a definition in a dictionary instead of simply checking the spelling of a word. We can use either English or foreign words. Of course, we have to teach LOGO the definitions. A *dictionary* can be a list of pairs: The first item is the word to be defined, and the second item the definition (or translation).

Suppose, for example, that we want to put the three German words *wie, gehen,* and *Sie* in a German-English dictionary which is to be named GTOE.DICT. The definitions are to be *how, to go,* and *you,* respectively. We could make this dictionary by entering MAKE "GTOE.DICT RL, and then entering [WIE HOW] [GEHEN [TO GO]] [SIE YOU]. (TO GO is entered as a list because it is two words.) We can then check the dictionary by entering :GTOE.DICT. In this case, LOGO will respond with TELL ME WHAT TO DO WITH [[WIE HOW] [GEHEN [TO GO]] [SIE YOU]].

Another way to define a dictionary is to write a procedure for the purpose. Then we could make up any number of dictionaries:

```
TO MAKEDICT :DNAME
CS
PRINT (SE "DEFINING :DNAME)
MAKE :DNAME []
PRINT [ENTER WORDS AND DEFINIT
IONS:]
1
MAKE "ENTRY RL
IF :ENTRY = [] THEN STOP
MAKE :DNAME LPUT :ENTRY THING
:DNAME
GO "1
END
```

This definition uses some fancy properties of LOGO. First of all, notice that the input of the procedure is a name to be given a value. For example, to give GTOE.DICT the same value as above, we would enter MAKEDICT "GTOE.DICT. The dialogue in this case would be:

```
DEFINING GTOE.DICT
ENTER WORDS AND DEHNITIONS:
>WIE HOW
>GEHEN [TO GO]
>SIE YOU
>
?
```

Whatever we type after a '>' is put into a list which is put into :GTOE.DICT. The procedure stops when we just press [ENTER].

It isn't necessary for a dictionary to contain words and definitions. It could contain any kind of paired information—for example, states and capitals. Suppose we were to enter MAKEDICT "CAPITALS and then give the following information after the >'s:

```
DEFINING CAPITALS
>MINNESOTA [ST. PAUL]
>OREGON SALEM
>TEXAS AUSTIN
>
?
```

(ST. PAUL is entered as a list because it is two words.) The value of CAPITALS will then be the list [[MINNESOTA [ST. PAUL]] [OREGON SALEM] [TEXAS AUSTIN]].

Now that we know how to make dictionaries, we can turn to the problem of writing DRILL. The idea is that if we enter DRILL :GTOE.DICT, or DRILL :CAPITALS, we should be presented with the first item of each pair and be asked to respond with the second item of each pair.

```
TO DRILL :DICT
IF :DICT = [] THEN STOP
CS
MAKE "W FIRST FIRST :DICT
TYPE [RESPOND TO:] PC 32
WAIT 120
PRINT :W
WAIT 2
CS
MAKE "ANS RL
IF BF :ANS = [] THEN MAKE "AN
S FIRST :ANS
MAKE "ENTRY LPUT :ANS (SE :W
[])
TEST MEMBER? :ENTRY :DICT
IFT PRINT [GOOD] MAKE "DICT B
F :DICT
IFF PRINT [TRY AGAIN]
WAIT 120
DRILL :DICT
END
```

Note that the value of W is the first item of the first pair in :DICT. The second member of this pair is a word or a list of words. If we respond with just one word, then :ANS is made equal to that word. Otherwise :ANS will be the list of entered words. A pair is then constructed for the value of ENTRY. (We use LPUT because if :ANS is a list, we want that list to be the second member of the pair.) Then we check to see whether the pair is in the dictionary. From this point on, DRILL is the same as SPELL.

DRILL makes use of a utility procedure MEMBER?, which we have to define as follows:

```
TO MEMBER? :ITEM :LIST
IF :LIST = [] THEN OUTPUT "FA
LSE
IF :ITEM = FIRST :LIST THEN OU
TPUT "TRUE
OUTPUT MEMBER? :ITEM BF :LIST
END
```

This definition assumes—correctly—that if an item is on a list, it is at some point the *first* member of a list obtained by removing successive items from the beginning of the list. MEMBER? is thus recursive: In the third line, it calls itself and removes the first item from :LIST

Try out SPELL and DRILL. If a word is presented for too short a time, then modify the definitions to make LOGO wait a little longer. If it is too easy to see the word, you can change WAIT 2 to WAIT 1, or remove the line altogether. You may also want to change the other delays. These procedures will give you valuable practice with the list primitives in LOGO as well as with your spelling and vocabulary.

DeBUGS on DisPlay

99'er Program Bug

You LOGO lovers out there who have been struggling along with our typeset program listings, take heart! You'll be glad to know that now they're untouched by human hands from computer to press. Those of you who were perplexed by *Mosaic Designs* (in July 1983, page 67) will find that the figures and the complete listing reproduced here will generate the LOGO mosaic masterpieces you may have missed out on.

1	4	4	1	1	4	4	1
2	3	3	2	2	3	3	2
2	3	3	2	2	3	3	2
1	4	4	1	1	4	4	1
1	4	4	1	1	4	4	1
2	3	3	2	2	3	3	2
2	3	3	2	2	3	3	2
1	4	4	1	1	4	4	1

Figure 1

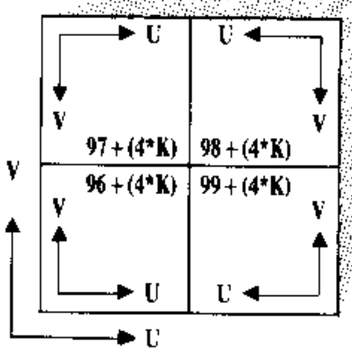


Figure 2

Figure 1 shows how the tiles defined in Figure 2 are repeated throughout the mosaic.

Figure 2 indicates the direction in which the level L-1 designs are generated relative to a level L design.

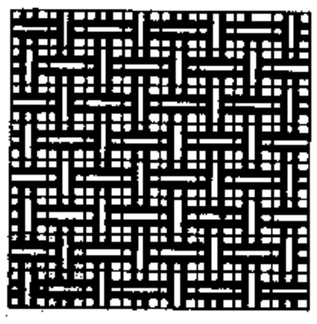


Figure 4

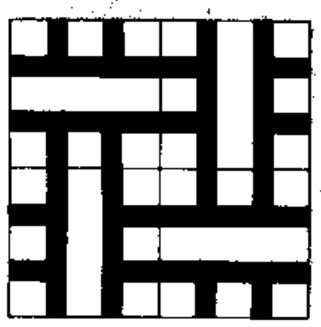


Figure 3

Figure 3 provides a tile design that will result in attractive mosaic (as in Figure 4).

```
TO DESIGN :P0
CS
CB :BLACK
MAKE "P :P0
1:
IF :P < 0 THEN MAKE "P :P0
TELL TILE 96 + 4 * :P
DES 8 4 1 1 5 0
MAKE "P :P - 1
GO "1
END

TO DES :X :Y :U :V :L :K
SCOLOR
IF :L = 0 THEN PLOT :X :Y :K STOP
DES :X :Y :U :V :L - 1 1
DES :X :Y + H * :V :U ( - :V )
:L - 1 2
DES :X + H * :U :Y + H * :V (
- :U ) ( - :V ) :L - 1 3
DES :X + H * :U :Y ( - :U ) :V
:L - 1 4
END
```

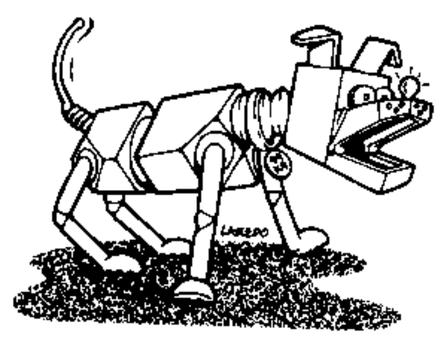
```
TO H
MAKE "H 1
REPEAT :L [MAKE "H :H * 2 ]
OUTPUT :H - 1
END

TO SCOLOR
IF NOT :L = 3 THEN STOP
MAKE "C1 2 + RND 13
MAKE "C2 1 + 14 * RND 2
SC SE :C1 :C2
END

TO PLOT :X :Y :K
PT 95 + 4 * :P + :K :X 23 - :Y
END

TO RND :N
1:
MAKE "RND 10 * RANDOM + RANDOM
IF :RND < 100 / :N * :N THEN O
O :RND / ( 100 / :N )
O "1
END
```


August's Choice



The votes are in! The winner of the August, 1983, B.A.R.C. Back author competition is Fred Ellis for his article *Graphic Persuasion*. Our hearty congratulations and a $100 prize go to Mr. Ellis.

America ... from p. 21

```
100 REM ************
110 REM * AMERICA *
120 REM ************
130 REM BY CHRISTOPHER FLOTTA
140 REM 99'ER VERSION 2.13.1XB
150 H=1 :: CALL MAGNIFY(3):: CALL SCREEN(2)
160 DISPLAY AT(12,5)ERASE ALL:"LETS BUILD AMERICA" :: OPTION BASE 1
170 DIM ST$(50),CA$(50),DUP(50)
180 RESTORE 1450 :: FOR I=1 TO 50 :: READ ST$(I):: NEXT I :: FOR I=1 TO 50 :: READ CA$(I):: NEXT I
190 FOR I=3 TO 8 :: CALL COLOR(I,16,1):: NEXT I
200 CALL COLOR(2,15,1,9,10,1,13,10,1,1,4,1,10,16,1,12,13,1,11,11,1,14,14,1)
210 RESTORE 220 :: FOR A=1 TO 45 :: READ B,A$ :: CALL CHAR(B,A$):: NEXT A :: GOTO 350
220 DATA 40,FFFFFFFFFFFFFFFFFF,33,FFFFFFFFFFFFFFFF,37,FF7F7F3F3F1F0303,138,203C7FFFFE7C7830,139,0070000000464302
230 DATA 140,0000081C3C0C06000,141,0000000C1ECC,39,F8F2F7FEFEFCFCF8,120,FFFFFFFFFFFED890,121,FFFFFFFFF3D18
240 DATA 42,00008080C0C0E0E0F0F0F8F8FCFCFEFE00000000000000000000000000000000
250 DATA 125,0000808040400000,126,7E7C3810,127,7F1F1F1F1F0F0F0F,122,FFFFFFFFFFFFFFFFFF
260 DATA 116,FFFFFFFFFFFFFFFFFF7F7F1F0F07070800008080C0C0E0E0E0E0C0C080800000
270 DATA 124,0F0F070703030100,136,FFFFFFFFFFFFFFFFFF,47,FFFAF4F8F0A0C080,115,000383C7E7CF9FFF
280 DATA 64,0F1F3F3F7F7FFFFF,92,FFFFFDF4E0C0C080,91,00008080C0C0E0E0,93,E0E080,96,FFFFFFFFFFFFFFFF
290 DATA 103,0000000001010101,101,0303030307070707,102,0F0F0F0F1F1F1F1F,35,FFFFFFFFFFFFFFFFFF
300 DATA 135,FFFF7F7F3F3F1F1F,99,0F0F070703,100,FEF0F0C0,128,7F3F3F1F0F070301,129,7F7F7F3F3F1F1F0F
310 DATA 130,0F0F070703030000,131,FCFCFCFCF8F8F80C,132,FFFFFFFFFEFEFEFFFFFFFEF8F0E080,62,FFFFFFFFFFFFFFFF
320 DATA 134,FFFCE0000000000,136,7F3F1F0F070301,104,FFFFFFFFFFFFFFFF1F1F3F7F7FFFFF7F7F3F3F7F7FFFFF7F
330 DATA 107,7F7F6F3F6F7F7F3F,108,3F7F7F7F7F7F3F3F7F7F3F3F7F7F3F1F0707030303010101FF7F7F3F3F3F3F1F
340 DATA 97,FFFFFFFFFFFFFFFFFF,112,FFFFFFFFFFFFFFFF
350 CALL CLEAR
360 CALL SCREEN(6)
370 IF RS=50 THEN 1350
380 DISPLAY AT(22,1):"STATE PLEASE"
390 ACCEPT AT(23,1)BEEP:NS$
400 ATS=ATS+1 :: P=0 :: J=0
410 FOR I=1 TO 50
420 IF ST$(I)=NS$ THEN P=I
430 NEXT I
440 IF P=0 THEN 1250
450 FOR I=1 TO 50
460 IF DUP(I)=P THEN J=1
470 NEXT I
480 IF J=1 THEN ATS=ATS-1 :: GOTO 1340
490 DUP(H)=P :: H=H+1 :: RS=RS+1 :: W=P
500 ON INT((P-.5)/10)+1 GOTO 510,530,560,590,620
510 ON P GOSUB 670,680,690,700,710,740,750,760,770,800
520 GO TO 640
530 P=P-10
540 ON P GOSUB 810,820,830,840,850,860,870,880,890,900
550 GOTO 640
560 P=P-20
570 ON P GOSUB 910,920,930,940,950,960,970,980,990,1000
580 GOTO 640
590 P=P-30
600 ON P GOSUB 1010,1020,1030,1040,1050,1060,1070,1090,1100,1110
610 GOTO 640
620 P=P-40
630 ON P GOSUB 1120,1130,1140,1180,1190,1200,1210,1220,1230,1240
640 DISPLAY AT(1,1):"STATE TRIES=";ATS
650 DISPLAY AT(1,17):"RIGHT=";RS
660 GO TO 1270
670 CALL VCHAR(15,23,40,3):: CALL VCHAR(15,24,40,3):: RETURN
680 CALL HCHAR(2,1,35,3):: CALL HCHAR(3,1,35,3):: CALL HCHAR(4,1,35,3):: RETURN
690 CALL VCHAR(13,8,35,3):: CALL HCHAR(16,8,36):: CALL VCHAR(13,9,35,4):: CALL VCHAR(13,10,35,4):: RETURN
700 CALL VCHAR(13,19,40,3):: CALL VCHAR(13,20,40,3):: RETURN
710 CALL HCHAR(9,4,105):: CALL HCHAR(10,4,106):: CALL HCHAR(11,4,107):: CALL HCHAR(12,4,108)
720 CALL HCHAR(13,4,109):: CALL HCHAR(14,4,110):: CALL HCHAR(15,5,111):: CALL VCHAR(9,5,104,6)
```

```
730 CALL VCHAR(13,6,104,3):: CALL VCHAR(14,7,104,2):: RETURN
740 CALL HCHAR(10,11,40,4):: CALL HCHAR(11,11,40,4):: CALL HCHAR(12,11,40,4):: RETURN
750 CALL HCHAR(7,30,33):: RETURN
760 CALL HCHAR(9,29,112):: RETURN
770 CALL HCHAR(18,23,120):: CALL HCHAR(18,24,121):: CALL HCHAR(19,25,126):: CALL HCHAR(20,27,127)
780 CALL SPRITE(#1,42,13,137,209,#2,116,13,158,217):: CALL HCHAR(20,29,125):: CALL HCHAR(18,25,122,3)
790 CALL HCHAR(19,27,122):: CALL VCHAR(20,28,122):: CALL HCHAR(21,27,124):: RETURN
800 CALL HCHAR(15,25,104,2):: CALL HCHAR(16,25,104,3):: CALL HCHAR(17,25,104,3):: RETURN
810 CALL HCHAR(20,6,138):: CALL VCHAR(19,5,139):: CALL VCHAR(18,4,140):: CALL VCHAR(17,2,141):: RETURN
820 CALL VCHAR(4,7,35,5):: CALL VCHAR(6,8,35,3):: CALL VCHAR(7,9,35,2):: RETURN
830 CALL VCHAR(8,20,40,4):: CALL VCHAR(8,21,40,4):: RETURN
840 CALL VCHAR(8,22,97,3):: CALL VCHAR(8,23,97,3):: RETURN
850 CALL HCHAR(7,18,104,2):: CALL HCHAR(8,18,104,2):: RETURN
860 CALL HCHAR(11,15,97,3):: CALL HCHAR(12,15,97,3):: RETURN
870 CALL HCHAR(11,22,35,3):: CALL HCHAR(12,21,35,4):: RETURN
880 CALL VCHAR(16,19,35,2):: CALL VCHAR(16,20,35,2):: CALL HCHAR(18,20,37):: CALL HCHAR(18,21,39):: RETURN
890 CALL HCHAR(4,31,64):: CALL HCHAR(5,31,92):: CALL HCHAR(4,32,91):: CALL HCHAR(5,32,93):: RETURN
900 CALL HCHAR(9,27,104,2):: CALL HCHAR(10,29,104):: RETURN
910 CALL HCHAR(6,29,40,2):: RETURN
920 CALL VCHAR(5,23,35,3):: CALL VCHAR(5,24,35,3):: RETURN
930 CALL HCHAR(4,18,97,3):: CALL HCHAR(5,18,97,2):: CALL HCHAR(6,18,97,2):: RETURN
940 CALL VCHAR(15,21,112,3):: CALL VCHAR(15,22,112,3):: RETURN
950 CALL VCHAR(9,18,122,4):: CALL VCHAR(9,19,122,4):: CALL VCHAR(12,20,122):: RETURN
960 CALL HCHAR(4,8,40,6):: CALL HCHAR(5,8,40,6):: CALL HCHAR(6,9,40,5):: RETURN
970 CALL HCHAR(8,14,33,4):: CALL HCHAR(9,14,33,4):: CALL HCHAR(10,15,33,3):: RETURN
980 CALL VCHAR(9,6,40,4):: CALL VCHAR(9,7,40,5):: RETURN
990 CALL HCHAR(5,30,33):: RETURN
1000 CALL VCHAR(7,29,96,2):: RETURN
1010 CALL VCHAR(13,11,104,4):: CALL VCHAR(13,12,104,3):: CALL VCHAR(13,13,104,3):: RETURN
1020 CALL HCHAR(5,27,122,2):: CALL HCHAR(6,26,122,3):: RETURN
1030 CALL HCHAR(13,26,96,4):: CALL HCHAR(14,26,96,4):: RETURN
1040 CALL HCHAR(4,14,33,4):: CALL HCHAR(5,14,33,4):: RETURN
1050 CALL VCHAR(8,24,40,3):: CALL VCHAR(8,25,40,3):: RETURN
1060 CALL HCHAR(13,14,136,5):: CALL HCHAR(14,16,136,3):: CALL HCHAR(15,16,136,3):: RETURN
```

Continued on p. 61

Come Work & Play In A Unique Environment . . .

Follow the Oregon Trail to the Beautiful Willamette Valley

Join a Dynamic Team of Creative Individuals Who Thrive on Challenge and Cherish Their Quality of Life.

We are offering—

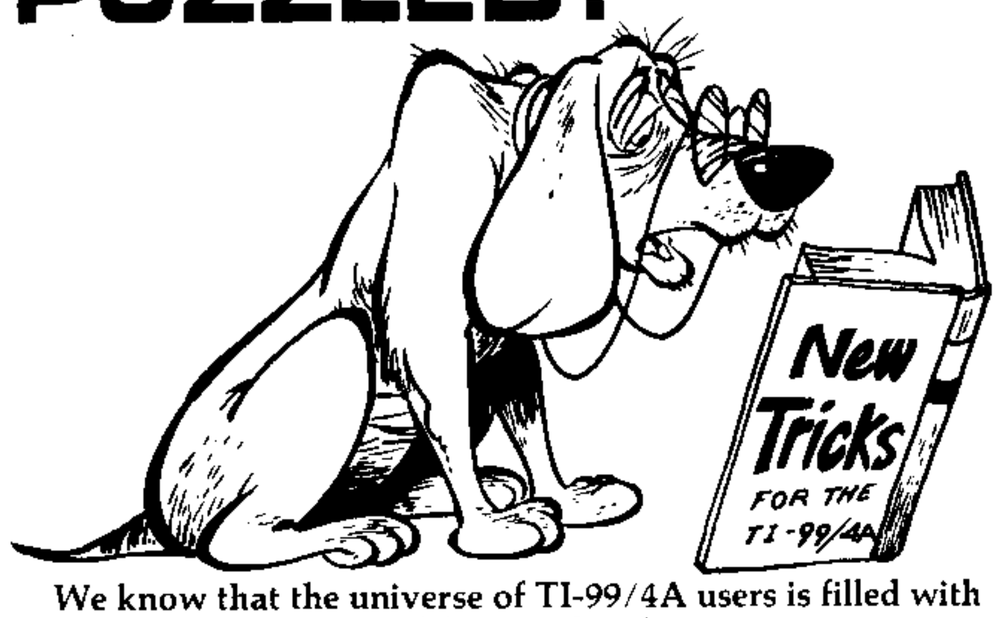
- Technical Writers/Editors
- Software Engineers
- Logic Designers
- Applications Programmers
- Advertising & Marketing Professionals

—The Opportunity of a Lifetime

Send your resume in strictest confidence to:

99'er Talent Hunt
Emerald Valley Publishing Co.
1500 Valley River Drive, Suite 250
Eugene, Oregon 97401

PUZZLED?



We know that the universe of TI-99/4A users is filled with folks who are captivated by puzzles, brain-teasers, conundrums and the like. That is, after all, part of the delight of programming computers—finding a faster way, saving memory, discovering double-duty programming tricks.

