A Hands-On Look At The Amiga 500

COMPUTE!

$3.00
October
1987
Issue 89
Vol. 9, No.10
$4.25 Canada
02193
ISSN 0194-357X

The Leading Magazine Of Home, Educational, And Recreational Computing

32-Bit Processing
How fast is fast enough?

Marbles
An exciting arcade-style game for the Atari, Commodore 64, Apple II, Amiga, and IBM PC/PCjr and compatibles

Super Turtle PILOT
A complete, flexible language for the 64 that includes powerful graphics, sound, and disk commands

Chrome
Double hi-res graphics commands for Applesoft

A BUTTON Command
A new, easy alternative to Amiga menus

Dazzlers
Brilliant graphics on the IBM PC/PCjr and compatibles

Double-Duty DOS
File compatibility with ProDOS and DOS 3.3

Smooth Out For Atari
Sharp graphics in mere seconds

Atari ST Menu Driver

Word War For The Amiga

32

0 71486 02193 3

From Origin comes the long-awaited sequel to the award-winning Ultima™ III

Ultima IV

Quest of the Avatar

Available on Apple®

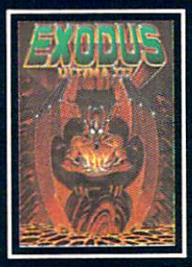
A state-of-the-art fantasy role-playing game of unprecedented magnitude by Lord British™.

Prepare yourself for a grand adventure: Ultima™ IV, sixteen times larger than Ultima III, is a milestone in computer gaming—one that challenges your physical and mental skills while testing the true fabric of your character.

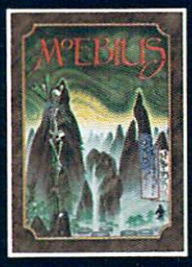
Enter Britannia, kingdom of Lord British. Journey through terrain of infinite proportions, conversing with characters on hundreds of topics. Unravel the mysteries of a superior magic system. At each turn beware of daemons, dragons and long-dead wizards haunting the most tranquil of places. Encounters with parties of mixed enemy types test your strategic abilities. Shrewd use of terrain can lead to victory against seemingly impossible odds.

Survive this multi-quest fantasy, then begin the final conflict, your quest of the Avatar. The ultimate challenge—the self—awaits....

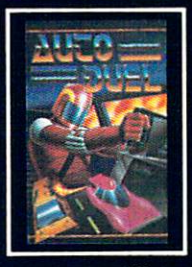
ORIGIN SYSTEMS INC. 136 HARVEY ROAD, LONDONDERRY, NH 03053 (603) 644-3360



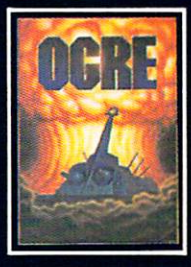
ULTIMA™ III sends you on an incredible fantasy role-playing journey through monster-plagued Sosaria in search of the elusive Exodus.



MOEBIUS™ takes you through the elemental planes of a colorful Oriental world of fantasy and adventure in search of the Orb of Celestial Harmony.



AUTODUEL™ is a futuristic, fast-paced strategy role-playing game where the right of way goes to the biggest guns.



OGRE™ is a strategy game fought on the nuclear battlefield of tomorrow as an inhuman juggernaut Cybertank battles conventional forces.

Ultima and Lord British are trademarks of Richard Garriott/Moebius is a trademark of Greg Malone/AutoDuel and Ogre are trademarks of Steve Jackson/Apple is a trademark of Apple Computer Inc/Previous Ultimas are not needed to enjoy Ultima IV **Authors wanted. Call us today.**

AIM HIGH

PRESENTING TECHNOLOGY THAT LETS YOU PROGRAM YOUR OWN DESTINY.

Electronics and computer equipment can be state-of-the-art today. And ready for the garbage can tomorrow.

Demand for technology is changing that fast.

And the people who can stay on top of it can write their own ticket in this world.

That's the beauty of the Air Force.

We can put you to work with technology that you may not read about in magazines for years.

We can teach you how to make sense of the most intimidating circuitry in existence anywhere.

If you attend college, we'll pick up 75% of your tuition. You can even earn an associate degree from the Community College of the Air Force.

What does all that mean?

It means there's no telling where technology is going in the future.

But with Air Force training, you've always got a future to look forward to. For more information, call an Air Force recruiter at 1-800-423-USAF.

AIR FORCE

* Suggested retail with TTL monochrome monitor, $1,295 with RGB color.