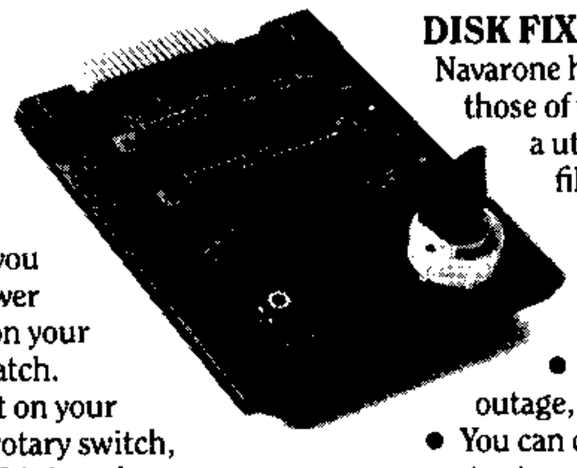
We'd like you to pose your programming puzzles for our readers to rack their brains over. If you have a solution for your puzzle, send it along; if you don't, we'll leave it to our readers to come up with one. (Don't despair if an answer doesn't appear right away—we'd like to give all our readers a chance to submit their best solutions.)

Please keep your "Puzzles-to-the-Editor" brief, and send them in typewritten and double-spaced if at all possible. Then watch for their appearance on our "Letters-to-the-Editor" pages. If you have a solution, send it—preferably on cassette, diskette or as a listing—to:

Letters-to-the-Editor/Puzzles Desk
99'er Home Computer Magazine
P. O. Box 5537
Eugene, Oregon 97405

Two for your TI from Navarone.

WIDGIT



Navarone's *Widgit* cartridge expansion board lets you plug in up to three command modules at one time. With three sockets and a rotary switch, you can choose between your favorite modules without plugging and unplugging. Our built-in reset button allows you to reset the computer without turning the power off and on. These features save wear and tear on your console. Its protective rubber feet will not scratch.

The *Widgit* plugs directly into the game slot on your TI-99/4A. Select your module by turning the rotary switch, press the reset button and you're ready to go. It's just that easy.

The *Widgit* is priced at a low $39.95. When ordering, ask for catalog number: WD-01.

DISK FIXER

Navarone has created this powerful Assembly language program for those of you who want more control of your TI-99/4A. *Disk Fixer* is a utility program written to access by sector rather than by file name.

Consider these features:

- A *Help* feature to give you menu driven self-documentation.
- You can locate and access a single sector of any files.
- Recover lost data from blown directories due to power outage, operator error, software bugs or improper handling.
- You can change any byte on any sector or move data from one sector to another.
- Locate lost data with the *Search and Find* feature.

This program is available on cartridge only and requires 32K memory expansion and at least one disk drive.

The *Disk Fixer* is priced at $39.95. When ordering ask for catalog number: AS-02.

NAVARONE

Visa and Master Charge welcome. To order by phone, call (408) 866-8579.
If ordering by mail, send Cashier's check or Money Order. (Personal checks take 10 days to clear.) C.O.D.'s add $5.00.
Please state items you are ordering, quantity and the price of each item. Allow 4 to 6 weeks delivery. Add 5% of total order for shipping and handling. CA residents add 6.5% sales tax.

For more information on other Navarone products request our catalog. Dealer inquiries welcome.

Navarone Industries, 510 Lawrence Expressway, #800, Sunnyvale, CA 94086, (408) 866-8579

Come see the complete 99'er-WARE product line in Las Vegas at the Winter Consumer Electronics Show in Emerald Valley Publishing Co.'s World Class Software exhibit—Space #6824 West Hall Annex.

Excerpts from the

99'ER DIGEST™

of news & happenings in the Home Computer world

NEW CONSOLE INTRO KILLED FOR 1983

TI's mid-October announcement that a new home computer console will not be introduced during the remainder of 1983 has set off speculation that the 99/8 as presently designed—and not yet formally announced—might be shelved permanently. Industry analysts hypothesize that (1) the new machine would be too expensive to produce for the targeted market segment, and that (2) TI would not be able to produce enough of the special peripheral equipment to meet demand. In light of the Texas giant's new positioning of the 99/4A and the mammoth ad campaign mounted in fourth quarter [see related item below], the postponement of the new product introduction is not viewed as a serious problem among Lubbock-watchers—as long as a new, market-suitable machine can be shipped in quantity before the Christmas season next year.

PRICE NO LONGER COMPELLING REASON TO BUY HOME COMPUTERS

TI's new marketing campaign is based on consumer research, results showing that the "early-innovator market" is now saturated. The lion's share of future sales must come from consumers who have been confused or intimidated by home computers. Rapid advances in technology have created a mass-market fear of buying too soon or paying too much. The Texas marketeers have surmised that "kids' education is today's most relevant and believable desire." The 99/4A will not, however, be positioned just as an educational tool for children—other TV spots slated for late fall emphasize the computer's appeal to adults. New print ads, also aimed at adults, stress the machine's powerful capabilities and differentiate it from the competition.

KIDS REPLACE COSBY IN 4TH-QUARTER TI MEDIA BLITZ

TI's new advertising blitz—without long-time spokesman Bill Cosby—focuses on the educational benefits of the 99/4A Home Computer rather than the price. Price promotion will be left up to the retailer; educating consumers will be TI's prime responsibility. The fourth-quarter media budget is said to be in the neighborhood of $30 million—more than 3 times larger than 1982's fourth-quarter spending. 80% of the expenditure is slated for TV, with full network coverage for family-oriented programming.

TI ACQUIRES P&G MARKETING EXPERTISE

With the hiring of Peter Field, a 19-year Procter & Gamble veteran, TI, like several other computer makers, has turned to a consumer-product executive for marketing expertise. Field, most recently general manager of the P&G coffee division, is said to be TI's first senior manager from outside the company's rank-and-file—representing a radical departure for a technology-driven company. In a related action, TI is moving most of its Consumer Group's administrative and marketing staff to Dallas from Lubbock—a better location for attracting advertising and marketing talent. The move sets the stage for closer cooperation and shared resources with the Austin-based Data Systems Group, makers of the TI Professional Computer known as Pegasus.

MORE TRAINING SITES AND CARTRIDGE PRICE REDUX HIGHLIGHT 4TH QUARTER

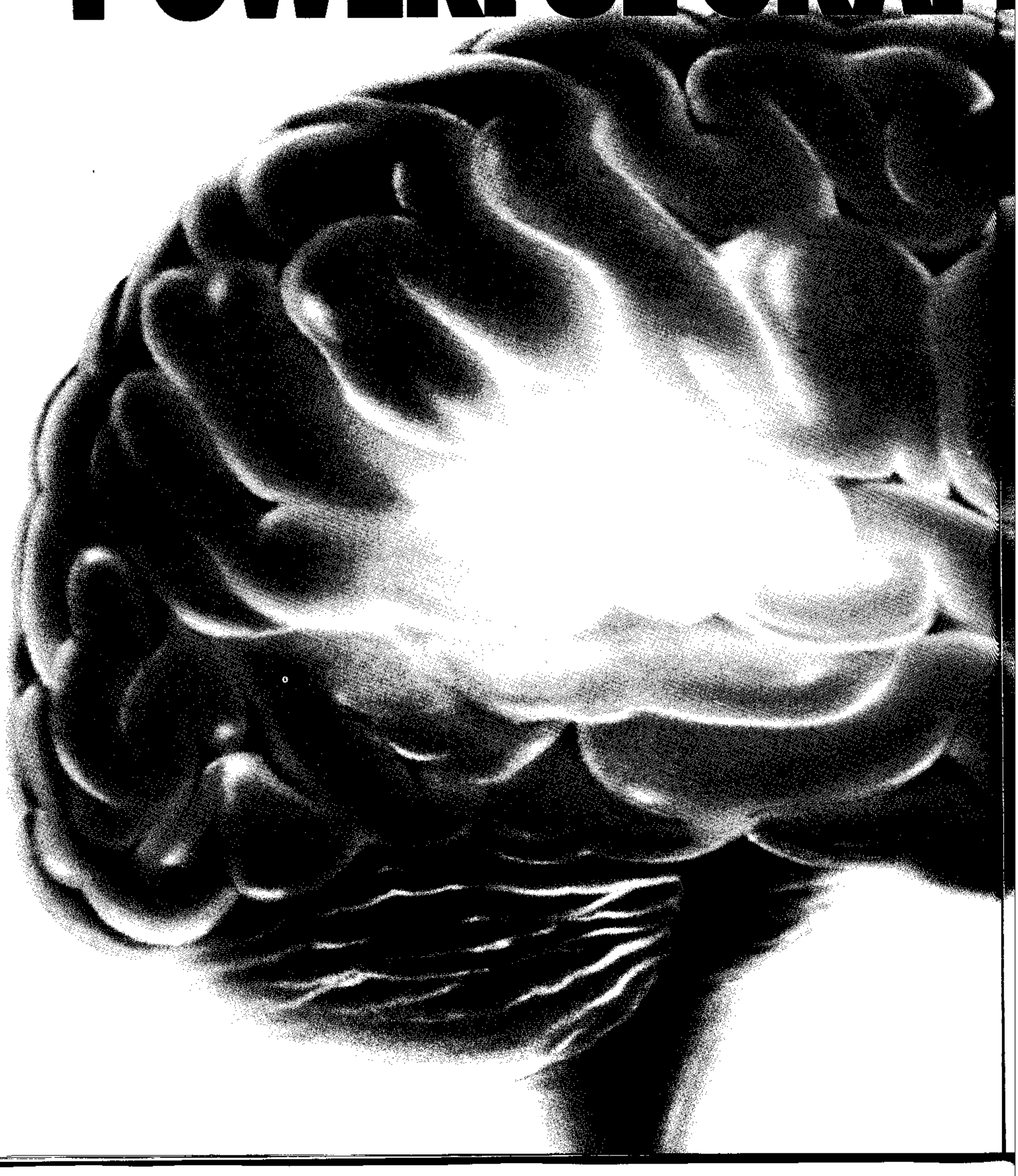
Up to 25 more training sites are slated for TI's burgeoning network of Computer Advantage Clubs (TICAC). Expected to bring the total to 230 sites, the expansion will service the 5 hours of free training offered with the purchase of a 99/4A. Current penetration includes all 50 states and the top 100 major metro markets. In an attempt to also stimulate software sales, the suggested retail prices of 44 educational, information management, and entertainment cartridges have been cut an average of 35%.

"MAGNIFICENT SEVEN" TO APPEAR ON HOLIDAY SHELVES

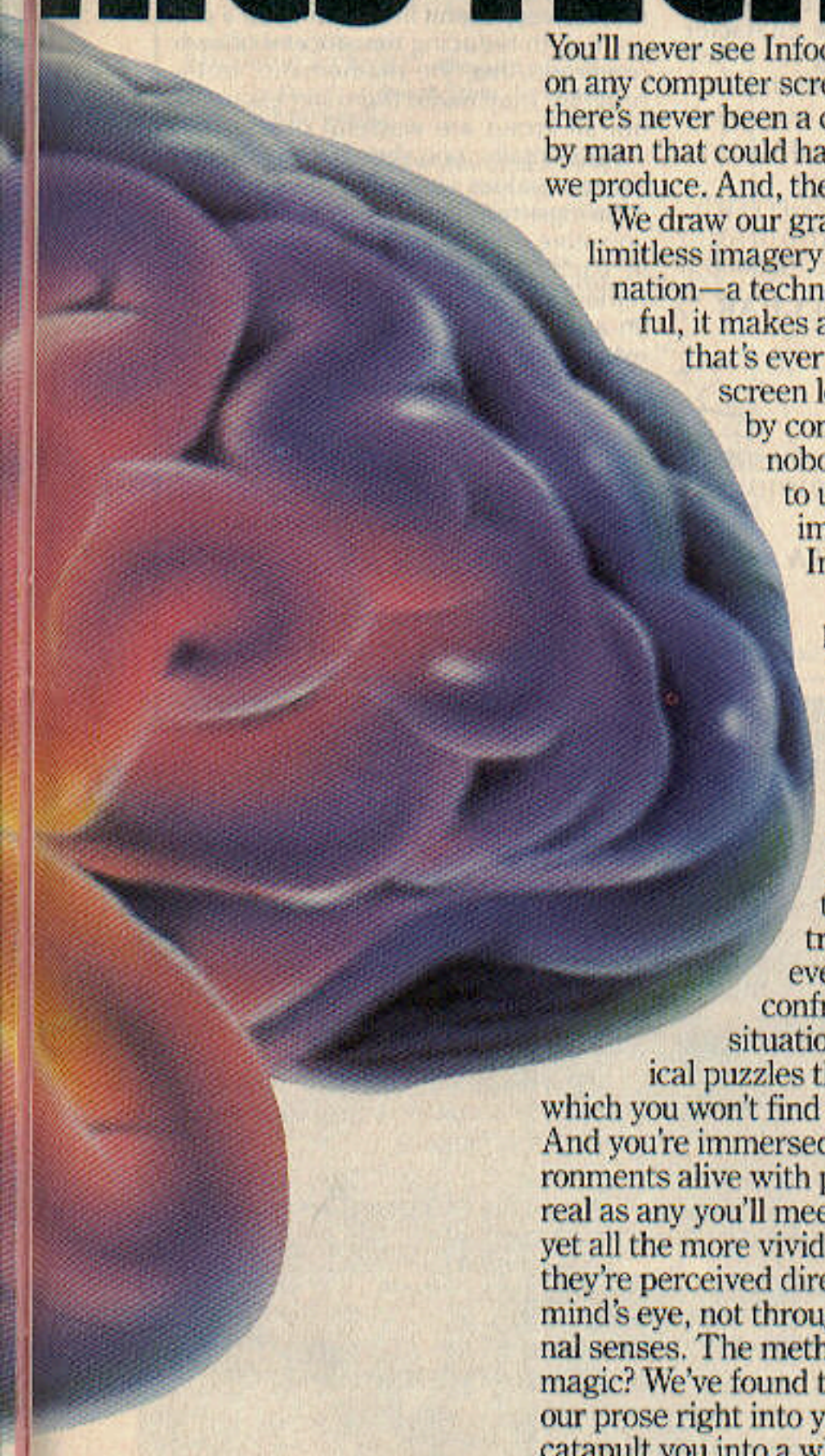
The seven new TI-marketed titles surfacing prior to Christmas include *Demon Attack* (Imagic), *Jawbreaker II* (Sierra-On-Line), *Star Trek* (Sega), *MoonMine* (TI), *FaceMaker* (Spinnaker), *Burger Time* (Data East), and *Hopper* (TI). Sales of the software cartridges are expected to be brisk this holiday season.

99'er Digest is a marketing information service for retailers, distributors, third-party vendors, sales representatives, industry analysts, and other TI-watchers interested in the home computing, personal computing, and portable computing markets in which Texas Instruments is present. The publication is issued biweekly and mailed First Class. Appropriate items of consumer interest are excerpted from the Digest in the monthly 99'er Home Computer Magazine. For subscription details contact: Emerald Valley Publishing Co., 1500 Valley River Drive, Suite 250, Eugene, OR 97401.

WE UNLEASH TH
POWERFUL GRAP



E WORLD'S MOST
HICS TECHNOLOGY.



You'll never see Infocom's graphics on any computer screen. Because there's never been a computer built by man that could handle the images we produce. And, there never will be.

We draw our graphics from the limitless imagery of your imagination—a technology so powerful, it makes any picture that's ever come out of a screen look like graffiti by comparison. And nobody knows how to unleash your imagination like Infocom.

Through our prose, your imagination makes you part of our stories, in control of what you do and where you go—yet unable to predict or control the course of events. You're confronted with situations and logical puzzles the like of which you won't find elsewhere. And you're immersed in rich environments alive with personalities as real as any you'll meet in the flesh—yet all the more vivid because they're perceived directly by your mind's eye, not through your external senses. The method to this magic? We've found the way to plug our prose right into your psyche, and catapult you into a whole new dimension.

Take some tough critics' words about our words. SOFTALK, for example, called ZORK® III's prose "far more graphic than any depiction yet achieved by an adventure with graphics." And the NEW YORK TIMES saw fit to print that our DEADLINE™ is "an amazing feat of programming." Even a journal as video-oriented as ELECTRONIC GAMES found Infocom prose to be such an eye-opener, they named one of our games their Best Adventure of 1983.

Better still, bring an Infocom game home with you. Discover firsthand why thousands upon thousands of discriminating game players keep turning everything we write into instantaneous bestsellers.

Step up to Infocom. All words. No graffiti. The secret reaches of your mind are beckoning. A whole new dimension is in there waiting for you.

INFOCOM™

The next dimension.

Infocom, Inc., 55 Wheeler St., Cambridge, MA 02138

For your: Apple II, Atari, Commodore 64, CP/M 8", DEC Rainbow, DEC RT-11, IBM, MS-DOS 2.0, NEC APC, NEC PC-8000, Osborne, TI Professional, TI 99/4A, TRS-80 Model I, TRS-80 Model III

GROUP GRAPEVINE

Group Grapevine: News of TI Users Groups from Around the World.

It's an emergency—the designer of TI-Horned Toad is in town and he's agreed to speak at your users group meeting this week! Such an event warrants a special announcement. You'd hate to have him show up to an empty Moose Hall. As you ponder whether to spend an evening on the phone or further deplete your post card budget, you grumble to yourself, "We can put a woman in space, but we still have to rely on archaic, time-consuming modes of communication." Those of you with on-line bulletin boards know there is a better way. These systems are a low-cost, efficient way to communicate with members (at least members with modem-equipped computers) in between meetings.

Now word has come from Atlanta that **TIBBS**, the Texas Instruments Bulletin Board System, is on-line and running 24 hours a day. Ralph Fowler, its creator, notes that this is the first free database of its kind to run on a TI-99/4A Home Computer. TIBBS offers the following services: dissemination of news concerning the TI-99/4A, a means for exchanging ideas and views on TI personal computing, and news from various users groups. In order to use TIBBS, your computer must be equipped with a modem, Terminal Emulator Cartridge and RS232 interface. Just phone (404)425-5254 any time day or night. If you have any questions about the development of TIBBS, write to Mr. Fowler at P.O. Box 383, Kennesaw, GA 30144.

If our mail is any indication, the TIBBS lines will be humming in double time. More and more new users groups have joined our growing ranks. Volume 1, Number 1 of the **Shoals 99'ers** newsletter arrived in the mail this week. An exceptionally nice first effort, the 8-page letter included a short chronological history of the TI-99/4A's prices (depressing reading for some of you old timers), a 2-page glossary of computer terms, two short game reviews, and reports from the group's various committees. Sounds like this is a group to watch. Contact them c/o Shoals 99'ers, P.O. Box 2928, Muscle Shoals, AL 35662.

Another new group—new to us anyway—is the **Lehigh 99'er Computer Group**. The group was formed in January of this year and now includes over 60 members, a number of whom are involved in the group's Education, Library, Newsletter, and Program committees. The Education committee holds monthly lectures on programming techniques, and the Program committee is responsible for displaying running programs at each meeting. The group can be reached at P.O.Box 4837, 1501 Lehigh St., Allentown, PA 18103.

A brand new spin-off user group is forming in Vancouver, WA. Weary of commuting to **PUNN** (Portland Users of the Ninety Nine) meetings, R.S.(Bob) Chase is putting out the call to all TI Users in the Vancouver/Clark County area. Here is your chance to get in on the ground floor and become a charter member. For information, contact Bob Chase at (206) 695-7002.

New spin-off or old veteran, we'd like to hear from you. Send news of your group's activities to the **Users Group Editor, 99'er Home Computer Magazine, 1500 Valley River Drive, Suite 250, Eugene, OR 97401.**

Squeeze. . . from p. 26

One of the best tools for saving memory is the concept of looping. Notice that programmers frequently write large programs with many sections having nearly identical code. A program which defines 5 characters could call the routine CHAR 5 times (see Listing 1a). Looping and using READ with DATA statements, (as shown in Listing 1b) saves memory. This really makes sense if your program defines many characters rather than just 5, as in Listing 1a. If the characters to be defined are contiguous, the loop index could be the character number, as shown in Listing 1c. In my program I define or redefine 87 characters, so this technique helps a lot.

```
10 CALL CHAR (10,"FE0C45F8902")
20 CALL CHAR (13,"0030F")
30 CALL CHAR (45,"FFFFF00FFFFF")
40 CALL CHAR (69,"1010101010101")
50 CALL CHAR (144,"010204080408")
```

Listing 1a: Multiple calls to CHAR

```
10 FOR I=1 TO 10
20 READ T,T$
30 CALL CHAR(T,T$)
40 NEXT I
50 DATA 10,FE0C45F8902,13,0030F,45,
   FFFFF00FFFFF,69,1010101010101,
   144,0102040810204O8
```

Listing 1b: Combining DATA and one call to CHAR

```
10 FOR T=100 TO 104
20 READ T$
30 CALL CHAR(T,T$)
40 NEXT T
50 DATA FE0C45F8902,0030F,FFFFF00F
   FFFF,1010101010101,0102040810
   20408
```

Listing 1c: Defining contiguous characters with DATA

You can use DATA statements to assign values to variables. This especially helps during initialization, when many variables have to be set. Compare Listings 2a and 2b. (Note: You may, of course, not need the RESTORE statement, depending on the sequence of READs in your program.)

```
10 A=1.6
20 B=8.2
30 C=0
40 D=0
50 E=-36
```

Listing 2a: Multiple assignment statements

```
10 RESTORE 30
20 READ A,B,C,D,E
30 DATA 1.6,8.2,0,0,-36
```

Listing 2b: Using READ and DATA instead of multiple assignment statements

Remark statements can help significantly to make a program more readable. However, they do occupy memory space. So you'll need to make them brief (and hope you won't have to omit them altogether). Keep as many as possible, and certainly keep one at the front of the program to identify the program, its version or date, and the disk file or cassette where it is stored.

The method you choose to perform computations can have a significant effect on program size. Sometimes people writing programs which process data (such as statistical programs) have the program first read all the data into arrays before any of it is processed. Frequently a rearrangement of the formulas yields a method of computing while reading, thereby eliminating entirely the need to keep any data in arrays. But at other times, as I have described above, arrays may be very useful in speeding up a program or in reducing the amount of code required. Also, the mathematics of the program may make them necessary, so not all arrays are wasteful of space.

Incidentally, you should know that an array of values (as opposed to an array of strings) requires 8 bytes per value. If you store only a small number of values in each cell, you waste memory. To remember 100 single-digit values with an array would require 800 bytes. You might be able to store the same data in a string 100 bytes long; this would save 700 bytes.

Using a string to store numeric data is especially convenient if the values do not change during execution. In that case, you merely assign the string to a string variable or use it directly. Examples:

```
S$="393211784"
X=ASC(SEG$(S$,I,1))-48
or
X=ASC(SEG$("393211784",I,1))-48
```

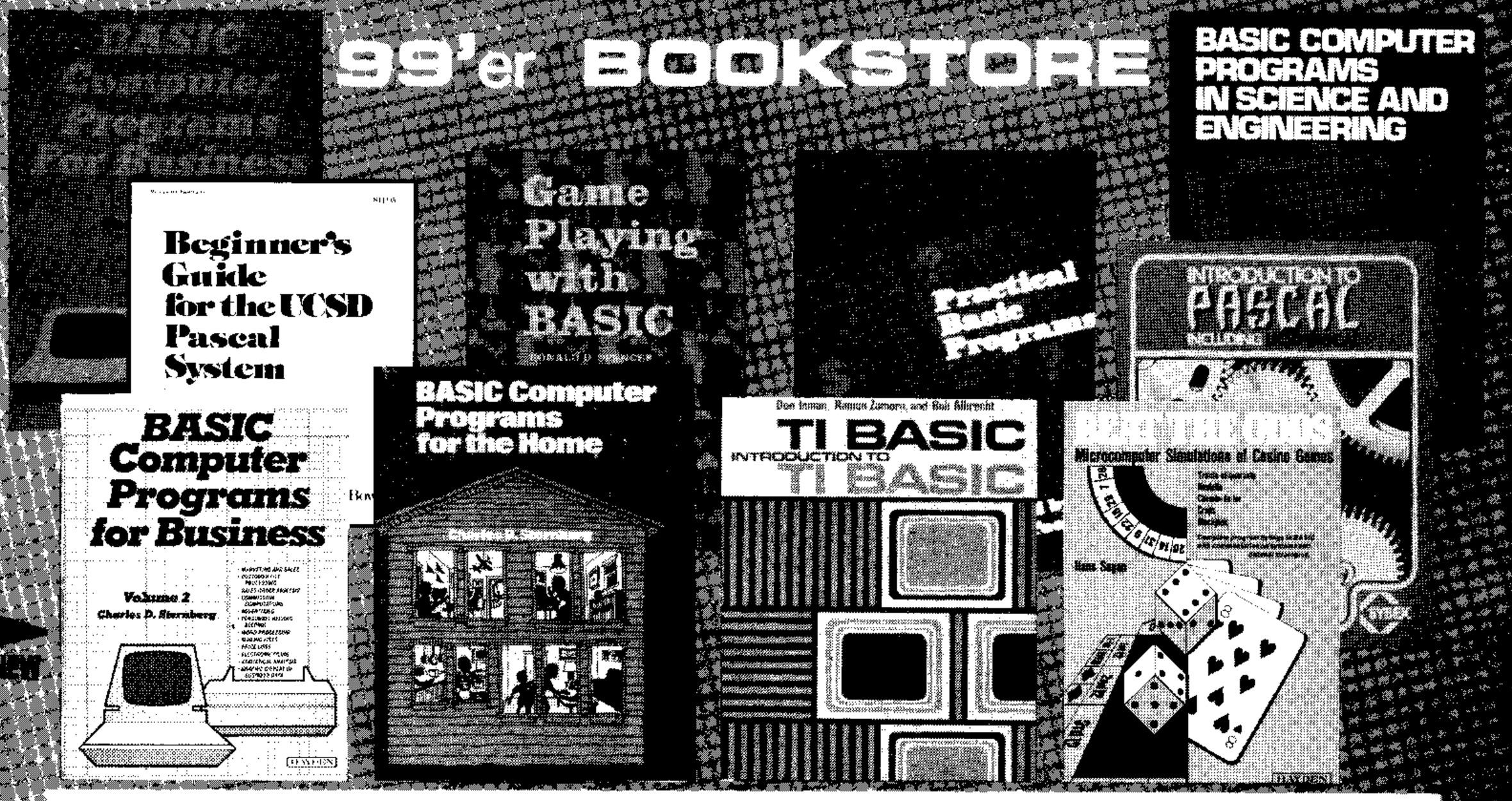
Either case sets X equal to the value of the digit I in the string. The first case would be appropriate if the string were referenced several times in the program; assigning the string to S$ would occur only once in the program, of course. Values greater than single digits can be stored as letters or punctuation. (This would make your program hard to understand without proper documentation.) Also, you can store any value from 0 to 255 if you build the string using CHR$. This example will remember as many as 255 numbers that you type in, each in the range 0 to 255:

```
10 S$=""
20 INPUT "ENTER NUMBER (-1 TO
   STOP):":N
30 IF N<0 THEN 60
40 S$=S$&CHR$(N)
50 GOTO 20
60 FOR I=1 TO LEN(S$)
70 PRINT "ITEM NUMBER:";I;"
   VALUE:";ASC(SEG$(S$,I,1))
80 NEXT I
```

In addition to the memory savings that can result from overall program design, exactly *how* you write computational statements can save you a little memory each time. Frequently you can replace a logical structure with a computation, although this can easily result in obscure coding.

Continued on p. 73

99'er BOOKSTORE



BASIC COMPUTER PROGRAMS FOR BUSINESS: VOL. 1

By Charles D. Sternberg.

Each program is documented with a description of its functions and operation, a listing in BASIC, a symbol table, sample data, and one or more samples.

Volume 1 contains over 35 programs covering: budgets, depreciation, cash flow, property comparisons, accounts payable, order entry, warehouse locations, inventory turnover analysis, job routine, resource allocation, production scheduling, etc.

paper, **$14.95**
1980, 384 pages, 7×10

BASIC COMPUTER PROGRAMS FOR BUSINESS: VOL. 2

By Charles D. Sternberg.

A collection of programs (with function and operation documentation) covering the critical facets of business operation: sales planning and analysis, personnel, word processing, mailing lists, price list systems, record control, scheduling systems, etc.

paper, **$15.95**
1982, 376 pages, 7×10

BEGINNER'S GUIDE FOR THE UCSD PASCAL SYSTEM

By Kenneth Bowles.

This highly informative book is written by the originator of the UCSD Pascal System. It is designed as an orientation guide for learning to use the UCSD Pascal System, and features tutorial examples of programming tasks in the form of self-study quiz programs. Once familiar with the system you will find the guide an invaluable reference tool for creating advanced applications.

paper, **$14.95**
1980, 204 pages, 6×9

BASIC COMPUTER PROGRAMS FOR THE HOME

By Charles D. Sternberg.

An invaluable book containing over 75 practical home application programs that will be helpful to the novice or experienced owner in increasing the usefulness of any home computer. Each program is documented with a description of its functions and operation, a listing in BASIC, a symbol table, sample data, and one or more samples.

paper, **$13.95**
1979, 336 pages, 7×10, illus.

GAME PLAYING WITH BASIC

By Donald D. Spencer.

Enjoy the challenge of competition with your computer. Amuse yourself with such games and puzzles as 3-D Tic-tac-toe, Nim, Roulette, Magic Squares, the 15 Puzzle, Baccarat, Knight's Magic Tour, and many others. The writing is nontechnical, allowing almost anyone to understand computerized game playing.

paper, **$13.95**
1977, 176 pages, 6×9, illus.

BASIC COMPUTER PROGRAMS IN SCIENCE AND ENGINEERING

By Jules. H. Gilder.

Save time and money with this collection of 114 ready-to-run BASIC programs for the hobbyist and engineer. There are programs to do such statistical operations as means, standard deviation averages, curve-fitting, and interpolation. There are programs that design antennas, filters, attenuators, matching networks, plotting, and histogram programs.

paper, **$13.95**
1980, 160 pages, 6×9, illus.

PRACTICAL BASIC PROGRAMS

Edited by Lon Poole

Here is a new collection of 40 programs you can easily key in and use on most microcomputers. Each program does something useful. Practical BASIC Programs is especially useful in small business applications. It solves problems in finance, management decision, mathematics and statistics. It requires no prior programming knowledge. Each program is thoroughly documented. The book contains sample runs, practical problems, BASIC source listings, and an easy to follow narrative to help you realize the potential uses of each program.

paper, **$17.95**
1980, 200 pages, 8½×11

INTRODUCTION TO TI BASIC

By D. Inman, R. Zamora, and R. Albrecht.

This comprehensive work will teach you all about computer and BASIC for use with the Texas Instruments Home Computer. Even if you've never worked with a computer, you can now teach yourself how to use, program and enjoy the TI Home Computer with this entertaining, and easy-to-read work. The authors have carefully constructed this introduction so that you will soon be writing BASIC programs and exploiting all of the excellent features of the TI machines. Its 14 chapters and Appendices cover all of the essential programming statements and machine features.

paper, **$14.95**
1980, 384 pager, 7×10

INTRODUCTION TO PASCAL (INCLUDING UCSD PASCAL)

By Rodnay Zaks

This is the first book on Pascal that can be used by persons who have never programmed before, but more generally it is a simple and comprehensive introduction to standard and UCSD Pascal for anyone—beginner to experienced programmer—who wants to learn the language rapidly. The logical progression and graduated exercises—designed to provide practice as well as test skill and comprehension—enable the reader to begin writing simple programs almost immediately.

paper, **$17.95**
1981, 440 pages, 7×9

BEAT THE ODDS: MICRO-COMPUTER SIMULATIONS OF CASINO GAMES

By Hans Sagan.

Here's an extremely useful programming guide that provides realistic simulations of five popular Casino games: Trente-et-Quarante (Thirty and Forty), Roulette, Chemin-de-Fer, Craps, and Blackjack. Each of the five chapters has the same structure. It begins with a computer run, displaying facets of the programs, followed by an explanation of the objectives and the physical execution of the game. Acceptable bets and how to place them are discussed and systems and/or strategies laid out. Finally, the computer program is developed and various modifications of the program are detailed.

paper, **$11.95**
1980, 128 pages, 6×9

Use the order card in the back of this magazine, or itemize your order on a separate piece of paper and mail to: 99'er HCM/Book Dept., P.O. Box 5537, Eugene, Oregon 97405. Be sure to include check or detailed credit card information. Shipping and Handling: In U.S.A.—$2.50 for one book; 75¢ for each additional book. Foreign Surface—add $3.00 to total U.S.A. shipping costs. Please allow 4-6 weeks for delivery. If there is a question regarding your order please write to Customer Service at the above address.
PRICES SUBJECT TO CHANGE WITHOUT NOTICE.

America ... from p. 53

```
1070 CALL HCHAR(6,5,96,2)::: CALL HCHAR(6,4,103)::: CALL HCHAR(7,4,101)::: CALL HCHAR(8,4,102)
1080 CALL HCHAR(7,5,96,2)::: CALL HCHAR(8,5,96,2)::: RETURN
1090 CALL HCHAR(7,26,136,3)::: CALL HCHAR(8,26,136,3)::: RETURN
1100 CALL HCHAR(7,31,97)::: RETURN
1110 CALL HCHAR(15,27,40,2)::: CALL HCHAR(16,27,40)::: CALL HCHAR(16,28,47)::: RETURN
1120 CALL HCHAR(6,14,136,4)::: CALL HCHAR(7,14,136,4)::: RETURN
1130 CALL HCHAR(13,21,136,5)::: CALL HCHAR(14,21,136,5)::: RETURN
1140 CALL HCHAR(14,14,97,2)::: CALL HCHAR(15,14,97,2)::: CALL HCHAR(16,12,97,7)::: CALL HCHAR(17,12,135)
1150 CALL HCHAR(18,12,99)::: CALL HCHAR(18,13,100)::: CALL HCHAR(18,14,128)::: CALL HCHAR(17,13,97,6)
1160 CALL HCHAR(19,15,129)::: CALL HCHAR(20,15,130)::: CALL HCHAR(20,16,131)::: CALL HCHAR(19,16,132)
1170 CALL HCHAR(18,15,97,2)::: CALL HCHAR(18,17,133)::: CALL HCHAR(18,18,134)::: RETURN
1180 CALL VCHAR(9,8,136,4)::: CALL VCHAR(9,9,136,4)::: CALL VCHAR(10,10,136,3)::: RETURN
1190 CALL HCHAR(5,29,136)::: RETURN
1200 CALL HCHAR(10,28,122)::: CALL HCHAR(11,25,122,5)::: CALL HCHAR(12,25,122,5)::: RETURN
1210 CALL HCHAR(4,6,112)::: CALL HCHAR(5,5,112,2)::: CALL HCHAR(4,5,115)::: RETURN
1220 CALL HCHAR(9,26,33)::: CALL HCHAR(10,26,33,2)::: RETURN
1230 CALL VCHAR(5,20,33,3)::: CALL VCHAR(5,21,33,3)::: RETURN
1240 CALL HCHAR(7,10,112,4)::: CALL HCHAR(8,10,112,4)::: CALL HCHAR(9,10,112,4)::: RETURN
1250 DISPLAY AT(22,1)BEEP:"WRONG NAME" :: FOR KK=1 TO 400 :: NEXT KK
1260 DISPLAY AT(1,1):"STATE TRIES=";ATS;TAB(17);"RIGHT=";RS :: GOTO 1300
1270 DISPLAY AT(22,1)BEEP:"CAPITAL? Y OR N"
1280 CALL KEY(0,A,S):: IF S=0 OR (A<>89 AND A<>78)THEN 1280 ELSE IF A=78 THEN 370
1290 ACCEPT AT(23,1)BEEP:CN$ :: AC=AC+1 :: IF CAS(W)<>CN$ THEN GOTO 1250 ELSE CC=CC+1
1300 DISPLAY AT(24,1):"CAP TRIES=";AC;TAB(17);"RIGHT=";CC :: IF RS=50 THEN 1300
1310 IF P=0 THEN 370
1320 IF CAS(W)<>CN$ THEN 1270
1330 IF RS=50 THEN 1350 ELSE GOTO 370
1340 DISPLAY AT(22,1)BEEP:"ALREADY USED ";NS$ :: FOR KK=1 TO 450 :: NEXT KK :: GOTO 370
1350 CALL CLEAR :: CALL DELSPRITE(ALL)
1360 CALL COLOR(10,16,1,13,16,5,14,7,1)
1370 CALL CHAR(128,"10103838FE383844482",104,"FFFFFFFFFFFFFFFF",136,"FFFFFFFFFFFFFFFF"):: CALL SCREEN(6)
1380 FOR R=6 TO 12 STEP 2 :: CALL HCHAR(R,12,136,15):: NEXT R
1390 FOR R=7 TO 11 STEP 2 :: CALL HCHAR(R,12,104,15):: NEXT R
1400 FOR R=14 TO 18 STEP 2 :: CALL HCHAR(R,6,136,21):: NEXT R
1410 FOR R=13 TO 17 STEP 2 :: CALL HCHAR(R,6,104,21):: NEXT R
1420 FOR R=6 TO 12 :: CALL HCHAR(R,6,128,7):: NEXT R
1430 CALL SOUND(600,349,0)::: CALL SOUND(500,330,0)::: CALL SOUND(400,294,0)::: CALL SOUND(400,330,0)
1440 CALL SOUND(350,294,0)::: CALL SOUND(500,262,0)::: GOTO 1260
1450 DATA ALABAMA,ALASKA,ARIZONA,ARKANSAS,CALIFORNIA,COLORADO,CONNECTICUT,DELAWARE,FLORIDA,GEORGIA,HAWAII,IDAHO
1460 DATA ILLINOIS,INDIANA,IOWA,KANSAS,KENTUCKY,LOUISIANA,MAINE,MARYLAND,MASSACHUSETTS,MICHIGAN,MINNESOTA
1470 DATA MISSISSIPPI,MISSOURI,MONTANA,NEBRASKA,NEVADA,NEW HAMPSHIRE,NEW JERSEY,NEW MEXICO,NEW YORK
1480 DATA NORTH CAROLINA,NORTH DAKOTA,OHIO,OKLAHOMA,OREGON,PENNSYLVANIA,RHODE ISLAND,SOUTH CAROLINA,SOUTH DAKOTA
1490 DATA TENNESSEE,TEXAS,UTAH,VERMONT,VIRGINIA,WASHINGTON,WEST VIRGINIA,WISCONSIN,WYOMING
1500 DATA MONTGOMERY,JUNEAU,PHOENIX,LITTLE ROCK,SACRAMENTO,DENVER,HARTFORD,DOVER,TALLAHASSEE,ATLANTA,HONOLULU,BOISE
1510 DATA SPRINGFIELD,INDIANAPOLIS,DES MOINES,TOPEKA,FRANKFORT,BATON ROUGE,AUGUSTA,ANNAPOLIS,BOSTON,LANSING,ST PAUL
1520 DATA JACKSON,JEFFERSON CITY,HELENA,LINCOLN,CARSON CITY,CONCORD,TRENTON,SANTA FE,ALBANY
1530 DATA RALEIGH,BISMARCK,COLUMBUS,OKLAHOMA CITY,SALEM,HARRISBURG,PROVIDENCE,COLUMBIA,PIERRE
1540 DATA NASHVILLE,AUSTIN,SALT LAKE CITY,MONTPELIER,RICHMOND,OLYMPIA,CHARLESTON,MADISON,CHEYENNE
1550 END
```

What's In A Word?

Using the Basic Reading Skills Survey

by Sharyn Lyon
Education Editor

Remember when being with your best friend was the most important thing in the world? Well, that's how it was for Myia Malone and her friend Jill. They'd been in the same class ever since kindergarten, had even gone on vacations together! Now that fourth grade was in full swing, however, they were not together in their reading groups, and both girls were devastated. They were, that is, until Myia's parents discovered PLATO and how the *Basic Skills Reading* program could help them help their daughter get back into Jill's group again.

"PLATO's learning atmosphere is relaxed. . . and the natural tension that sometimes prevents parents from being effective teachers for their own children is reduced."

At the Parent/Teacher Conference, Myia's teacher told Mr. and Mrs. Malone about the PLATO system they were using at the school and asked them if they would like to come in a few times a week and use it with Myia. At first they were hesitant because, like many people of their generation, the computer made them nervous. When, however, the teacher showed them how easy the system was to use—the automatic load feature particularly impressed them—two more computer-assisted instruction fans were born.

It wasn't long after their introduction to PLATO's CAI system that the Malones were not only coming in to work with their daughter in class, but also started checking the system out to use at home on the weekends!

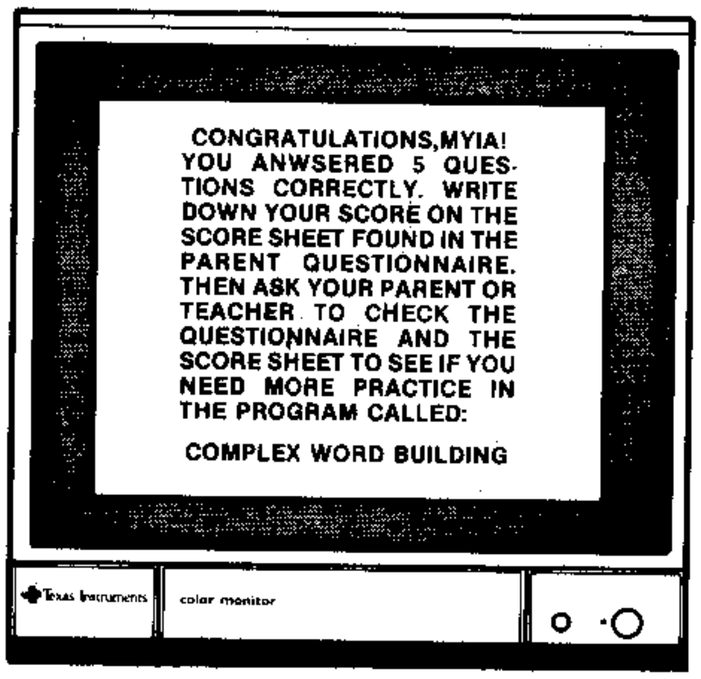
The Malones began their program by using the *Basic Skills Reading Survey* system to help them determine what concepts were holding Myia back. After spending only about ten minutes using the Keyboard Tutorial on the diskette to familiarize herself with the keyboard, she typed in her name and was

*PLATO is a registered trademark of Control Data Corporation.

presented with two subject areas from which to choose: 1) *Making New Words* and 2) *Understanding New Words*.

Encouraging Words

These two categories encompassed adding -ed, -s, and -ing; root words; compound words; contractions and abbreviations; and using context to decipher the meaning of a new word. Myia came through all these survey quizzes with flying colors, and at the end of each section she completed she kept seeing a screen that looked like this:



All of the rewards in this package are textual but easy to read, motivating and varied. "Congratulations" is used only at the end of the survey quizzes, but the encouraging words, SUPER GOOD WORK, EXCELLENT, TERRIFIC, and GOOD JOB were used throughout to keep Myia feeling successful and anxious to go on.

Where Do I Begin?

While Myia was working with words and word attack skills on the survey diskette, her parents were filling out the Parent Questionnaire. Within the Reading curriculum area there are 22 packages, and for each package there are from 3 to 5 questions for the parents to answer pertaining to what they think their child really can do. Although that means a total of over one hundred questions, they are easy Yes or No questions relating to the child's ability in specific skills and take very little time to answer. The following question, for example, asks the parents whether they think their child has an understanding of figurative expressions:

"Can your child tell you what the underlined word means in the following sentence: 'My car is a lemon!'?"