Finally, a complete, serious computing system for under $1,000.

☐ Introducing HeadStart—a top-quality IBM-XT compatible computer with all the important bells and whistles standard plus a $500 software package plus $1,000 in coupons good for computer goodies plus a full year's limited warranty plus a toll-free user hotline—all for only $995.*

☐ What's more, only HeadStart has the revolutionary HeadStart Operating Environment, designed to make even the most advanced user more productive instantly.

☐ For the HeadStart dealer nearest you call (800) 227-3900, in CA (800) 632-2122.

HEADSTART™

Vendex Pacific Inc., 40 Cutter Mill Road, Suite 438, Great Neck, NY 11021 (516) 482-4255

1. Unique HeadStart Operating Environment makes both novice and experienced user far more productive

2. Fully IBM-XT compatible with dual-speed 8 MHz 8088-2 processor, 512K RAM (expands to 768K)

3. 3-in-one CGA/Hercules/ monochrome graphics

4. Choice of high-quality RGB color or monochrome monitor with tilt/swivel base

5. Parallel, serial, mouse, light-pen and two game ports

6. Clock/calendar with battery backup

7. Dual 5¼" drives, 7 slots, hefty 135W power supply

8. Contemporary small-footprint design fits anywhere

9. Advanced AT-style sculptured 84-key keyboard

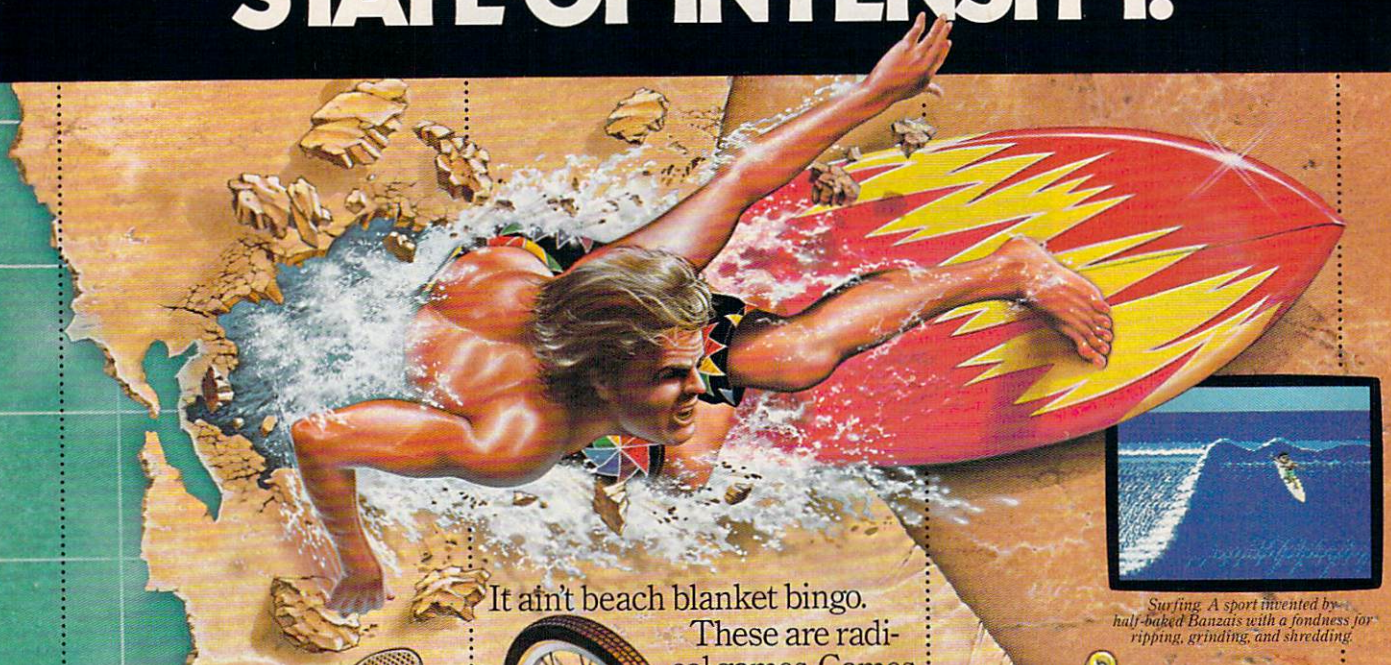
10. MS-DOS 3.2, GW BASIC included

11. $500 software package: award-winning interactive training, word processing, database, spreadsheet, plus pop-up programs

Full year limited warranty on parts and labor, toll-free user hotline

Extended warranty, on-site installation and service available

CALIFORNIA GAMES.™ BRACE YOURSELF FOR A NEW STATE OF INTENSITY.