The Malones found that if they used initials (i.e.,Y for Yes) to record their answers, they could re-use the forms for Myia should she want to do that part of the survey again, or for another child (see Fig.1).

On The Other Hand

From the Survey and Parent Questionnaire results it was apparent that Myia had a firm grasp of words—their structures, functions, and meanings. Questions like this one from the first side of the Basic Skills Reading diskette gave her no trouble at all:

CHOOSE THE WORD THAT COMPLETES THE SENTENCE.
SHE HAS GOOD _____.

A. WILDLIFE B. WOODPILE
C. PIPELINE D. SALESPERSON
E. EYESIGHT

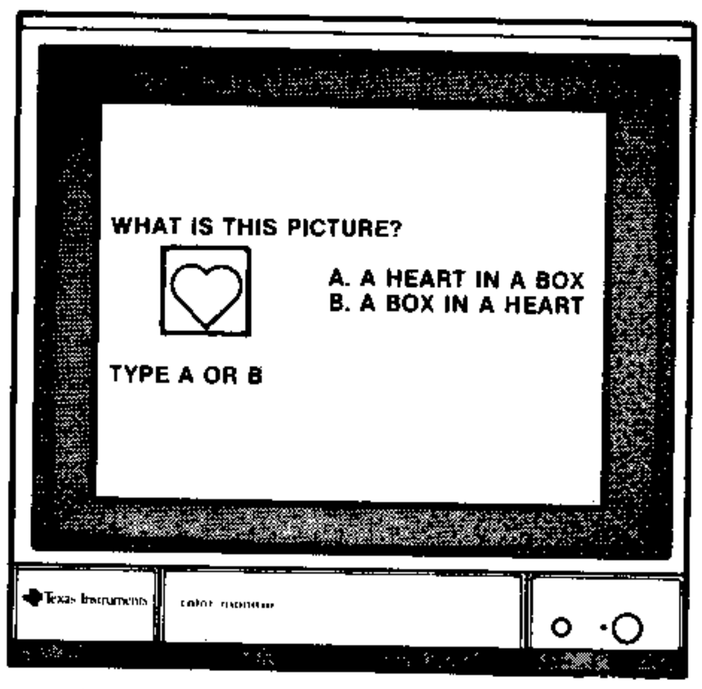
The next step was to turn over the diskette to work on other reading skills. The menu for the second side displays the following subject areas which focus mainly on comprehension and interpretation skills:

1. Understanding What You Read
2. Thinking About What You Read
3. Judging What You Read

Within each of these major subject areas are an average of 6 categories; for example, under the first subject area there are these categories:

1. Locating Basic Facts
2. More Basic Facts from Reading
3. Understanding What You Read
4. Remembering What You Read
5. Remembering More Of What You Read
6. Interpreting What You Read

The first question from the second category, *More Basic Facts from Reading,* was really the first one Myia had to stop and think about:



After a few seconds of consideration she did get this right as well as the other questions in the section dealing with tenses, sequence and concept visualization. When she tried category three, however, Myia met her Waterloo. The focus here is on referents, sequencing story events, and identifying facts that don't pertain to the subject matter of a given paragraph.

Myia repeatedly missed questions of the latter variety, which really puzzled her parents and teacher. One theory was that because Myia was entering her answers in lower case, the program was not recognizing them. But since PLATO is not fussy and accepts both upper and lower case, that wasn't the reason for her low scores.

Her teacher then tested Myia on identifying pertinent details using sections from the basal reader they were using in class and got the same results—Myia was not always able to pick out details relevant to the subject of a story segment. When her father tested her from books she had checked out from the library, she did not, strangely enough, have any trouble sorting out facts that didn't belong. At first this discrepancy was perplexing, but after a little more work with the PLATO surveys and some more supplementary testing using books Myia picked out for herself, her teacher determined that 1) Myia was trying to skim through material that did not directly interest her, and 2) she was unclear as to the difference between the author's intent and her own reasons for reading.

"Using PLATO's carefully wrought system, students move logically and gradually through a learning process custom-designed for each one of them."

Once her parents and teacher had assessed what was holding Myia back, they made a concerted effort to teach her how to scan instead of skim, and helped her learn how to see the author's purpose in what she read using the *Author's Purpose and Your Conclusions* program package from the PLATO courseware.

PLATOnic Pitfalls

In the process of going through the surveys, the Malones identified a few areas to which parents and teachers need to pay particular attention. Their biggest concern seemed to be that some questions lend themselves to guessing. If a student relies on guessing, she can be either lucky or unlucky in her score and could be assessed at either too high or too low a level. Even though guessing can work to some students' advantage, whoever is guiding students through the tests should caution them against guessing because of the way these tests are constructed. PLATO's survey is spirally structured so that the same type of question gets asked over again in different ways. A guessable question like

CHOOSE THE WORD THAT MEANS THE SAME AS THIS WORD. **IT'S**

A. I AM B. YOU ARE
C. HE IS D. IT IS
E. SHE IS

will be asked in a less transparent way later on in the test. So it's important that parents and teachers discourage guessing and encourage students to complete the *entire* survey test no matter what it looks like their score is going to be.

Continued on p. 66

INTERVIEW

DALE OSBORN

On a recent visit to Texas, 99'er HCM talked with TI's Software Operations Manager, Dale Osborn, who provided some insight into the educational software development industry. Mr. Osborn has been with TI's Consumer Group for three years and has been with the four-months-old Home Computer Software Division since its inception. He has been associated with Texas Instruments' new products since 1973.

Manager of Software Operations for the Home Computer Division of the Consumer Group, Texas Instruments.

"The computer and the software are tools to be used in everyday lives, not just as amusement items."

99'er: Isn't it true that most people buy Home Computers because of the entertainment possibilities of games like *Parsec* and *Munch Man*?

DO: Our market research tells us that over the last several months, the rate of educational software purchasing is growing more rapidly than that of entertainment software. We hear the Home Computer owner saying, "I can use this machine for more than just playing games!" So they're turning to the educational applications. We have an excellent line-up of educational software, and have third-party publishers like Milliken; Scott, Foresman; and DLM [Development Learning Materials, Inc.] writing outstanding educational software for our machine.

99'er: Will the products being developed by TI and third-party publishers enable the Home Computer to substitute for the classroom process itself?

DO: We strongly believe that the computer is an aid, a tool, in support of a teacher (or a parent for that matter) in helping children to advance their education. We don't believe that the computer will replace the teacher. We think that we need to educate the teachers as to how well the computer can serve them in productivity improvements in the classroom.

99'er: So you see parents, teachers and computers as a teaching team then?

DO: Yes. What we are doing is working with quality third-party educators and authors to develop software that teaches the *experience* of learning, rather than just drill-and-practice. In the case of an animated astronomy package, for example, children can go through the experience and the theory of astronomy, and at the same time, the software teaches them about the solar system, the planets, and the stars. We think that the same type of experience could take place in the schools. The computer can aid the teacher in the theoretical process rather than in just helping students learn by drill-and-practice. We can see that the computer enhances the *total* teaching process. TI is therefore stepping into that area as well. The PLATO software, of course, does that in an excellent fashion.

99'er: TI has been making education learning tools and teaching aids for years now, but the Home Computer seems to be a more flexible tool than your individually dedicated products like Speak-And-Spells and Little Professors are. Does the Home Computer actually add another dimension to the learning process that these dedicated devices can't?

DO: The Home Computer of course, does bring a new dimension to the educational process. TI's software packages can teach the concept, the theory, of the learning process, and can show you graphically

why one plus one is two. Theoretical instruction is what the computer brings to education rather than just the rote drill-and-practice.

99'er: We're wondering if the 99/4A—an "old timer" by present market standard—is still suitable as an educational tool? Has its full potential been reached already?

DO: Well, the machine itself is a very capable unit, and the answer to the question really lies in the availability of the educational software. The capabilities of the machine can be further enhanced by more highly developed software, rather than by changing the structure of the machine. We think the 99/4A itself is very competitive in its current configuration, and our emphasis will be to provide more powerful software to show the capabilities of the machine better.

99'er: OK then, this seems to suggest, in looking at TI's new product announcements, that "education" means *cartridges*. Is this the way it's going?

DO: Currently, the majority of software that we have in education is targeted for the under-twelve-years-old market. We feel that in the younger age categories, the ease of use of the machine and software is *critical*. Consequently, we will continue to develop educational programs for the elementary-age user in cartridge format. As disk capabilities become more routine in the home, I believe that seven- and eight-year-olds can begin using a soft media disk or cassette storage form and will be able to operate our machine quite handily. We've actually seen some eight-year-old youngsters who are already working very impressively and capably in this area. Software like PLATO's High School Skills and Basic Skills packages are already being developed on disk. We do believe disk is the wave of the future as far as bringing low cost educational software into the home is concerned. We will not abandon cartridges, however, because they are so easy for young computer users to work with.

99'er: Hearing you speak about young users, we wonder about the TI machine's speech capability. How important do you think speech is? Is it a legitimate tool in the learning process for children, or is it, as some critics have labeled it—just a gimmick, to sell machines?

DO: Actually, we do think that speech adds a true extra dimension to the capability of the system. A young (even preschool-aged) child can interact with the machine and learn from it, even if he hasn't learned to read yet.

99'er: A few minutes ago, you mentioned that as disk usage spreads, it will give Texas Instruments the ability to deliver more low-cost educational software. You pointed out that an obvious example of this is the PLATO series, where the courses are now disk based. What's happening with the PLATO series, and what is the significance of it?

DO: PLATO is a software system (jointly developed for the 99/4A in a cooperative effort with Control Data Corporation) that actually contains two series. One is the Basic Skills and the other is High School Skills. Each of the series has a set of curricula which is made up of program packages. Each program package is a thorough look at a specific segment within that curricula. We might, for example, look at the Basic Skills Mathematics series, in which a first-level program package would be *Numbers 0 through 9*. Its purpose would be to help a young child learn the concept of numbers, recognize numbers, and understand what numbers do.

Each package in the PLATO format contains a larger volume of information and a more enhanced learning experience than several cartridge-based educational packages. PLATO is designed for the parents who want their child to have a head start into the educational process and

"We do believe disk is the wave of the future as far as bringing low cost educational software into the home is concerned."

who are concerned about the quality of education in the public school system. These parents are concerned about giving their child every possible advantage in the educational process. School systems want PLATO software because it will function very well as a remedial type process. They can see how a child having difficulty or one who wants to move to a higher group within his class can use PLATO software to enhance his ability and move on to a higher group. Really, the question parents have to ask themselves is "What's it worth to me as a parent to see my children move ahead?"

99'er: We've seen some of your recent announcements, about new products from DLM and Spinnaker, and about the new directions in educational software coming from Texas Instruments. Will these products be available soon to the Home Computer user?

DO: Yes, in fact this is something we're very excited about here. We're in the process now of completing "conversions," if you will, of two excellent products developed from Spinnaker's *Face Maker* and *Story Teller*. Both of these products have been sold primarily in the school systems and more affluent homes. We're also developing a series of software with DLM, related primarily to grammar, reading, and word usage. And DLM right now has a mathematically oriented series which includes titles like *Alligator Mix*.

Perhaps the most exciting new direction for us is our move into the theoretical aspect of the learning process, of how to actually *teach* a child. We've found that by using the Home Computer with our upcoming courseware, children can have a good time discovering how things work in areas of science such as astronomy and chemistry. In these fields drill-and-practice type methods would not really be teaching children much. Computer-assisted instruction in science-related fields is currently being developed with a more theoretical approach to the teaching process. You'll see *enhanced* educational software that actually teaches the concept of the subject as well as provides the appropriate practice. We call it a *Tutorial Concept*.

99'er: TI also seems to be exploring new ways of telling the public about its concepts in educational software—by participating in state fairs, amusement parks, and a variety of non-traditional marketing settings. Is all this exposure suitable for promoting the use of the Home Computer in the educational area, or is it more geared towards amusement and entertainment?

DO: Well, actually we think that since computers are rapidly starting to play a major role in virtually every person's life, it is absolutely appropriate that managers of the entertainment parks, the state fairs, and even shopping malls are picking up on this and promoting computer use in their environments. What we all are emphasizing is the use of the computer in our daily lives, and that it can be used in many areas—not just in the entertainment area. Specifically, we are doing two programs: one with Six Flags at Magic Mountain in California and another at Knott's Berry Farm in California. Both of those management teams have had the foresight to promote the computer and software as tools to be used in everyday lives, not just as amusement items.

The state fairs and the shopping malls are using the computers primarily to provide menus of what is occurring within their specific realms. For example, a state fair's computers tell you what is located where, and what can be done within the fair. In the shopping malls, computers are used to tell the public what is available for children of specific ages or to suggest a gift selection item for a special occasion. So computers and software are being promoted by these people, with obvious foresight, as an indispensable tool for management and customer service activities.

PLATO. . . from p. 63

Pictures Or A Thousand Words

Some students need more encouragement than others to keep going in any learning program. The more the student has played computer games, especially educational games, such as *Alligator Mix*, the harder it may be for them to stay enthralled with the purely textual presentations of the PLATO survey tests. It may be that sixteen-color graphics and complex sound and music would only serve to slow down and obscure the purpose of the survey tests. But it could be argued that a few more pictorial rewards or illustrations accompanying questions (like the heart in the box cited above) would help keep students moving through the surveys. Students who are doing well on the tests do have their scores to encourage them (as did Myia on side one of the diskette), but when they start missing a few questions and begin to get discouraged, it might help to have PLATO make more fuss over correct answers with an eye catching graphic.

PLATO's Pluses

If a caring adult is close at hand, Plato's lack of graphics and elaborate encouragement may not be problems at all. An adult devoted to the child and her learning progress will see to it that she moves properly through the surveys and feels adequately rewarded and encouraged along the way. A caring adult might have a harder time, though, providing a poorly organized program with an educationally sound plan of action. Fortunately, with Plato this isn't necessary. The program is superbly organized to present concepts in an educationally perceptive way.

The years of research that Control Data has devoted (and continues to devote) to PLATO show up in the complete and clearly written documentation and in the way that concepts are presented to the student. Back in the Fall quarter of 1981, Control Data's Evaluation Manager instituted a thorough peer review system for even finer quality control. Both the *Basic Skills* and *High School Skills* packages were reviewed by teacher representatives from professional associations like the National Council of Teachers of English. These reviewers were just as impressed as we are with the educational principles upon which PLATO is based. Using PLATO's carefully wrought system, students move logically and gradually through a learning process that seems custom-designed for each one of them.

From Assessment To Achievenent

Like the *Math Skills Survey* package, the *Reading Skills Survey* assesses student abilities thoroughly, using a combination of parent/student inputs. From the moment Myia picked a subject area from the program package menu, she was taking on some of the responsibility for her own learning. After she completed the test, recorded her score on the Parent Questionnaire, and compared her results with what her parents thought they would be, she and her parents became partners in her learning process. Now they could decide together which packages Myia should work on. Sharing the learning and teaching responsibilities like this allows both the parent and the child to be relieved of some of the pressure to succeed. PLATO's learning atmosphere is relaxed, and the natural tension that sometimes prevents parents from being effective teachers for their own children is reduced.

The spiral staircase configuration of the survey's test questions also represents a sound educational progression that's used throughout the PLATO courseware. The method not only assesses the student's abilities and presents the material, but instructs, provides practice, and periodically reviews the con-

Continued on p. 68

WE CHALLENGE YOU TO COMPARE —

☞ *Try To Find Any Other* **Home Computer**
That Gives You More For Your Money
Than **Texas Instruments . . .**

. . . Or Any Other Magazine
That Helps You Benefit More
From a **Home Computer** *Than*

99'er HOME COMPUTER magazine™

☞ **There Are None!**

THAT'S WHY THE EXPERTS SAY:

Once You Compare — There's No Comparison℠

Finding the best deal in a Home Computer isn't the whole story. You also need a timely information resource to help you get the maximum value out of your purchase. And that's where *99'er Home Computer Magazine* fits in . . .

As the *ONLY* magazine exclusively for the Texas Instruments brand of home, personal, and portable computers—including the versatile 16-bit TI-99/4A Home Computer, and the battery-powered Compact Computer series—*99'er Home Computer Magazine* is a MUST for all current users interested in entertainment, education, business, professional, and home applications. Also, it is an essential evaluation resource for those who want to know more about the friendly, value-packed TI computers in order to make intelligent purchase decisions in the future . . .

A Unique Combination of Resources

As a bonus, *99'er Home Computer Magazine* contains several special-interest supplements —*LOGO Times*, a journey of applied imagination, exploration, and self-discovery in the TI LOGO language environment; *Computer Gaming*, a treasure-trove of fun, excitement, and strategy; *Portable Computing*, an applications compendium for transportable computing power; and *PLATO's Progress*, a valuable users guide to an extensive computer-assisted learning library.

A Multi-Level Teacher, News Medium, & Buyer's Guide With Lots of FREE Software

Each BIG monthly issue of *99'er Home Computer Magazine* contains tutorials and applications for beginners, tips and "How-To" articles for intermediate-level users, advanced programming techniques that keep the pros coming back for more, as well as *half a dozen(!)* ready-to-run computer programs for EVERYONE. Additionally, there are photos and news items from shows where TI exhibits its products; advertisements from the leading producers and vendors of TI and TI-compatible software, hardware, and accessories; plus in-depth descriptions and reviews of the latest products and books—timely information to keep readers well informed and help them make wise purchase decisions.

99'er HOME COMPUTER magazine™

. . . From The People Who Know The Home Computer Best

SATISFACTION GUARANTEED

Or the Unfilled Portion of Your Subscription Will Be Refunded

☐ **YES—Please sign me up as a subscriber. Enclosed is payment or credit card billing information.**

Term	U S A	Canada & Mexico	Foreign Surface	Foreign Air
1-yr (12 issues)	☐ $25	☐ $32	☐ $43	Inquire
2-yr (24 issues)	☐ $45	☐ $52		
3-yr (36 issues)	☐ $63	☐ $70	Sample Issue	☐ $3.95

☐ Check or Money Order enclosed
(MUST BE **IN U.S. FUNDS** DRAWN ON U.S. BANK)

Name ..... **PLEASE**

Address ..... **PRINT**

City ..... **State** ..... **Zip** .....

Bill my: ☐ **VISA** ☐ **MasterCard**

► **Expiration Date** .....

Signature .....

Charge Card Number:

Please Mail Your Order To: ► **99'er Home Computer Magazine**
P. O. Box 5537
Eugene, OR 97405

MasterCard or VISA phone orders call:

TOLL-FREE **1-800-828-2212** WEST COAST TIME

Subscription Ordering Service M – F 7:00 AM – 5:00 PM, Sat 8:00 AM – 4:00 PM

In Oregon, Alaska, Hawaii Tel. (503) 485-8796

℠ **A service mark of Emerald Valley Publishing Co.**

Arithmetex . . . from p. 43

What is difficult about the game, especially for young people, is the sketchiness of its documentation. We're not told what we need or would like to know: In order to run the game you must disconnect your disk system if you have one; there's no way back to the menu but end-of-game or BREAK [FCTN] [4]. On the other hand, two of the four small paragraphs are devoted to a repetition of information available in the game's on-screen instructions.

It is evident that *Arithmetex*'s author has gone to some trouble to create pleasing graphics and interactive features for this "educational Math game." I wish that more thought had gone into a consideration of what kinds of learning might be desirable in a Math program, what kinds of incidental learning might take place as a result of peculiarities in the game's performance, and whether the range of ages the program purports to reach is really served.

PLEASE DON'T FORGET TO RETURN THE QUESTIONNAIRE ON THE FRONT BIND-IN CARD.

PLATO. . . from p. 66

cept with the student. The survey diskettes, though their purpose is not to instruct, also present the material in this way. This logical presentation of concepts means that students will actually *learn* something from completing the survey tests too.

PLATO Reading Skills Diskette

$49.95 suggested retail price

*These categories will be used primarily to evaluate the courseware. Whether or not they should be used to rate the Survey Diskettes is debatable.

System Requirements:
- TI 99/4A Home Computer
- Disk Memory System
- Memory Expansion
- PLATO Interpreter Solid State Cartridge
- Basic Reading Skills Survey Diskette

Up The Up Staircase

Myia's progress up this spiral staircase of learning did not stop when she rejoined Jill's reading group. In fact, one of the activities these two young friends enjoy doing together *now* is "playing PLATO."

Educational Activity Review Criteria

Documentation—rates the printed matter that comes with the activity. It notes whether the instructions are clear, comprehensive and easy to use, whether the machine configuration requirements are spelled out, and looks for such information as how to load the program, use the keyboard, and restart the activity.

Independence—focuses on how complicated it is for the user to understand the steps needed to progress through the learning activity. It measures to what degree the activity may be done by a user and the computer alone without parent/teacher guidance. Whether the actual reading level of the activity is appropriate for the suggested grade level is another major concern of this rating category.

Rewards—rates the audio-visual rewards as to their motivational effectiveness and appropriateness to the activity.

Graphics—rates the quality of the graphics and whether they enhance or detract from the educational purposes of the activity.

Concept Presentation—focuses on whether the concepts are presented clearly, in logical order and in enough depth for the learner to be able to apply learnings from the activity to other situations.

For the Holidays

GIFT SUBSCRIPTION CARD

SEND A GIFT SUBSCRIPTION TO 99'er HOME COMPUTER MAGAZINE THIS HOLIDAY SEASON!

Thanks to your thoughtfulness. . . .your friends, family and associates can enjoy a gift that keeps on giving all through the year. They'll enjoy 12 big issues of 99'er delivered right to their door each month.

When you place your subscription order, we will RUSH you an attractive Holiday Gift Card to convey your best wishes and announce your gift.

BONUS!

As our gift to all 99'er Gift-Givers, we have a *FREE* 99'er Discount Coupon Book. You can keep it for your own use, or give it to someone as another gift. The Discount Coupon Book contains over $25.00 worth of savings on 99'er-WARE and 99'er Bookstore items plus additional, surprise Bonus Coupons!?!

Call us TOLL FREE: 1-800-828-2212

with your credit card order, or send check or money order with coupon to:

99'er HOME COMPUTER magazine™

Attn: Gift Subscription Order Department
P.O. Box 5537 Eugene, OR 97405.

CALL OR MAIL TODAY!

This offer will expire Jan. 31, 1984

CLIP OR PHOTOCOPY THIS COUPON IF ORDERING WITH CHECK OR MONEY ORDER

Please enter a 1-year gift subscription for:

Name

Address

City State Zip

FROM: (Gift-Giver must complete this section)

Your Name

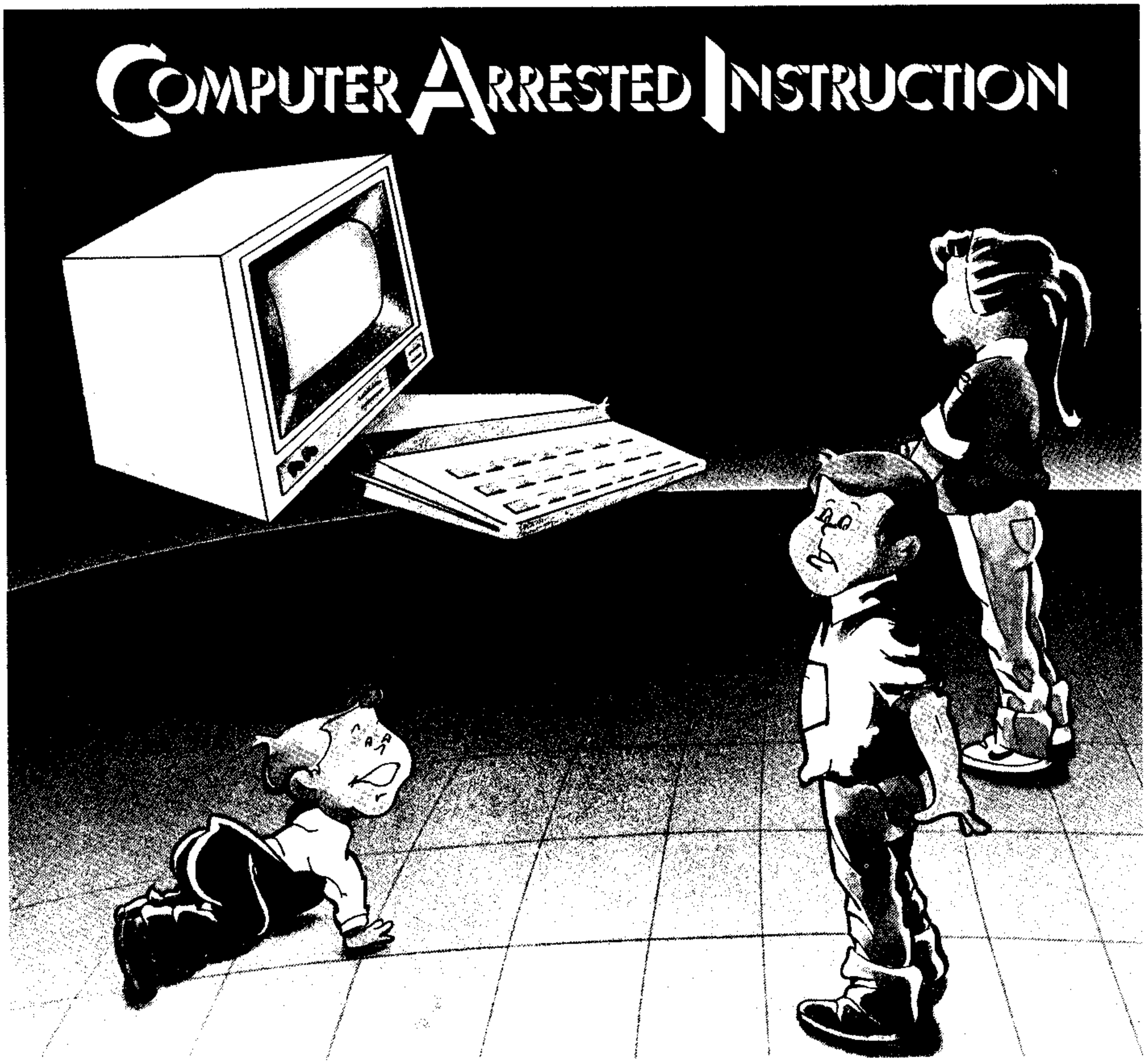
Address

City State Zip

☐ My payment of $25 is enclosed.
☐ I am giving myself a gift. It is a ☐ Renewal ☐ New Subscription.

PLEASE PRINT ALL INFORMATION

NOTE: For orders received by December 1, 1983, we'll send the gift card(s) and Discount Coupon Book to you, the Gift-Giver, unless otherwise requested. After December 1, we'll send the Gift Card directly to the recipient—with your name indicated as Gift-Giver. The Discount Coupon Book will be mailed separately to you.



It was 1958. The big blocky TV loomed in a corner like a mahogany refrigerator, its feeble speaker warning us that the Soviets had put up a satellite the size of a baby porcupine, and that our kids had better start working harder in science class or we'd be wiped out--if not that evening, maybe the next.

My uncle snorted at me from a turquoise vinyl easy chair, "You better get your grades up, if you know what's good for you." Out of the corner of my eye I could see his hairy arm plastered to the plastic chair arm, a can in his hand. My uncle was worried that our country was in a mess, and he expected a nine-year-old kid to take us out of it. Yes, it was my turn. My uncle had already done his part in "The Big One," WWII.

"I'd like to be a scientist," I told him. "Go to the Galapagos. Get iguanas and study them. Maybe have a deadly cobra as a pet—I mean—to study it."

"Lizards don't count when Russian rockets are whirling around our heads, he

by Greg Roberts

99'er HCM Staff

grumbled."

His remarks filled me with apprehension and guilt, and probably twisted me in some way that could be understood only by a very expensive shrink. Kids shouldn't have to go through that kind of pressure. The fifties were harder on me than they were on the folks on McCarthy's list.

Twenty-five years later, some people are acting as though Sputnik is back—this time sporting a keyboard and speaking a dozen languages. And this time there's more than one enemy: Not only are the Japanese trying to beat us in coming up with a "super computer"—the neighbors' kids are out to trample our own offspring in the race for "computer literacy." Panicky, some people are rounding up the kids, stationing them at work terminals, and even in—here comes that word—camps.

"Hey, back off a second," you say. "Computers are a fact of life now, and should be a mystery to no one, especially the young people who will make their livings with them. Besides, kids love 'em."

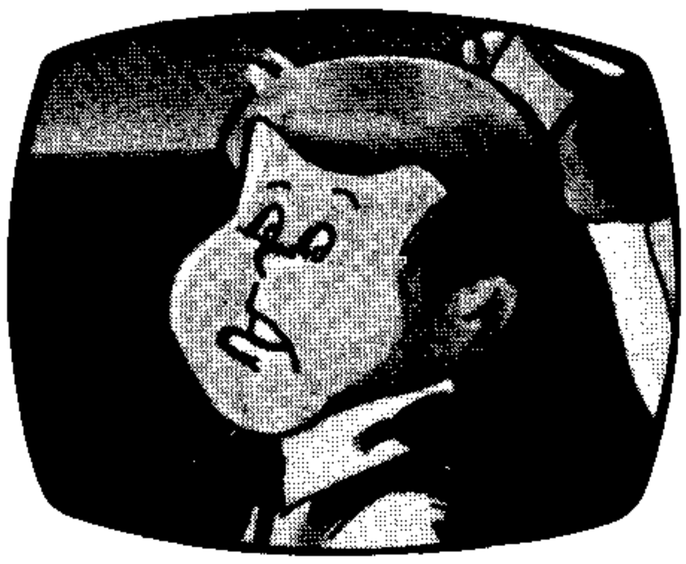
So, as a fierce debate uncurls its tentacles, *99'er HCM* attempts to light a cathode ray tube in the darkness by presenting the minutes of an unprecedented symposium on kids and computers—one that goes beyond the old theories—straight to the people who know this topic best. Our distinguished panel consists of three observers: the first two being a fifteen- and ten-year-old—with the last, a small-fry at just three and a half.

HCM: Welcome, and thank you for being here today.

Sarah Belem: The table is too high, and the lemonade's warm, but we're grateful for a chance to talk. Hardly anyone listens to kids, especially the computer industry. Most companies get their input from a bunch of old fogies—college students and

such—who are scared to death of anything new. So we end up with all sorts of problems such as clumsy "QWERTY" keyboards and killer doses of BASIC.

HCM: What would a kid look for in a computer?



Buddy Ardist: Whenever I get my hands on a machine, my first impulse is to try to wreck it; it's the ultimate test of a product. Now, that seems wrongheaded to adults, but it's one way to weed out the weak. You recall how the dinosaurs (my dad went to a supermarket and got me a book on these amazing lizards) went through an agonizing process of trampling out the aberrations among them—well, computers are still right in the midst of a similar struggle, and the kid factor will be very important in determining the outcome. Only with our help will a Tyrannosaurus take its place at the top.

HCM: Hey, that's not a bad name for a peripheral or something.

Buddy: Take it, it's yours. Personally, I find marketing and advertising about as exciting as bedtime in July. But, getting back to the main question: Until the industry gives us a computer we can take into the bathtub, this technology is going to walk on wobbly legs.

Rick Ottavit: There's this guy at our high school that's got a Trans Am with a computer thing on the dash that tells him all kinds of stuff with little blue lights.

HCM: Real good, Rick. Now let's get back to kids in computer careers.

Sarah: My family gave me some educational software that taught me a lot while fooling me into thinking I was playing video games. I liked it up to a point. . . and then I realized I needed some human contact. My parents wouldn't talk to me. They were downstairs trying to get a pie-shaped chart to grow money or something.

Buddy: That's not the computer's fault. That kind of abuse takes place because the machines are still novelties. Look at airplanes. When they first came out, people acted crazy around them—dancing on a biplane wing and all that. Now they're just a machine for visiting Grandma and Grandpa. Computers will settle down like that too.

Sarah: I'm sick of people saying that we don't have to know anything about programming—that it will quickly pass—like Dukes of Hazzard tricycles. Some may choose to sit back with their teddy bears, thinking that soon you'll be able to talk to a computer as if it were a telephone operator; well, don't hold your breath—unless you're throwing a tantrum.

Buddy: Sarah's way ahead of this game. She stands to gain by the apathy of others. Let's face it; no matter how simple computers eventually become—there will always be a software demand that must be fed. And someone's going to get paid a lot to keep all those babies from crying.

Sarah: I'm betting on it. This software boom is not just some overnight Kool-Aid-stand enterprise. It's bigger than mowing lawns and babysitting combined; bigger than having a paper route.

Buddy: Wow.

"Until the industry gives us a computer we can take into the bathtub, this technology is going to walk on wobbly legs."

Rick: I'm glad I don't hafta learn computers. I'm gonna be a race-car mechanic when I grow up.

Buddy: Not so fast. You'll need a computer—at least to play with—just to keep your mind from rotting. Being a mechanic is so boring, you'll need the diversion.

Rick: Being a mechanic is not boring.

Buddy: Oh yes it is.

Rick: No it isn't.

Buddy: Yes it is.

Rick: No. it isn't

HCM: Hold it now. Let's get relevant. For one thing, mechanics are already using computers to diagnose problems in engines, and those applications will soon be much more widespread. For another, computer hardware is going to become a vitally important component of the car of the future.

Rick: But if the computer is doing its job, I should just have to press buttons—maybe pull out a dip stick once in a while. I mean, I don't have to write a program, do I?

Sarah: Don't panic, Rick. You don't even have to write your name if you don't want to. Just make your X and pay the witness.

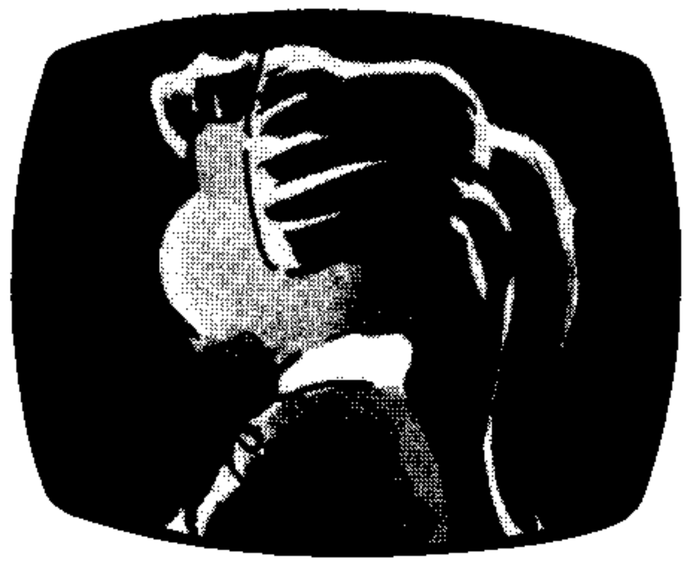
Rick: How would you like a knuckle sandwich?

Sarah: I can't believe you said that, you dope. Don't you know that they're taking this all down?

HCM: How do you feel about getting some training in a computer camp?

Rick: I can't see why anyone would want to sit in a cabin on a nice day and play with a computer when they could be out on a dirt bike tearing up the hillside.

Buddy: I agree with Rick, sort of. I mean, I was at camp last summer, and while we were writing some dinky little program to regulate the cash flow in our piggy banks, I could look out the window at the lake and see fish jumping all over the place—just waiting to be caught. I felt so frustrated, I tried to get the counselors to move the computer time to the evening—but somebody remembered that we had campfires and singing then. In the end I was able to stay at my console only by nailing my shoe soles to the floor. And from now on I'll do my computing in winter.



Sarah: Camp was fun for me, but the dough my parents spent on tuition could have bought me my own Pegasus and a private tutor.

Buddy: Your parents are financially naive.

HCM: Folks, we're running out of time here. Sarah, would you like to wrap things up for us?

Sarah: It seems to me that some will program, and some will be deprogrammed; some will ride, and some will steer.

Buddy: I can't argue that.

Rick: Is it over?

DISK DRIVES DISK DRIVES DISK DRIVES DISK DRIVES DISK DRIVES DISK DRIVES DISK DRIVES DISK DRIVES

PRICE BREAKTHROUGH

Super Sale on New Disk Drives

Starting at **$199.95 complete**
with Power Supply and Case!!

Tandon — Siemens — Remex — MPI — Teac — Shugart — Tabor

*All of Our Drives are Capable of Single **AND** Double Density Operation!*

RADIO SHACK[1] — HEATH/ZENITH[2] — APPLE[3]/FRANKLIN[4]
IBM/PC-TEXAS INSTRUMENTS[6] & MOST OTHER COMPUTERS

Drive a Hard Bargain!!™

5 M.B. - 12 M.B. with Power Supply Case, Cables & Software
Complete Systems . . . . . from $999.95

Dual sided 40 TK or 80 TK drives with Power Supply and Case—Call Toll Free

Diskette Breakthrough — 10 Pack in Library Case — $18.95

TOLL FREE ORDERING GENERAL AND TECHNICAL

1-800-343-8841 **1-617-872-9090**

Model I/III/IV Drives *(0 1 2 3)*	*starting at* ***$175.00***
Color Computer Drive (1 2 3)	***$Call Toll Free***
Apple/Franklin Compatible Add-On Drives with Case & Cables	$219.95
Apple/Franklin Compatible Drives with Controller	***$259.95***
Model I/III/IV Memory Upgrade	***Call Toll-Free***
Printers — Daisywheel/Dot Matrix	***$Call Toll Free***
Diskettes in Library Cases	$18.95
Cases and Power Supplies — (Single-Dual-**1/2 Height**)	*starting at $49.95*
Printer **Buffers** 8K to 512K	starting at $143.95
Percom Double Density Controller (Model I)	***$Call Toll Free***
Holmes Model I/III Speed-up Mod-VID/80	*starting at* $90.00
Gold Fingered **Edge Card Extenders**	*starting at* $13.00
Cables — Printer/Disk Drive	*starting at $25.00*
DOSPLUS	***$Special Prices***
Repair Services Now Offered — *Fast Turn-a-Round*	***$Call Toll-Free***

Warranty on Disk Drives — 6 Months — *Extended Warranty $ Call Toll Free*

SOFTWARE SUPPORT, INC.

One Edgell Road, Framingham, MA 01701 (617) 872-9090
Hours: Mon. thru Fri. 9:30 am to 5:30 pm (E.S.T.) Sat. 10:00 am to 4:30 pm

Dealer inquiries invited.

TERMS:
M.C./Visa/Amex and personal checks accepted at no extra charge.
C.O.D. please add $3.00.
Shipping: Please call for amount.
Not responsible for typographical errors.

Canada
MICRO R.G.S. INC.
751, CARRE VICTORIA, SUITE 405
MONTREAL, QUEBEC, CANADA, H2Y 2J3
Regular Tel. (514) 845-1534
Canadian Toll Free 800-361-5155

1 "TANDY CORPORATION
2 "ZENITH DATA SYSTEMS
3 "APPLE COMPUTER CORP.
4 "FRANKLIN COMPUTER, INC.
5 "IBM CORPORATION
6 "TEXAS INSTRUMENTS

DISK DRIVES DISK DRIVES DISK DRIVES DISK DRIVES DISK DRIVES DISK DRIVES DISK DRIVES DISK DRIVES

Squeeze. . . from p. 58

Logical variables are equal to either −1 or 0, corresponding to true and false. If a term is to be included in a formula conditional on a logical variable, multiply the variable by the complement of the term (or subtract the product). For example, the location of the plane is a function not only of its own speed and direction, but also of the wind's speed and direction (once the plane has taken off). So I add a wind term to the plane's position in both the X and Y coordinates, but only if the plane is in flight. The variable INF is true if the plane is in flight; WX is the wind velocity component in the X direction.
The formula looks like this:

XPOS = XPOS + [*term due to motion*] −INF * WX

Division by zero can occur in some formulas and you have to guard against it. A simple technique for preventing this is to first test the divisor, then branch around the division. Here is a method that eliminates the test and produces a value which may be entirely acceptable:

X=B/(C−(C=0)*.001)

If C is exactly 0, the logical expression C=0 will be −1 and the divisor expression will be equal to .001. X would thus be B/.001. If C is not exactly 0, the divisor is equal to whatever C is, and X would be B/C. The factor .001 could be set to whatever you desire, or it could be left out (which would make the divisor equal to 1 and thus set X equal to B if C were 0).

If you press the [C] or [D] keys when running the flight simulator, power is increased or decreased, respectively. The letter C is for climb (increased power), and D is for descend (decreased power). Power is shown as a percentage of full power; that is, power ranges from 0 to 100%. If you press down the [C] key, the power will increase by 5%. It might go from 50% to 55%. If you continue to hold the key down, it will increase again, but this time by 10%, to 65%. If you still hold it down, it increases by 15%, to 80%. In other words, the rate of increase (or decrease) increments by 5% as you continue to hold the key down. This allows the two keys to be used for both fine tuning and rapid adjustment, either up or down.

In order to implement this scheme, three variables are needed:

PP	Percent power	0—100%
PPI	PP increment	+ or − 5%, 10%, 15%, etc.
K	The key pressed	67 if C, 68 if D

```
10 IF K=68 THEN 90
20 IF PPI>0 THEN 40
30 PPI=0
40 PPI=PPI+.05
50 PP=PP+PPI
60 IF PP<1 THEN 150
70 PP=1
80 GO TO 150
90 IF PPI<0 THEN 110
100 PPI=0
110 PPI=PPI-.05
120 PP=PP+PPI
130 IF PP>0 THEN 150
140 PP=0
150 program continues
```

Listing 3a: Simple way to program C and D power keys

Listing 3a shows a simple way to code this part of the program. (Note: This part of the program is not invoked if [C] or [D] is not pressed.) Listing 3b shows a method which is about twice as compact. It depends on two "tricks."

```
10 A=.1*(67.5-K)
20 IF SGN (A)=SGN(PPI) THEN 40
30 PPI=0
50 PP=PP+PPI
60 IF ABS(PP-.5)<.5 THEN 80
70 PP=.5*(1+SGN(PP-5))
80 program continues
```

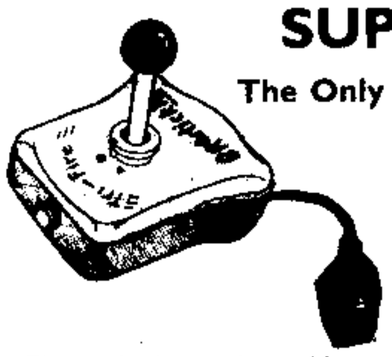
Listing 3b: More compact way to program C and D power keys

The first trick is in statement 10, which sets the temporary variable A to + or −.05, depending on the value of K but without using a conditional. This value of A is then used to in-

TEX-COMP™

USERS SUPPLY DIVISION PROUDLY INTRODUCES

SUPER JOYSTICK II



The Only JOYSTICK that can control THE MUNCHMAN!