Surfing. A sport invented by half-baked Banzais with a fondness for ripping, grinding, and shredding.

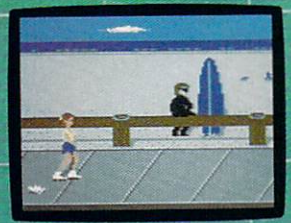
It ain't beach blanket bingo. These are radical games. Games that take the honorable tradition of *Summer Games™ I and II, Winter Games,™* and *World Games,™* wax it down and load it on the roof.

Try shredding the face of a totally tubular wave. Join the airforce in a BMX bike race. Screech around on skates and then rocket off the skateboard ramp.

You'll be playing for sponsors like Ocean Pacific, NHS Santa Cruz, CASIO, Costa Del Mar, Kawasaki, and Spinjammer. For trophies and an overall championship.

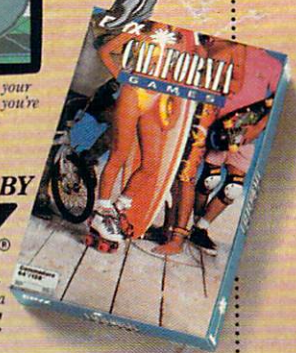
So get air. Go crazy. Welcome to the state of California.

BMX. Where wheelies are the breakfast of champions. And everyone else eats dirt.



Slalom the boardwalk obstacle course on roller skates.

Launch a few feet off-the-lip with your skateboard tucked high. And while you're up there, dance for the crowd.



CALIFORNIA GAMES BY

EPYX®

Commodore 64/128, Apple II & Compatibles, IBM & Compatibles, Amiga

SEE GAME BOX FOR DETAILS ON INSTANT WINNER CONTEST.

COMPUTE!

OCTOBER 1987
VOLUME 9
NUMBER 10
ISSUE 89

FEATURES

REVIEWS

COLUMNS AND DEPARTMENTS

THE JOURNAL

NOTE: See page 107 before typing in programs.

AP Apple, **GS** Apple IIGS **Mac** Macintosh, **AT** Atari, **ST** Atari ST, **AM** Amiga, **64** Commodore 64, **128** Commodore 128, **PC** IBM PC, **PCjr** IBM PCjr, • General Interest.

COMPUTE! Publications, Inc.
Part of ABC Consumer Magazines, Inc.
One of the ABC Publishing Companies
ABC Publishing, President, Robert G. Burton
1330 Avenue of the Americas, New York, New York 10019

COMPUTE! The Journal for Progressive Computing (USPS: 537250) is published monthly by COMPUTE! Publications, Inc., 825 7th Ave., New York, NY 10019 USA. Phone: (212) 265-8360. Editorial Offices are located at 324 West Wendover Avenue, Greensboro, NC 27408. Domestic Subscriptions: 12 issues, $24. POSTMASTER: Send address changes to: **COMPUTE!** Magazine, P.O. Box 10955, Des Moines, IA 50950. Second class postage paid at Greensboro, NC 27403 and additional mailing offices. ISSN 0194-357X.

Editor's Notes

MIT researcher Eric Drexler expects the Breakthrough in the next 20 years, give or take a decade or two. He says advances in molecular engineering and artificial intelligence will result in a Genie Machine. "What you ask for, it will produce."

> Eons of evolution and millennia of history have prepared this challenge and quietly presented it to our generation. The coming years will bring the greatest turning point in the history of life on Earth. To guide life and civilization through this transition is the great task of our time.
>
> If we succeed (and if you survive) then you may be honored with endless questions from pesky great-grandchildren: "What was it like when you were a kid, back before the Breakthrough?" and "What was it like growing *old*?" and "What did you think when you heard the breakthrough was coming?"

Drexler is not—in spite of the astonishing assertions in his book *Engines of Creation*—a mystic on the fringes of modern science. His ideas have the backing of noted scientists, including Marvin Minsky, widely acclaimed for his work in artificial intelligence at MIT.

The breakthrough described in Drexler's book is *nanotechnology*: the ability to rearrange atoms, creating molecules at will. Little machines the size of enzymes will work by the millions under the direction of computers the size of a pinpoint. In less than a day, using ordinary air and dirt, they could build a seamless rocket engine by rearranging atoms of mud. This engine would be made of the most ideal materials: carbon turned into diamond, aluminum oxide rearranged into sapphire.