NEW Available exclusively from Tex-Comp

Rated #1 August 1982 Creative Computing

Improved version

The case has been redesigned for easy hand-held operation. The solid steel, short-throw control shaft with an arcade-style ball on the top offers precise, 8-position action. The "soft-touch" fire button is located at the top end of the molded base allowing either right or left-hand play. This improved fire button location is designed to take advantage of the greater dexterity of the index finger resulting in quicker response time and less fatigue during extended play.

SAME ACTION AS THE REAL ARCADE GAMES

•SWITCHABLE GATE-PLATE™: Setting the switchable gate-plate indicator to the 4-way position allows only horizontal or vertical motion dramatically improving response for maze-type games

Helps you make higher levels of Munchman . . . if you're fast enough.

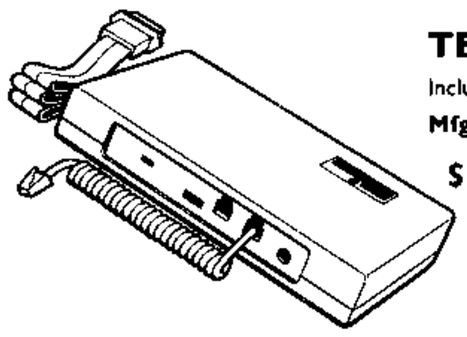
NEW PRICE $29.95
Effective Sept. 1, 1983
Free Shipping*
Second Unit Only **$24.95**
*Continental U.S.

With FREE Port Expander

AND

The SIGNALMAN™ MARK III TI99/4A COMPATIBLE MODEM

Finally, a low cost, direct connect, high quality and super reliable TI-99/4 and 99/4A compatible modem that comes complete and ready to use—just plug it into a TI RS/232 interface or expansion card.



TEX-COMP PRICE $94.95
Including shipping and insurance
Mfg. Suggested List Price $139.00

$100. SPECIAL—FREE Source/ TEXNET sign on and first hour, with purchase of Signalman MK III

ACCESSORIES: 9V-DC Power Supply $10.95 p.p.

SPECIFICATIONS
Data Format. . . Serial, binary, asynchronous
Operate Mode Manual dial. Automatic ANSW/ORIG selection
Data rate. . . . . 0 to 300 bps. full duplex
Modulation. . . . . Frequency shift-keyed (FSK)
Line Interface. . . . . Direct-connect
Data Interface. TI RS-232C compatible, built-in cable to computer

Transmit Frequency.	ORIG	ANSW
MARK	1270 Hz	2225 Hz
SPACE	1070 Hz	2025 Hz

Transmit Frequency Accuracy. . . . . ±0.01%
Transmit Level. . . . . -12 dbm typical
Receive Frequency. . . . . ORIG . . ANSW

MARK	2225 Hz	1270 Hz
SPACE	2025 Hz	1070 Hz

Receive Frequency Tolerance. . . . . ±0.5%
Carrier Detect Threshold. . . . . -44 dbm typical
Carrier Detect Indicator . . . . . Audible tone
Power Requirement Internal 9V transistor battery.* or 110 VAC through adapter*
Size. . . . . 8 1/4" × 4 3/8" × 1 3/8"
*Not included

Meet the direct-connect SIGNALMAN MARK III designed for use with your TI-99/4A Accessories Interface . . . the smallest, lightest, most compact modem available today, its long life 9-volt internal battery and exclusive audible Carrier Detect Signal allow you to install the SIGNALMAN anywhere . . . out of the way, and out of sight. Now, there's no need for messy cables, and no need to look at a LED to verify carrier.

Your SIGNALMAN transmits both voice and data over all common telephone lines, and is fully compatible with Bell 103 modems—putting your computer in instant communications with thousands of other computers. And when you're in the data position, your SIGNALMAN automatically changes from ORIGINATE to ANSWER and back again as the need arises—ending all that confusion.

Anchor Automation has taken the fuss out of communications.
For business or fun, SIGNALMAN is the ideal modem.

NEW FEATURE—WORKS WITH ANY PHONE

Comes complete with cable and cord. This is a unit specially designed for the TI-99/4A and should not be confused with any other Anchor/Signalman models that sell for less and/or are not compatible with the TI-99/4A.

Send for our free Order Kit- discounts on entire product line
Mail Check or Money Order to:

Tex-Comp
P.O. Box 33084, Granada Hills, CA 91344
213-366-6631

VISA/Mastercharge-add3% for credit card orders. Send Cashiers Check or Money Order to avoid delay. Calif. orders add 6½% sales tax.

SPECTACULAR HOME COMPUTER GRAPHICS ANIMATION CONTEST

99'er HCM is looking for TI Home Computer programs that cross new computer graphics animation frontiers into uncharted territory. Some of the factors we are looking for in the submissions are:

- The graphics should show animated sequences such as vehicles crossing a bridge or a plane flying through a mountain pass.
- The graphics must be visually attractive.
- The animation sequence must last for at least 3 minutes duration and must be non-repetitive.
- The program should use a variety of random branching techniques as it passes through the animation sequence loop.
- The animation sequence may use any language that will run on the TI Home Computer (Forth, TI BASIC, Extended BASIC, LOGO, Assembly Language, . . .).

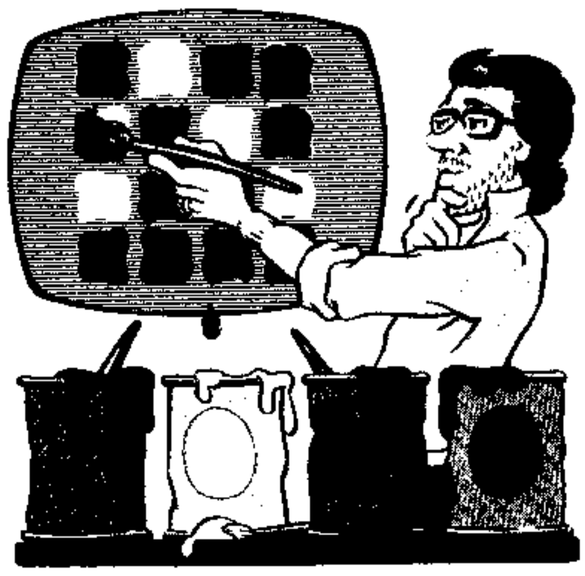
An official entry form must accompany each program that you submit. To request your entry submission form, call (503) 485-8796 or write to *99'er HCM* Graphics Animation Contest, 1500 Valley River Drive, Suite 250, Eugene, OR 97401. Submissions without an official entry form will not be accepted or acknowledged.

Cash prizes will be awarded to the most exciting, dazzling, breathtaking graphics sequence received!

Grand Prize of $500
2 Second-place prizes of $250
5 Third-place prizes of $100
10 Fourth-place prizes of $50

All prize winners will also receive a one year free subscription to *99'er Home Computer Magazine*.

All submissions must be received no later than December 31, 1983—so get your entry form soon.



This offer void where prohibited by law.
Home Computer Graphics Animation Contest not open to *Emerald Valley Publishing Co.* employees or their families.

SUBSCRIBE TODAY
SAVE OVER 40% on the SINGLE-COPY PRICE!

RENEW TODAY
DON'T MISS A SINGLE ISSUE!

99'er HOME COMPUTER T.M. **Subscription**

☐ NEW SUBSCRIPTION ☐ SUBSCRIPTION RENEWAL

(Allow 6-8 wks for your first issue)

Term	U.S.A	Canada & Mexico	Foreign Surface	Foreign Air
1-yr (12 issues)	☐$25	☐$32	☐$43	Inquire
2-yr (24 issues)	☐$45	☐$52	Please enclose payment in U.S. FUNDS or Credit Card billing information as indicated below.	
3-yr (36 issues)	☐$63	☐$70		

BACK ISSUES AVAILABLE WHILE SUPPLIES LAST

Circle Issues Desired **Vol. 1 No. 6** **Nov '82** **Dec '82** **Jan '83** **Feb '83** **Mar '83** **Apr '83** **May '83** **Jun '83** **Jul '83** **Aug '83** **Sep '83** **Oct '83**

Postpaid
$3.95 ea. USA
$5.50 ea. Foreign Surface
$4.50 ea. Canada & Mexico
$7.50 ea. Foreign Air Mail

TOTAL ☐

Bill my: ☐VISA ☐Master Card

☐☐☐☐☐☐☐☐☐☐☐☐☐☐☐☐ ► _____ Expiration Date

Account No.

_____ Tel. No.

_____ Signature

RENEWAL SUBSCRIBERS PLEASE INCLUDE INFORMATION BELOW FROM PREVIOUS MAILING LABEL:

_____ Subscriber Code Number (line above your name on mailing label)

NAME _____ PLEASE

ADDRESS _____ PRINT

CITY _____ STATE _____ ZIP _____

Please indicate below any change in name or address.

NAME _____ PLEASE

ADDRESS _____ PRINT

CITY _____ STATE _____ ZIP _____

Check enclosed ☐ MUST BE IN U.S. FUNDS DRAWN ON A U.S. BANK

Address shown is ☐Business ☐Home

SUBSCRIPTION BONUS: ***Only Subscribers Will Receive*** **The 99'er BUYERS'S GUIDE** ***Bound Into Each Issue***

_____

_____

_____

ORDER TOLL-FREE
1-800-828-2212
ORDER LINE ONLY—operators can only take subscription or merchandise credit card orders (MasterCard/VISA only)
In Oregon, Alaska, Hawaii Tel. (503) 485-8796

PLACE FIRST CLASS POSTAGE STAMP (CANADIANS, PLEASE AFFIX 37¢ POSTAGE)

99'er HOME COMPUTER magazine ™

P.O. Box 5537
Eugene, OR 97405

NAME _____ PLEASE

ADDRESS _____ PRINT

CITY _____ STATE _____ ZIP _____

Check enclosed ☐ MUST BE IN U.S. FUNDS DRAWN ON A U.S. BANK

Address shown is ☐Business ☐Home

Bill my: ☐VISA ☐Master Card

☐☐☐☐☐☐☐☐☐☐☐☐☐☐☐☐ ► _____ Expiration Date

Account No.

_____ Tel. No.

_____ Signature

99'er BOOKSTORE™

QTY	TITLE	PRICE	TOTAL AMOUNT

Shipping and Handling: In U.S.A.—$2.50 for 1 book; 75¢ for each additional book. Foreign Surface—add $3.00 to total U.S.A. shipping costs.

POSTAGE SUBTOTAL _____

99'er MAGAZINE ON TAPE™

Circle tapes desired

M1/6, M2/1, M2/2, M2/3, M2/4, M2/5

TOTAL _____

SHIPPING—$1.00 for 1st TAPE, 50¢ ea. add.
Foreign Orders Shipped Airmail—$3.00 for 1st TAPE, 75¢ ea. add.

SUBTOTAL _____

99'er-ware™

*U.S. ONLY—FOREIGN SURFACE ADD $2.00 TO TOTAL SHIPPING COSTS.

QTY	ITEM	PRICE	TOTAL AMOUNT
	99'er FINDER – BINDER	**$11.95**	
	SHIPPING—$3.00 ea.*		_____
	6—DIGITAL COMPUTER CASSETTES	**$7.00**	
	SHIPPING—2.00 ea.*		_____
	SPECIAL: 99'er Finder – Binder		
	•with 6 blank cassettes for only	**$16.95**	
	•with 12 blank cassettes for only	**$21.95**	
	SHIPPING—$4.00 ea.*		_____
	TEX – SETTE ADAPTOR	**$5.95**	
	SHIPPING—$1.00 ea.*		_____
	DUST COVERS: See prices on page 84. Indicate choices below.		
	SHIPPING—$2.00 for 1st COVER, 50¢ ea. add.*		_____
		SUBTOTAL	
		TOTAL AMOUNT OF ORDER	

Les Izmore and DeBug

BY LAREDO & ROBERTS

LES AND DEBUG ARE TAKING A MONDAY MORNING TOUR OF THEIR BUSINESS: CHAMELEON COMPUTER & PETFOOD Co, Ltd.

$Z = \frac{M^2}{\sqrt{HCM}}$

WE'RE LUCKY TO HAVE ALL THESE ECCENTRIC GENIUSES WORKING FOR US – BUT WE DON'T HAVE ONE BUSINESS MIND IN THE BUNCH!

NO ONE WITH THE MARKETING SAVVY THAT COULD MAKE US NUMBER ONE!

SOME COMPUTER COMPANIES ARE NOW IMPLEMENTING EXECS FROM OLD-GROWTH INDUSTRIES LIKE SODA POP AND CIGARETTES. IT CREATES A WHOLE NEW MARKETING DYNAMIC! LET'S PLUG INTO A DATABASE AND SCAN FOR TALENT.

HMMM... CEO's FROM HOOLA HOOP COMPANIES, PET ROCKS, LAVA LAMPS. HEY, HERE'S A GUY WHO'S BEEN IN AN OLD GROWTH COMPANY FOR 30 YEARS. HE'S RUDY REDDENRUFFIN FROM NIKKEM INDUSTRIES. THEY MAKE REDWOOD CHAINSAW SCULPTURES: BEARS, TOTEM POLES AND SASQUATCHES!

I WONDER IF WE COULD BRING HIM ON LINE AS A CONSULTANT?

SOON.

THE WAY I FIGURE IT, THE COMPUTER MAKERS ARE ONLY REACHIN' BETWEEN 1 AND 43% OF THE HOUSEHOLDS. THEY'RE NOT GETTIN' THEIR WEDGES INTO A WHOLE BIG VIRGIN STAND OF WOODCUTTERS, TRAPPERS, PROSPECTORS AND MOUNTAIN MEN. THESE BOYS ARE JUST DYIN' TO BE REACHED!

$ $ $

MOST COMPUTER HARDWARE IS EVEN BUILT WRONG! IT'S MADE OUT OF THREE VERY LIMITED RESOURCES: PETROLEUM, SILICON AND ALUMINUM.

INCORRECT STATEMENT! ALUMINUM AND SILICON ARE THE TWO MOST ABUNDANT ELEMENTS ON EARTH!

STILL, THEY'VE GOT TO RUN OUT EVENTUALLY!

AND LET ME WARN YA' ABOUT DEPENDIN' ON PETROLEUM DERIVED PLASTICS. A PRODUCT THAT COMES FROM DINOSAURS IS APT TO BECOME ONE!

HOW CRUDE!

WHAT ARE YOU SUGGESTING?

RETOOL! CONVERT TO WOOD - EVERYTHING FROM KEYBOARDS TO DISK DRIVES! IT LASTS FOREVER! LOOK AT A STRADIVARIUS: 300 YEARS OLD AND STILL WORKS. CAN PLASTIC MAKE A CLAIM LIKE THAT?

BUT WHAT ABOUT QUALITY CONTROL?

WELL, YA GOTTA WATCH FOR YER KNOTHOLES AND YER TERMITES. YA GOTTA TURN ON YER SYSTEM ONCE IN A WHILE TO DRIVE OUT YER CARPENTER ANTS!

HOLY BAD SYNTAX, DeBUG! I'M AFRAID RUDY'S PROGRAM WON'T LOAD, WON'T RUN AND WOULDN'T DEBUG WITH AN ARMY OF AARDVARKS!

TRY TO ENDURE THE PAIN A BIT LONGER. HE'S GOING TO A CHAINSAW SHOW LATER. MEANWHILE, I'M TRYING TO SET UP A TELE-CONFERENCE WITH LEE IACOCCA!

END

Creative . . . from p. 18

```
 840 DISPLAY AT(12,1):"MINUTE HA
     ND IS" :: DISPLAY AT(13,1):
     "ON THE";:BH
 850 CALL SPRITE(#13,CDE(BH),2,B
     HR(BH),BHC(BH)):: CALL SPRI
     TE(#BH,132,13,ROW(BH),COL(B
     H))
 860 FOR PAUSE=1 TO 300 :: NEXT
     PAUSE
 870 DISPLAY AT(17,1):"WHAT TIME
      IS IT?"
 880 DISPLAY AT(19,1):"(TYPE HOU
     R,THEN 'ENTER')"
 890 DISPLAY AT(20,1):"(AFTER :
     TYPE MINUTES)"
 900 ACCEPT AT(17,17)BEEP VALIDA
     TE(DIGIT)SIZE(2):HOUR
 910 DISPLAY AT(17,19):":"
 920 ACCEPT AT(17,20)BEEP VALIDA
     TE(DIGIT)SIZE(2):M
 930 IF HOUR=2 AND M=5*V THEN 10
     60 ELSE 940
 940 IF TRY=2 THEN 1000 ELSE 950
 950 TRY=TRY+1
 960 DISPLAY AT(21,8):"WRONG!! T
     RY AGAIN" :: CALL HCHAR(17,
     19,32,7)
 970 FOR PAUSE=1 TO 400 :: NEXT
     PAUSE
 980 CALL HCHAR(21,8,32,23)
 990 GOTO 900
1000 CALL SAY("THE CORRECT TIME
     IS")
1010 IF V=1 THEN 1040 ELSE 1020
1020 DISPLAY AT(22,2):"THE CORRE
     CT TIME IS";Z;":";V*5
1030 GOTO 1130
1040 DISPLAY AT(22,2):"THE CORRE
     CT TIME IS";Z;":05"
1050 GOTO 1130
1060 CALL SAY("CORRECT VERY GOOD
     ")
1070 FOR COUNT=1 TO 5
1080 CALL HCHAR(22,12,32,10)
1090 FOR PAUSE=1 TO 75 :: NEXT P
     AUSE
1100 DISPLAY AT(22,10)BEEP:"CORR
     ECT!"
1110 FOR PAUSE=1 TO 75 :: NEXT P
     AUSE
1120 NEXT COUNT
1130 CALL SAY("WANT TO TRY AGAIN
     ")
1140 DISPLAY AT(24,1):"ANOTHER T
     IME?"
1150 ACCEPT AT(24,14)BEEP VALIDA
     TE("YESNO")SIZE(3):ANS$
1160 IF SEG$(ANS$,1,1)<>"Y" THEN
      1360
1170 FOR R=9 TO 13
1180 CALL HCHAR(R,1,32,32)
1190 NEXT R
1200 FOR R=15 TO 24
1210 CALL HCHAR(R,1,32,32)
1220 NEXT R
1230 CALL HCHAR(SHR(SH),SHC(SH),
     32)
```

```
1240 FOR I=1 TO 13
1250 CALL DELSPRITE(#I)
1260 NEXT I
1270 GOSUB 1300
1280 GOTO 730
1290 STOP
1300 REM DISPLAYING THE CLOCK
1310 C=2
1320 FOR I=1 TO 12
1330 CALL SPRITE(#I,CODE(I),C,RO
     W(I),COL(I))
1340 NEXT I
1350 RETURN
1360 CALL DELSPRITE(ALL)
1370 CALL CLEAR
1380 DISPLAY AT(12,12):"GOODBYE"
1390 CALL SAY("GOODBYE")
1400 RUN "DSK1.LOAD"
1410 END
```

Working with Money

```
 100 REM ********************
 110 REM * WORKING WITH MONEY *
 120 REM ********************
 130 REM BY DAN ROZLER
 140 REM 99'ER VERSION 2.13.1XB
 150 CALL CHAR(37,"083C4A484844A3
     C08")
 160 CALL CHAR(136,"000000FFFF00
     0000"):: CALL COLOR(14,7,8)
 170 FOR I=1 TO 6
 180 READ CENTS(I),MONEY$(I)
 190 DATA 1,PENNIES
 200 DATA 5,NICKELS
 210 DATA 10,DIMES
 220 DATA 25,QUARTERS
 230 DATA 50,"HALF DOLLARS"
 240 DATA 100,DOLLARS
 250 NEXT I
 260 CALL CLEAR
 270 ROW=2
 280 FOR I=1 TO 5
 290 READ N$(I),S$(I)
 300 DISPLAY AT(ROW,8):N$(I);CHR
     $(37);TAB(13);S$(I)
 310 ROW=ROW+2
 320 DATA 1,PENNY
 330 DATA 5,NICKEL
 340 DATA 10,DIME
 350 DATA 25,QUARTER
 360 DATA 50,"HALF DOLLAR"
 370 NEXT I
 380 DISPLAY AT(12,9):"$1";TAB(1
     3);"DOLLAR"
 390 CALL HCHAR(1,9,136,18)::  CA
     LL HCHAR(13,9,136,18)
 400 RANDOMIZE
 410 A=INT(RND*6+1)
 420 B=INT(RND*6+1)
 430 K=INT(RND*4+1)
 440 IF A=B THEN 400 ELSE 450
 450 IF A<B THEN AMOUNT=K*CENTS(
     B)::  D=CENTS(A)::  F$=MONEY$
     (A)::  L$=MONEY$(B)::  GOTO 4
     70
```

Loader

```
100 REM  *********
110 REM  *  LOADER  *
120 REM  *********
130 REM  BY DAN ROZLER
140 REM  99'ER VERSION 2.13.1XB
150 REM
160 REM  THIS PROGRAM MUST BE NAMED
    "LOAD", AND MUST RESIDE
    ON A DISKETTE IN DISK DRIV
    E 1.
170 REM  THE PROGRAMS IT CALLS I
    N LINES 390-430 MUST BE ON
    THE SAME DISKETTE.
180 REM
190 CALL CLEAR
200 CALL CHAR(136,"FFFFFFFFFFFFFF
    FF")
210 CALL COLOR(14,7,7)
220 CALL HCHAR(4,2,136,30)::CALL HCHAR(16,2,136,30)
230 DISPLAY AT(1,5):"----------"
240 DISPLAY AT(2,5):"ACTIVITIES AVAILABLE"
250 DISPLAY AT(3,5):"----------"
260 FOR C=1 TO 5
270 READ N$
280 DISPLAY AT(C+5,2):N$
290 NEXT C
300 DATA "1. GUESS A NUMBER"
310 DATA "2. GUESS A LETTER"
320 DATA "3. GUESS A MUSICAL NOTE"
330 DATA "4. TELL THE TIME"
340 DATA "5. WORKING WITH MONEY"
350 DISPLAY AT(18,3):"IF NO SELECTION IS TO BE MADE,PRESS QUIT."
360 DISPLAY AT(19,3):"MADE, PRESS ENTER"
370 DISPLAY AT(21,11):"CHOICE?":: DISPLAY AT(23,11):"(PRESS ENTER)":: ACCEPT AT(21,18)VALIDATE(DIGIT)BEEP:A
380 ON A GOTO 390,400,410,420,430
390 RUN "DSK1.GUESSN"
400 RUN "DSK1.GUESSL"
410 RUN "DSK1.MUSIC"
420 RUN "DSK1.TIME"
430 RUN "DSK1.MONEY"
440 END
```

```
460 AMOUNT=K*CENTS(A)::  D=CENTS(B)::  F$=MONEY$(B)::  L$=MONEY$(A)
470 TRY=1
480 QUANTITY=INT(AMOUNT/D)
490 DISPLAY AT(16,1):"HOW MANY";F$;:  DISPLAY AT(17,1):"I";K;L$;"?"
500 CALL HCHAR(19,3,136,12)::  CALL HCHAR(21,3,136,12)::  DISPLAY AT(20,1):"ANSWER";
510 ACCEPT AT(20,8)BEEP VALIDATE(DIGIT)SIZE(3):ANS
520 IF QUANTITY=ANS THEN 610 ELSE 530
530 IF TRY=2 THEN 590 ELSE TRY=TRY+1
540 DISPLAY AT(19,19):"WRONG!":: DISPLAY AT(20,18):"TRY AGAIN"
550 CALL SAY("TRY AGAIN")
560 FOR PAUSE=1 TO 300 :: NEXT PAUSE
570 CALL HCHAR(19,20,32,10)::  CALL HCHAR(20,18,32,13)
580 GOTO 510
590 CALL HCHAR(19,3,136,28)::  CALL HCHAR(21,3,136,28)::  DISPLAY AT(20,1):"THE CORRECT ANSWER IS";QUANTITY
600 GOTO 690
610 DISPLAY AT(19,14):"THAT'S CORRECT"
620 CALL SAY("THAT IS CORRECT")
630 FOR COUNT=1 TO 5
640 CALL HCHAR(21,17,32,14)
650 FOR PAUSE=1 TO 75 :: NEXT PAUSE
660 DISPLAY AT(21,17)BEEP:"VERY GOOD"
670 FOR PAUSE=1 TO 75 :: NEXT PAUSE
680 NEXT COUNT
690 DISPLAY AT(24,1):"ANOTHER PROBLEM(YES/NO)?"
700 ACCEPT AT(24,25)BEEP VALIDATE("YESNO")SIZE(3):ANS$
710 IF SEG$(ANS$,1,1)="N" THEN 760
720 FOR E=14 TO 24
730 CALL HCHAR(E,3,32,28)
740 NEXT E
750 GOTO 400
760 CALL CLEAR
770 DISPLAY AT(12,12):"GOODBYE"
780 CALL SAY("GOODBYE")
790 RUN "DSK1.LOAD"
800 END
```

Back Issues of 99'er HOME COMPUTER magazine

are Still Available . . .
but quantities are limited

so ORDER TODAY!

Each Only $3.95 postpaid

ISSUE #1 OUT OF PRINT
ISSUE #2 OUT OF PRINT
ISSUE #3 OUT OF PRINT
ISSUE #4 OUT OF PRINT
ISSUE #5

ISSUE #6 (Partial Contents)
• How to Produce Sound Effects • Debugging a Game Program • How to Start a Users' Group • Verbose: A Speech Vocabulary Expansion Aid • Color Mapping • Dynamic Manipulation of Screen Character Graphics • The Beginner's Guide to Cassette Operation With the Home Computer • Pre-School Block Letters and Data Compaction • Picking the Ponies in TI BASIC • Battle Star Space Game • 3-D Animation on the Home Computer • Programming Tips • Who is LOGO For? • Tower of Hanoi in TI LOGO • A Review of the TI Lesson-Development Software • An Interview with a Game Designer • Learning Assembly Language with a Magic Crayon • and much, much more.

NOVEMBER 1982 (Partial Contents)
• Chatting with Your Micro: Languages for the Home Computer • A Review of the Smith Corona TP-1 Daisy Wheel Printer • The Micro Jaws Arcade Game • A Knight's Tour in TI BASIC • LOGO Has Style • ASPIC: A Language for Children • A p-System Beginners' Tutorial • An Interview with a p-System Pioneer • A Mini-Memory Screen Dump to the Home Computer Printer • Up Scope!—An exciting Undersea Combat Game • Strategy for Munch Man • A Brief Encounter with a TI Hand-Held Computer • 99'er Shopping Bus • A Pocket Battleship • Sub-Programs in Extended BASIC • Arcade & Adventure Game Reviews • and much, much more.

DECEMBER 1982 (Partial Contents)
• Tex-Scribe: A Text Editor for the Home Computer • A Christmas Computer Carol • Managing a Mailing List the Futura Way • Parsec: The Arcade Game • Plotting With the Home Computer—Pixel by Pixel • Preventing the Situation—Oh No! Memory Full • A Colorful Tour of TI-Fest: The Home Computer Show • Santa's Workshop: The Making of a Home Computer • The Turtle Arcade: Movies & Video Games in LOGO • Controlling a BASIC Termite • The 99'er Gold Rush—An Arcade/Adventure in the Home • 99'er Digest of News & Happenings in the TI World • Plus Games, Reviews, and much, much more.

JANUARY 1983 (Partial Contents)
• Computer Assisted Instruction for the Handicapped • p-System Basics • Debugging in LOGO • The Dow-4 Gazelle Flight Simulator • Note Whiz and Pitch Master Musical Game Reviews • Learning With the PLATO Computer Library • Strategies for Adventure Gaming • Death Drones • Using the Line-By-Line Assembler • Close Encounters of the Simon Kind • Electrical Engineering Education Program • Interview With an Arcade Game Designer • TI Invaders • Programming With Pascal • Cyber-Dice • News and Happenings in the Home Computer World • Arcade Game Reviews • The Thief Adventure Game • Programming Tips • and much, much more.

FEBRUARY 1983 (Partial Contents)
• Texas Instruments at the Winter Consumer Electronics Show • Home Computer Printers on Review • How to Create Math Daisies in LOGO • Vectors in LOGO • ASPIC: A Language for Teachers • The Joys of Adventuring—Part 2 • Programming Pointers with Chuck-A-Luck—Part 4 • Interview With the Voice of Parsec • Why You Need a Printer for Your Home Computer • Lifeline to Titan Space Game • Night Blockade Battleship Game • Tower of Hanoi Pocket Program • Computer Gaming Software Reviews • News of Late Developments in the World of Home Computers • and much, much, more.

MARCH 1983 (Partial Contents)
• An Introduction to the TI-99/2 Basic Computer • The Hex-bus and the 4/A Connection • Making Your Own Say and Spell Game • Disabled Children Learn and Grow • Super Cataloger—A Review of a Disk Library Utility Program • TI's New CC-40 Compact Computer • Robots and Their Social Impact • Twenty Questions With Robot Redford • The Gravity of LOGO • Joystick Jockey—An Overview of Remote Controllers • Parsec Strategy • Converting Extended BASIC to Assembly Language • Matrix Muncher • Mini Memory Disassembler Utility • Pulling the Shade on Sprites • Letters on LOGO • Tiny Tutorials • Games programs, reviews, and much, much more.

APRIL 1983 (Partial Contents)
• Computer Assisted Savings Planning to Build Your Nest Egg • Tex-Cipher Writes and Decodes Secret Messages • Crossbytes—Computer Vocabulary Crossword Puzzle • Cutting Corners On Your Food Budget Using Coupons • Introducing Financial Planning with Multiplan • The Design Philosophy of the Compact Computer • LOGO Takes On the Popular Fifteen Puzzle • Super Language—Programming Sprites in Mini Memory • Colorful Word World—Reading Readiness for Pre-schoolers • Gameware Buffet's A-Maze-ing Boa Alley Game • Giant and Dwarfs Entrapment Game • Game Reviews • Programming Tips • Money Saving Hints • and much, much more.

MAY 1983 (Partial Contents)
• A Consumer's Guide to Word Processing • Word Processing Market Basket • A Generalized Filing Program for VIPs • The Multiplan Medium Balances Your Checkbook and Budget • Activity Accountant Helps School Secretaries with Extracurricular Activities • Maximizing Your Mini Memory's 4K of RAM • Exploring Enhanced BASIC on the Compact Computer • The LOGO Tortoise Debates the BASIC Hare • A Pocket Program to Organize Data with Linked Lists • Mentally Handicapped Learners Team Up with the TI-99/4A • The Wonders of Diskette Storage • Beeline—a Multi-Screen Strategy Game • Lost Ruins—an Archeological Dig Game • 3-D Illusions with Sprites in Depth • Game reviews, Group Grapevine, and much, much more.

JUNE 1983 (Partial Contents)
• Children and Computers Make the 99/4H Connection • Tune Your Guitar with Our TI Tuning Fork • Talk to Your Computer—Voice Technology Is Here • Gameware Buffet's Eat or Be Eaten Aardvark Game • Protect Your Station in the Space Zapper Game • What Multiplan Can and Can't Do • Understanding Inputs and Outputs in Drive For Diskettes—Part 2 • Calculate Loan Schedules on the CC-40 • Go on a LOGO Vacation • Letters on LOGO • A Review of Upper Room Software's Programs for Special Learners • Construct an RS232/Joystick Interface • Group Grapevine • Shopping Bus • A Natural Language Interface for the Professional • Game Reviews • and much, much more.

JULY 1983 (Partial Contents)
• The Evolution of Home Computer Graphics Comes Alive in Graphics Grows Up • Five Data Organizers in Never Out of Sorts • TI & 99'er at the Consumer Electronics Show • WarGames: The Movie and the Book • Editing with Multiplan • The LOGO Logician Presents To Model Is to Learn • LOGO Mosaic Designs Fill the Screen • Your Speech Synthesizer as a Spelling and Foreign Language Teacher • Software for Your Low-cost Printer Port • Gameware Buffet's Treasure Island and the Colorful Switch-A-Row • A Book Review of Learn BASIC for CC-40 Users • 3-D Animation with the TMS 9918A Video Chip • Games Reviews • Group Grapevine • and much, much more.

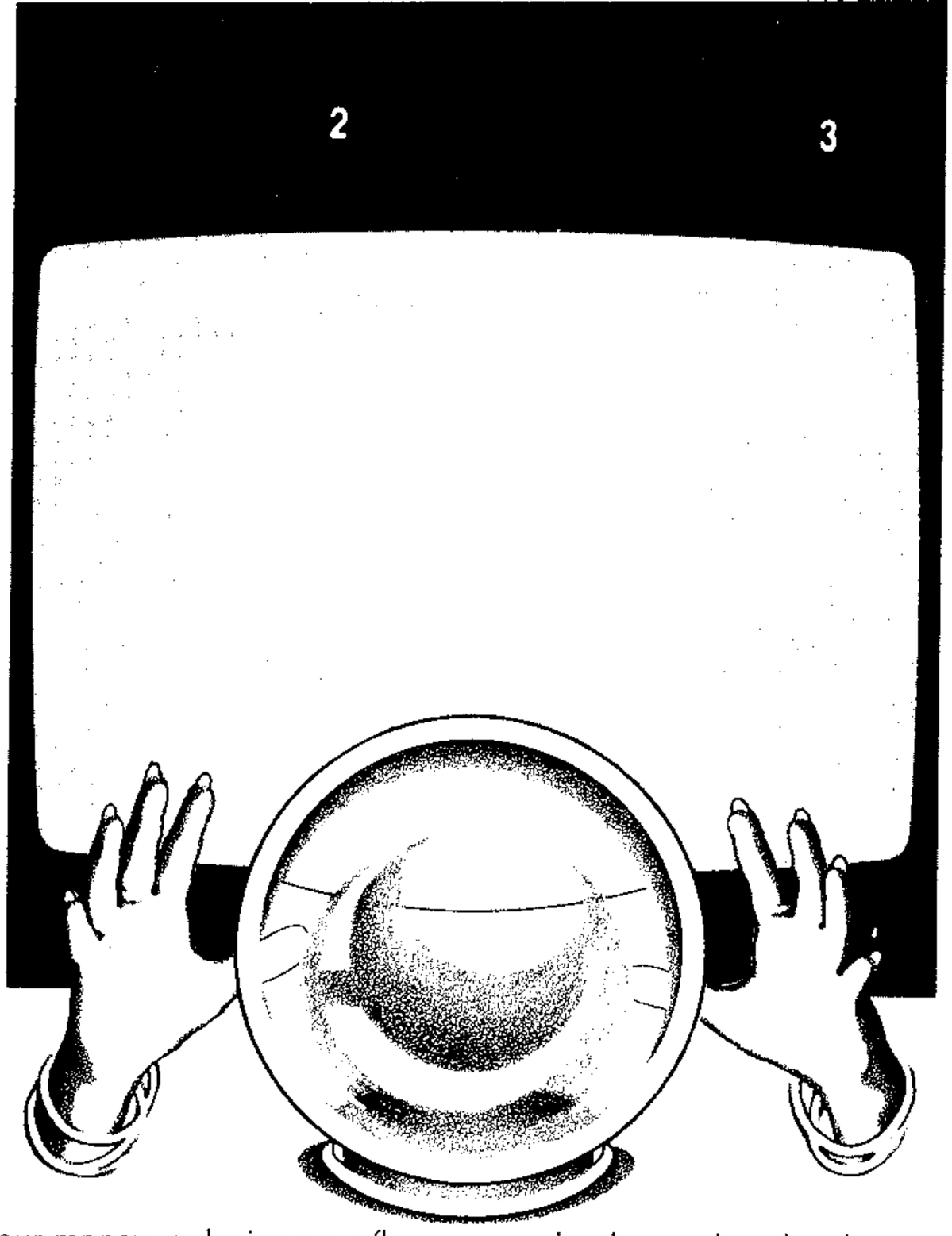
AUGUST, 1983 (Partial Contents)
• The Home Computer Goes To Work • Bit One, Purl Two at the Fashion Factory • Better Business Bar Graphs in Graphic Persuasion • An Ensemble of Assemblers • Cashflow Helps Money Management • Keystrokes for Thrifty Folks—a Review of Typwriter • Game Reviews of Cavern Quest and Starprobe '99 • Counting Fun for Preschoolers • Peripheral Vision 99 • Mean Machines and Small Potatoes • Multiplan Medium Groups Cells into Rectangles • Turtle Text, a LOGO Word Processor • Group Grapevine • Hello, Little Brother—CC-40 Speaks to 99/4A • Plato's Progress Looks at Student Assessment • Pocket Sunrise • Public Investigator • Gameware Buffet's Jungle Jim, Success Formula, and much, much more!

SEPTEMBER, 1983 (Partial Contents)
•Adding on to Your Home Computer • 99'er Directory of Commercially Available Software, Accessories and Peripherals • Peripheral Vision 99 Hardware Reviews • Pocket Sunrise, Part Two in Extended BASIC • Byte Lightning—Mini Memory Plays a Game • TI-WRITER Tutorial • LOGO's Functions, Sets and Turtles • The CC-40 and 99/4A Take the Data, and RUN • PLATO's Progress Looks at Geometry Courseware and the Shape of Things to Come • Gameware Buffet's Challenge of Camelot and The Fly • Game Reviews of Nit Wit and Crime and Punishment • Group Grapevine • 99'er Hall of Fame • 99'er Digest Update on New Products, and much, much more!

OCTOBER, 1983 (Partial Contents)
• Adventures and Fantasy with Your Home Computer • Do-It-Yourself Adventure • Picoprocessor Emulator, a 4-Bit Microprocessor • Have No Fear: Assembly Language Won't Byte, Part 1 • Make Your Mark • Les Izmore and Debug Computing Cartoon • Once Upon a Tortoise Shell—A LOGO Adventure • Turtle Strut • Number Nibbler for Children • Lots of Plots on Your CC-40 and HX1000 Four-Color Printer/Plotter • Multiplan Medium—Bartender • A Grisly Adventure Bear Hunt • Escape From Wizard's Keep in Extended BASIC • Game Reviews • Shopping Bus • and much, much more.

This column is an ongoing tutorial on the *Multiplan* software package. To obtain full benefits from this column, a newcomer to *Multiplan* may find it useful to read the previously printed columns.

One of *Multiplan's* more powerful features is its ability to retrieve information from inactive worksheets. The contents of inactive models already stored on disk can be included in the active worksheet (the one you're currently using). This feature greatly expands the usefulness of *Multiplan*, and this month's article will explore just one way in which the "eXternal Copy" command can be used.



Explanation of the Model

To understand this feature, we will look at an inventory control application. One of the reasons for inventory control is to maintain the ideal number of items in stock. Keeping too many items on hand ties up your money, reducing your flexibility; keeping too few means lost sales when you run out of items. An inventory control program can help you avoid these problems by telling you when and how many to order. The simple example presented here does just this for a very small inventory.

If we think of the model as a daily inventory worksheet, then we need to represent the previous state of the inventory and today's inventory activity. This model is therefore expressed in terms of yesterday and today. However, you could think of it in terms of last week and this week, or as any other time period that suits your needs.

Figure 1 shows the model. The first five columns reflect the state of the inventory at the time the model is first built. ITEM NUMBER is used because most people order stock by number, but ITEM NAME could be used just as well. NUMBER ON HAND is the actual count of each item in stock as of yesterday. ON ORDER shows the number already on order. STOCKING LEVEL is the minimum number to have on hand. This number is arrived at by experience. A business person considers sales history and the difficulty of obtaining each item in setting this number. REORDER QTY is the number to order when an order is placed. Again, experience dictates the proper number to use. Some items will be much less expensive when ordered, say, four at a time. It is possible to have your computer help you arrive at the stocking level and reorder quantity, but that is beyond the scope of this model.

Given the information in the first five columns, you'll need a place to enter inventory activity and a way to decide how many, if any, of each item to order. Inventory activity generally consists of sales and receivings. Sales represent items leaving the inventory, and receivings represent items being added to the inventory. Columns 6 and 7, SOLD TODAY and RECV'D TODAY, are the places where you will enter the day's activity. The information for these columns comes from your sales invoices and receivings documents.

The decisions about what to order will be made by *Multiplan*. The logic behind this is fairly simple. If we add the number on hand (column 2) to the number received today (column 7) and subtract the number sold today (column 6), we arrive at the number on hand right now (column 10). You might think that if this number on hand right now is less than the stocking level, then the item should be ordered; but it's not quite that simple. We must also take into account the number already on order so as not to order too many. Therefore, we should order an item if the number on hand right now plus the number already on order is less than the stocking level.

Multiplan must decide not only whether to order an item, but how many to order. This is where the reorder quantity (column 5) comes in; the number to order will be a multiple of the reorder quantity. In fact, it will be the lowest multiple which, when received, will bring the number on hand to at least the stocking level.

In Figure 1, column 8 is the number still on order. OUTST'G ORDERS is just the difference between the number on order yesterday (column 3) and the number received today (column 7). This is the number to be used in deciding how many more should be ordered. As already mentioned, column 10 contains the current number on hand. Column 9 is where the calculation of how many to order now is done. Assume for a minute that an item needs to be reordered. The number to order would be an integer multiple of the reorder quantity, so the problem comes down to what integer multiplier to use. For example, say that we have 1 on hand now, 2 already on order, and the item has a stocking level of 8 and a reorder quantity of 2. Clearly we need to order 6 more of this item; the 1 on hand plus the 2 on order is 5 below the stocking level, but we must order in multiples of 2. In this example, the integer multiplier to use is 3 (3 × reorder quantity 2 = 6 to order). To express this as a formula, we start with the difference between the stocking level and the sum of the number on hand right now plus the number already on order, and divide the whole thing by the reorder quantity. In this example, this would be $(8-1-2)/2=2.5$. The result, 2.5, represents the multiplier we seek, but it's not quite right because we need an integer value. We don't want to round the result; we want to make it come out to the next higher integer. To do this, we could add a decimal value which is very close to 1 to the result (I have used .999 in the model) and then

	1	2	3	4	5	6	7	8	9	10	11
1		----YESTERDAY---									
2	ITEM	NUMBER	ON	STOCKING	REORDER	SOLD	RECV'D	OUTST'G	ORDER	ON HAND	TOTAL
3	NUMBER	ON HAND	ORDER	LEVEL	QTY	TODAY	TODAY	ORDERS	NOW	TODAY	ON ORDER
4	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------
5	345AE2	10		8	4	5		0	4	5	4
6	455784	25		24	12	1		0	0	24	0
7	84R556Y	0	2	3	2		1	1	2	1	3
8	84RJF67	5		4	4	5		0	4	0	4
9	AB5868T	5	1	6	2	2	1	0	2	4	2

RC[−5]−RC[−1]

*MAX(0,INT((RC[−5]−RC[+1]−RC[−1])/RC[−4]+0.999)*RC[−4])*

RC[−8]−RC[−4]+RC[−3]

RC[−3]+RC[−2]

Figure 1: THE INVENTORY MODEL

truncate it to an integer via *Multiplan's* INT function. In the example, if we use the formula INT((8−1−2)/2+.999), we come out with INT(2.5+.999)=INT(3.499)=3, which is just what we want. We have only to multiply this by the reorder quantity of 2 to come out with the correct number to order now, which is 6.