This engine would be 90 percent lighter than contemporary rocket engines, able to repair itself during flight, able to rearrange its shape (different shapes are optimal at various points along a trajectory), lighter than wood, stronger than steel. And best of all, the only cost would be for the dirt and water.

Where do we get these minute robots and computers? They are, according to Drexler, the inevitable result of current progress in genetic engineering and artificial intelligence. This progress, he argues convincingly, does not depend on future scientific revolutions or undiscovered technologies. Instead, the development of this technology is proceeding rapidly and the few remaining barriers are more related to engineering than to scientific theory. You've doubtless heard that last year some scientists managed to cross tobacco with fireflies, creating, for some reason, plants that glowed in the dark.

It used to be that the best definition of life was "something that can reproduce itself." These molecular robots (Drexler calls them *replicators*) will destroy that definition. They can make anything, including copies of themselves. Once we create the first replicator, it will build its own offspring in about 15 minutes (using air or mud). Then the two of them will build two more children in the next 15 minutes. After about ten hours of this, there will be 68 billion of them working on their next generation. Obviously this sort of thing could get out of hand. After a couple of days they would outweigh the total mass of the solar system.

There is precedent for building huge things from molecular accretion: biological reproduction. This is, after all, how a whale sperm cell builds into a sperm whale. The biological building process is just slower by orders of magnitude than replicator building, in the same way that human thinking is enormously slower than computer thinking.

Drexler suggests inserting a loop counter which would shut replicators off after a certain number of generations. In fact, he spends the second half of his book exploring strategies to contain the unimaginable power of nanotechology.

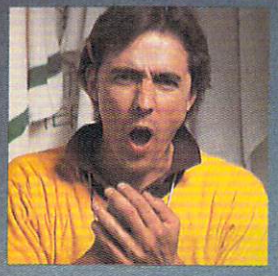
There is indeed a dark side to this power. When anything can be built for free, when artificial thinking is far deeper and faster than the human mind can fathom, when invisible machines can transform a solar system in days—we'd better be on the lookout. The first replicators could doom human muscle and mind to irrelevance. Important new techologies have always replaced a previously human activity. Electric light didn't eliminate candlemaking, but candlemaking *did* become relatively meaningless. Replicators—able to do anything within the limits set by the laws of physics—could supplant all human endeavor.

Engines of Creation makes a compelling case, saying replicators "promise to bring changes as profound as the industrial revolution, antibiotics, and nuclear weapons all rolled up in one massive breakthrough."

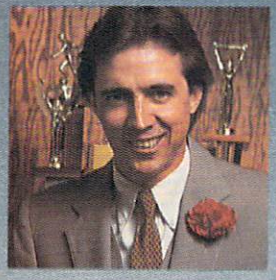
Richard Mansfield
Editorial Director

The best hockey and soccer for all of you.

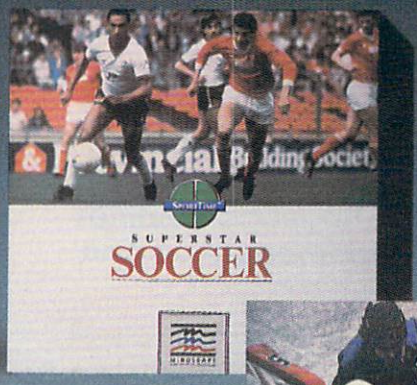
The Star.

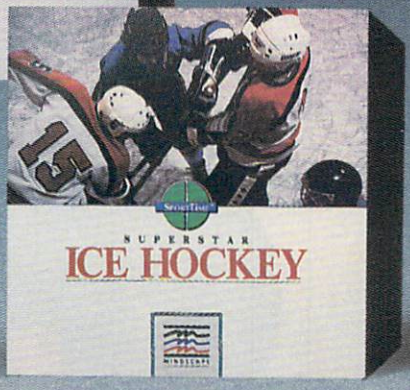


The Coach.

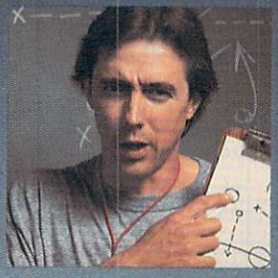


The Owner/G.M.





The Star.



The Coach.

The Owner/G.M.

Superstar Ice Hockey™ and Superstar Soccer™ by SportTime™ may be the best way of all to play sports because they bring out all your best sides.