Let's recap: The formula for the number to order now (assuming that items need to be reordered) is:

$$\text{INT}\left(\frac{\text{stock level}-\text{on hand}-\text{on order}}{\text{reorder qty}}+.999\right)*\text{reorder qty}.$$

In the example we've just considered, the item needs to be ordered. But what about an item which doesn't need to be ordered? If we apply the same formula to those types of items, we will come out with a negative number or 0. For example, take an item with 8 on hand and 0 on order, with a stocking level of 6 and a reorder quantity of 1. The formula above comes out to INT((6−8−0)/1+.999)*1 =INT(−2+.999)*1 = INT(−1.001)*1 = −2*1 = −2. In this case, we have too many in stock, and we should send some back! Of course, this is not what we'll do; *Multiplan* should simply tell us not to order any more of this item. To cope with overstocking expressed with a minus number, I have used the MAX func-

Building the Model

Here are the steps for building the model. Refer to Figure 1 for text, constant values, and the formulas.

1. Change the default column width to 10 using the Format Default Width command.
2. Change the Default Cell format to Integer and the Default Alignment to Centered via the Format Default Cells command.
3. Fill in the headings in rows 1 through 3 via the Alpha command. Note that the cells in R1C2 and R1C3 are continuous.
4. Put a line of dashes across row 4 by filling in R4C1 with 10 dashes via the Alpha command, and then Copying Right for 10 cells.
5. Fill in the information in rows 5 through 9, columns 1 through 7, as shown in Figure 1.
6. Enter the formulas in row 5, columns 8 through 11. *Figure 1* necessarily shows the formulas after they have been entered, but you need not type all that. Instead, use the cell pointer to pick out the columns you need.
7. Finish filling in the model by Copying cells R5C8:R5C11 Down 4 cells.
8. Assign the name TODAY to the area R5C10:R9C11 via the Name command. The reason for this will be explained shortly.
9. Fill in the setup string for your printer (if you have one) using the Printer Options command.
10. Save your finished model to disk via the Transfer Save command. I have used the filename INVEN.MP, but feel free to use any filename you like.
11. You might want to experiment with the model now. For example, see what happens when you increase the number SOLD TODAY of the second item to 2.

Index to Advertisers

*99'er Buyers Guide found only in subscription copies.

tion as part of the formula in column 9. Instead of using just the INT formula given above, the model asks for the larger of 0 or the result of the INT computation. Thus when the result of the INT computation is a minus integer, items which do not need to be ordered will show a 0 in column 9.

Column 11 of the model is the total number on order, assuming that the ORDER NOW recommendations in column 9 have been put into effect. TOTAL ON ORDER is just the sum of OUTST'G ORDERS and ORDER NOW.

How to Use the Model

To use the model for your own inventory, you will want to substitute your item numbers as well as your on hand, on order, stocking level, and reorder quantities. If you have more than five inventory items (which is very likely), use more rows and copy the formulas in columns 8 through 11 Down as needed. Don't forget to expand the definition of the name TODAY if you add more rows.

Save the revised model to disk. When you are ready to process the day's information, simply fill in the SOLD TODAY and RECV'D TODAY columns. *Multiplan* will tell you what to order now.

Extending the Usefulness of the Model

As presented so far, the model is great for reflecting today's activity, but what about tomorrow? The results shown in columns 10 and 11 should form the basis for tomorrow's work; that is, the figures in columns 10 and 11 should replace those in columns 2 and 3 in order to process the sales and receivings for the next day's business. Of course you could do this by hand, but *Multiplan* can do it for you.

The model INVEN.MP (or whatever you called it) was saved on disk with the current ON HAND and ON ORDER figures stored in the table named TODAY. You can use the *eXternal Copy* command to copy these numbers into columns 2 and 3 on your current worksheet.

To see how this works, first Transfer Load INVEN.MP if you do not already have the worksheet present on your screen. Then Blank out the cells R5C2:R9C3. This is necessary because *Multiplan* will not let you copy on top of cells containing data with the *eXternal* command. Don't be alarmed at the effect this has on columns 8 through 11; your worksheet is in transition right now. Next, use the *eXternal* Copy command to copy from sheet INVEN.MP on disk, copying from name TODAY to cells starting at R5C2. Be sure to set the linked parameter to No. You don't want to set up a permanent link between the active worksheet and the worksheet on disk; you simply want to obtain the values this one time. You will see columns 2 and 3 fill up with the data which was in columns 10 and 11 on the worksheet you saved. Now you can go ahead and fill in your SOLD TODAY and RECV'D TODAY columns to reflect your latest business, and *Multiplan* will make new recommendations about what to ORDER NOW. Don't forget to save the new worksheet on disk when you're done, under either the same name as before (if you are really finished with yesterday's information) or a new name if you want to preserve the old one for future reference.

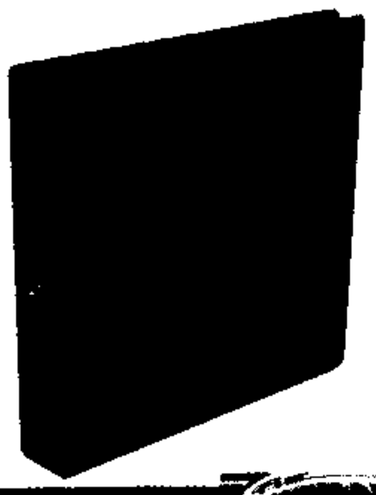
This technique can be used on any worksheet where the results of one session become the starting point of the next session. Since this situation is quite common, it's a handy trick to add to your *Multiplan* repertoire.

99'er-WARE™
P.O. Box 5537
Eugene, Oregon 97405
Tel. (503) 485-8796

ALL 99'er-ware PRODUCTS ON THIS PAGE MAY BE ORDERED USING THE BIND-IN CARDS NEAR THE REAR OF THIS MAGAZINE

Keep Your Magazines & Tapes Together

With a FINDER-BINDER™



- **Big enough to hold 6 magazines and 12 tapes**
- **Uses wire straps to hold magazines so that no hole punching is necessary**
- **Attractive and Durable**

—Only $11.95

(magazines and tapes not included)

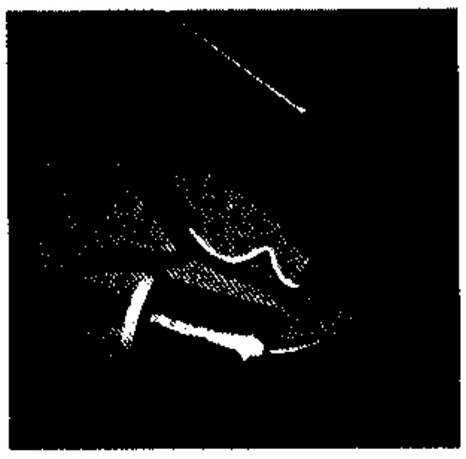
FREE 99'er Master Index with each 99'er Finder-Binder order (will be mailed when available in 3rd Quarter, 1983).

**Only $11.95 without cassettes, plus $3.00 shipping & handling. 6 High-Quality 99'er-ware C-10 Digital Computer Cassettes (with special BASE tape and 5-screw housing for data integrity). Available separately for $7.00 plus $2.00 shipping & handling.*

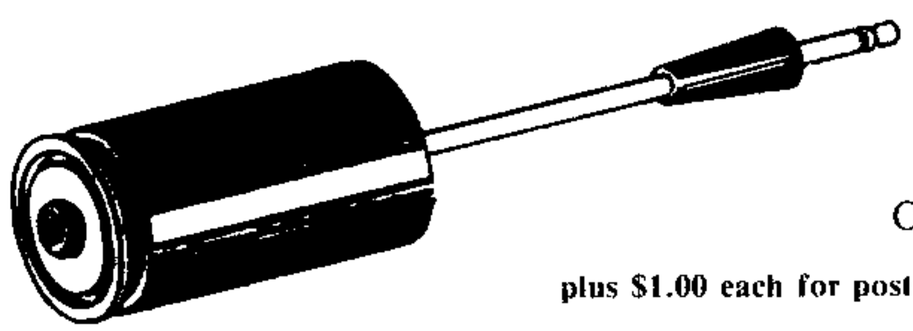
SPECIAL: 99'er Finder-Binder that is packed:
- **with 6 of the above blank cassettes for only $16.95**
- **with 12 of the above blank cassettes for only $21.95**

Add $4.00 shipping & handling to either order.

If The TI-99/4A Will Not Control Your Cassette Recorder Through Its Remote Jack, We Have The Solution For You . . .



The TEX-SETTE™ Adapter



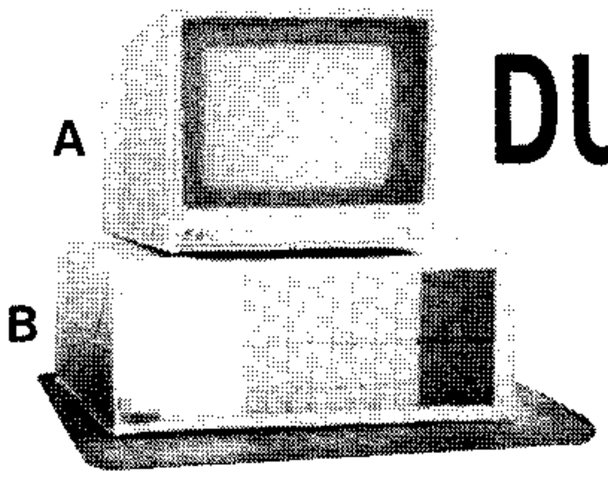
Low cost

Only **$5.95**

plus $1.00 each for postage and handling.

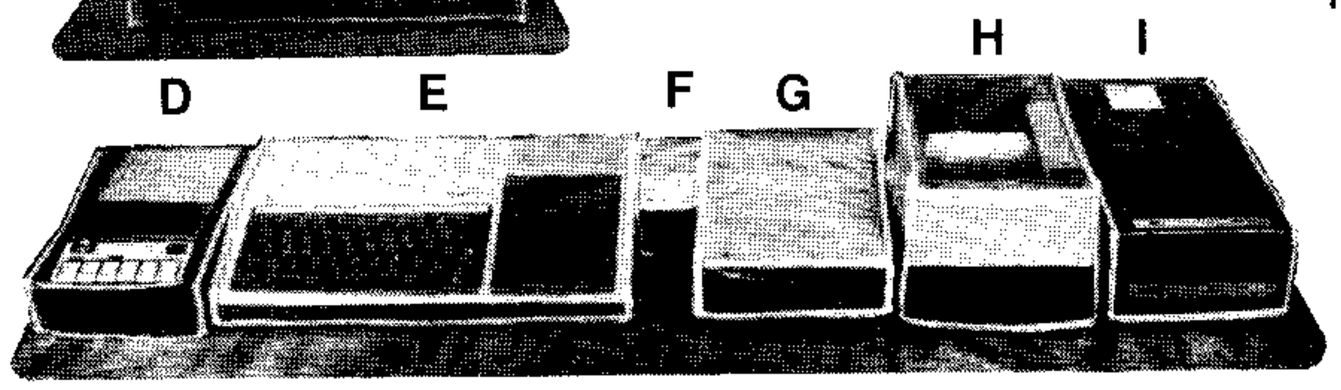
Cassette Compatibility At Last!

DUST COVERS



Features:

- Equipment Protection
- Handsome Appearance
- Custom-Fit
- Antistatic Treated
- Quality Construction



A	10″ Color Monitor Cover	$10.95
	13″ Color Monitor Cover (now shown)	$12.95
B	Peripheral Expansion Box Cover	$12.95
C	TI's 99/4 Matrix Printer Cover (same as Epson MX-80 Cover)	$9.95
D	Cassette Recorder Cover (1 size fits up to 10″ × 6″)	$4.95
E	TI-99/4(A) Console Cover	$8.95
F	Speech Synthesizer Cover	$3.95
G	Peripheral box Cover (Specify: 32K Memory Expansion, RS232 Interface, or Disk Controller)	$5.95
H	Thermal Printer Cover	$8.95
I	Disk Memory Drive Cover	$5.95

DEALER INQUIRIES INVITED

Add $2.00 shipping/handling for the first dustcover; 50 cents for each additional cover.

99'er GUIDE™ BOOKS

COMING SOON TO YOUR FAVORITE DEALER OR BOOKSELLER

This giant compendium of material drawn from the first publication year of *99'er Magazine* (original name) contains nearly 400 magazine-size pages chock full of valuable tips, tutorials, applications and "How-To" articles, plus 75 type-in-and-RUN computer programs in four languages (BASIC, Extended BASIC, LOGO, and Assembly) for the TI-99/4A Home Computer. The information, programming and attractive artwork contained within this first compendium have been completely revised and updated from the material as originally published. The collection also contains some new material never before published. This mammoth reference work is organized into eight comprehensive chapters—the first dealing with "Starting Out"; the next four with languages and programming; and the remaining three covering computer-assisted instruction, game-playing, plus applications and utilities. **$19.95**

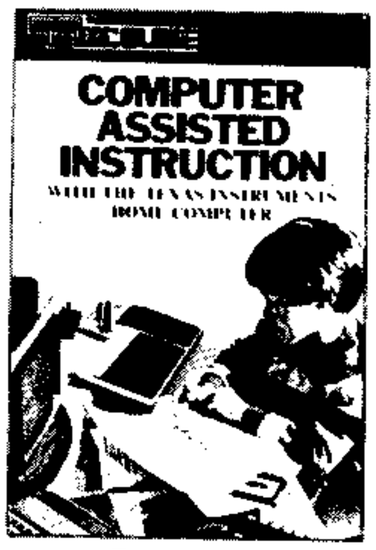
paperback, 8½ x 11; 376 pages + color plates

This newly-revised and updated selection of material drawn from the first two years of *99'er Home Computer Magazine* introduces families to the myriad of entertainment possibilities with their Texas Instruments 99/4A Home Computers. Fresh artwork and careful organization of the material make computer literacy both fun and easily attainable for family members of all ages. Neophytes will learn how to program sound effects and graphics and design their own games. Additionally, there are six complete game programs ready to be typed in and played. A unique "Programmer's Guide" appendix transforms this volume into a powerful learning tool for those who desire to go deeper into various aspects of programming.

paperback, 128 pages; 5½ x 8½ **$5.95**

This is the ideal companion to Volume 1 in the *Family Fun* series. It takes up where the first book leaves off—more tips, tutorials, attractive artwork, half a dozen game programs, plus an entertaining home bartending utility. The "Programmer's Guide" appendix adds to the learning aid material found in the first volume. **$5.95**

paperback, 128 pages; 5½ x 8½

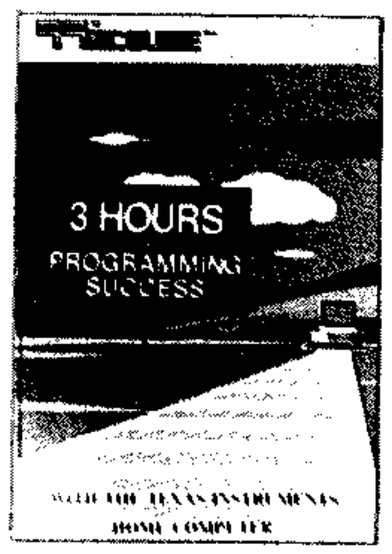


This newly-revised and updated selection of material drawn from the first two years of *99'er Home Computer Magazine* brings together in one handy volume a wealth of material for all those interested in the educational use of the Texas Instruments Home Computer. Readers are exposed to the exciting world of Computer-Assisted Instruction (CAI) where the TI-99/4A Home Computer serves as tutor, mentor, and ever-patient helper performing tasks that range from guiding preschool children in drawing block letters to using the extensive PLATO library. The book also contains ten complete CAI programs that are ready to be typed in and used. Included in this batch is the entire program listing, syntax, and user's guide to the ASPIC language—a LOGO-like language specifically designed for teaching purposes, that is itself written in TI BASIC. A unique "Programmer's Guide" appendix (see *Family Fun* books) is included. **$5.95**

paperback, 128 pages; 5½ x 8½

This book is the cure for "cyberphobia" (the fear of computers and computer-related activities) in new owners of the Texas Instruments Home Computer. After using "canned" software in the form of video game modules, home financial packages, and educational programs, most people become curious about computer programming. They want to know how to make screen graphics, generate computer music, and produce special color effects. When they try on their own, they often have problems with BASIC communications. This friendly, 99'er-Guide™ sweeps away the mystery and quickly shows new TI owners of all ages how to get single programs up and RUNing as fast as possible. After completing this entertaining 3-hour course, the reader will be pleasantly surprised to find that the computer has become a friend as well as a useful tool. Complete with comprehensive glossary, reference charts, and attractive illustrations throughout.

paperback, 128 pages, 5½ x 8½ **$5.95**



Selected Programs From This Publication Are Now Available on Cassette Tape

99'er magazine™ ON TAPE

- All purchasers of these packages are responsible for obtaining the individual documentation and program instructions in the indicated back issues of the magazine. When an issue goes out of print, copies of the program articles will be made available for purchase.
- If programs have been updated/enhanced since original publication, a printed copy of documentation changes will be provided with the tapes.
- Both subscriber & non-subscriber prices are indicated. To be eligible for the lower subscriber prices, you must be a current, paid magazine subscriber. All orders will be verified prior to shipment.
- Authors of these and future programs distributed on tape and disk will receive **Bonus Payments** based on the number of tapes sold; in fairness to them, please observe the Copyright laws, and report any incidents of "piracy" to our office.
- Recognized TI User Groups should contact us for special bulk terms & rates ----- No dealers, please.

Note: Programs in this issue will be available on a future tape release.

Use the bind-in card in the back of the magazine for your convenience in ordering.

Shipping—$1.00 for 1st tape, 50 cents each additional. Foreign Orders Shipped Airmail—$3.00 for 1st tape, 75 cents each additional.

PACKAGE #M2/5
From Vol. 2, No. 6 and Vol. 2, No. 7

Giant & Dwarfs (Extended BASIC)
Boa Alley (TI BASIC)
Word World (Extended BASIC)
Lost Ruins (TI BASIC)
Beeline (Extended BASIC)
M M Relocator (TI BASIC w Mini Memory)

$12/sub; $20/non-sub

PACKAGE #M2/2
From Vol. 2, No. 2

Termite (TI BASIC)
Tex-Scribe (Extended BASIC)
Gold Rush (Extended BASIC)
Plotting Routines (Assembly Object Code for Mini Memory Loader)

$12/sub; $20/non-sub

PACKAGE #M2/4
From Vol. 2, No. 4 and Vol. 2, No. 5

Lifeline to Titan (Extended BASIC)
Night Blockade (TI BASIC)
Quintus (TI BASIC)
Space Junket (Extended BASIC)
M M Disassembler (TI BASIC w Mini Memory)
Say & Spell (Extended BASIC)

$12/sub; $20/non-sub

PACKAGE #M2/1
From Vol. 2, No. 1

Up Scope (Extended BASIC)
Micro Jaws (Extended BASIC)
Knight's Tour (TI BASIC)
Screen Dump (Assembly Object Code for Mini Memory Loader)
ASPIC (TI BASIC)

$12/sub; $20/non-sub

PACKAGE #M2/3
From Vol. 2, No. 3

Close Encounters of the Simon Kind (TI BASIC)
Learning the Alphabet (TI BASIC)
Cyber-Dice (Extended BASIC)
Electrical Engineering Resistance (TI BASIC)

$12/sub; $20/non-sub

PACKAGE #M1/6
From Vol. 1, No. 6

Verbose (TI BASIC with Speech)
Color Mapping (TI BASIC)
County Fair Derby (TI BASIC)
Battle Star (Extended BASIC)
N-VADER (Extended BASIC)
Pre-School Block Letters (TI BASIC)

$12/sub; $20/non-sub

TI's Home Computer Software. The best way to bring the classroom home.

Keep the whole family learning with the Texas Instruments 99/4A Home Computer. Our library of exciting educational software programs goes from preschool through college and beyond. And no one offers a better educational software selection for traditional subjects or computer literacy.

Every subject in our library has been carefully developed with the help of leading educators: Scott, Foresman; Addison Wesley; Milliken; DLM; Scholastic; MECC and others. Many of their programs use synthesized speech to enhance learning.

And only the TI Home Computer has PLATO™ Basic Skills and High School Skills courseware. Until recently, only found in schools, PLATO teaches math, reading, social studies, grammar, science —more than 108 essential courses designed to give your kids an edge in education at home.

You can even teach yourself to write your own programs for your Home Computer, using any of six available languages: TI BASIC, TI Extended BASIC, TI LOGO, UCSD Pascal*; TI PILOT and TMS 9900 Assembly language.

If it's an education you want from your computer, the computer you want is from Texas Instruments.

For more information about our educational software or our broad selection of other software applications, call toll free: (800) 858-4565.

Creating useful products and services for you.

TEXAS INSTRUMENTS

TI LOGO: It opened a door they thought was locked.

It opened a door to their minds.

The key: a Texas Instruments Learning Computer and TI LOGO, a programming language developed by TI and MIT.

In his inner-city, New York junior high classroom, teacher Steve Siegelbaum explains why it works so well.

"When they use it, they think they're teaching the machine. In reality, it's teaching them how to learn. It definitely improves their attitude toward their other courses. Written and verbal expression improve–they're eager to show you, to tell you, what they've done."

Another teacher, Pete Rentof, adds, "What it fights is fear of failure – a mistake becomes a starting point. The whole learning process turns into a positive experience. It works."

The TI Learning Computer, with TI LOGO and many other educational programs, is equipped to help open doors in any classroom. Including yours. For information on this remarkable system, contact: Texas Instruments Customer Relations, P.O. Box 53 Lubbock, Texas 79408.

TEXAS INSTRUMENTS