You can be the star player, winning coach or even the wheeling and dealing owner/general manager in these amazing home games from Mindscape. On the ice, you can play center or goalie. On the bench, you can build or change the game plan. In the front office you can trade, draft or bring up new stars from your minor league club.

Watch the animated moves of all the players all the time.

Complete stats keep you posted on all the action.

Brilliant graphics make both simulations so real you'll expect to be interrupted by commercials. And a variety of play levels and time frames provide endless combinations for victory or defeat in your club's charge for the SportTime Cup.

All in all, these are the best sports games for all of you.

Mindscape sports lead the standings.

MINDSCAPE INC

Visit your retailer or call 1-800-221-9884 (in Illinois 1-800-942-7315) for VISA or MasterCard orders. To purchase by mail, send your card number with expiration date, check or money order for $34.95 (C64/128) or $39.95 (IBM PC®) for each game plus $3.00 for handling to Mindscape, Inc. P.O. Box 1167, Northbrook, IL 60065. Allow 3-5 weeks for delivery. Lawyers like this part: © 1987 Mindscape, Inc.

Publisher	James A. Casella
Editorial Director	Richard Mansfield
Managing Editor	Kathleen Martinek
Associate Publisher	Selby Bateman
Editor, COMPUTE! & COMPUTE!'s GAZETTE	Lance Elko
Assistant Editors, COMPUTE!	Rhett Anderson, Randy Thompson
Production Director	Tony Roberts
Editor, COMPUTE!'s Atari ST Disk & Magazine and COMPUTE!'s PC Magazine	Tom R. Halfhill
Editor, COMPUTE!s Apple Applications Magazine	Gregg Keizer
Features Editor	Keith Ferrell
Technical Editor	Ottis R. Cowper
Assistant Technical Editors	George Miller, Dale McBane, Jim Fuchs
Assistant Editor, COMPUTE!'s Atari ST Disk & Magazine	Todd Heimarck
Assistant Editor	John Shadle
Programming Supervisor	Patrick Parrish
Editorial Programmers	Tim Victor, Tim Midkiff, William Chin
Buyer's Guide Coordinator	Caroline Hanlon
Copy Editors	Karen Uhlendorf, Karen Siepak, Jill Champion
Submissions Reviewer	David Hensley
Programming Assistants	Troy Tucker, Joyce Sides
Executive Assistant	Debi Nash
Administrative Assistants	Julia Fleming, Iris Brooks, Sybil Agee
Receptionist	Anita Armfield
Associate Editors	Jim Butterfield, Toronto, Canada Fred D'Ignazio, Birmingham, AL David Thornburg, Los Altos, CA
Contributing Editor	Bill Wilkinson
COMPUTE!'s Book Division Editor	Stephen Levy
Assistant Editors	Tammie Taylor, Robert Bixby, Lynne Weatherman
Programming Assistant	David Florance
Director of National Sales	Joseph W. Hatcher
Production Manager	Irma Swain
Art Director	Janice R. Fary
Assistant Art Director	Lee Noel, Jr.
Assistant Production Manager	De Potter
Artists	Robin Case, Kim Potts, Scotty Billings, Tony Jacobson
Typesetting	Terry Cash, Carole Dunton
Illustrator	Harry Blair
Director of Advertising Sales	Peter Johnsmeyer
Advertising Director	Bernard J. Theobald, Jr.
Production Coordinator	Kathleen Hanlon
Customer Service Manager	Diane Longo
Dealer Sales Supervisor	Jose Cruz
Individual Order Supervisor	Cassandra Green

James A. Casella, President
Richard Mansfield, Vice President, Editorial Director
Richard J. Marino, Vice President, Advertising Sales
Ilene Berson Weiner, Vice President, Production

Editorial Board
Richard Mansfield, Kathleen Martinek, Selby Bateman, Lance Elko, Tom R. Halfhill, Stephen Levy
Robert Lock, Founder and Editorial Consultant

COMPUTE! Publications, Inc. publishes:

COMPUTE! ***COMPUTE! Books***
COMPUTE!'s Gazette
COMPUTE!'s Gazette Disk
COMPUTE!'s Apple Applications
COMPUTE!'s Atari ST Disk & Magazine

Editorial offices:	324 West Wendover Avenue Suite 200 Greensboro, NC 27408 USA
Corporate offices:	825 7th Avenue New York, NY 10019 212-265-8360
Customer Service: (Single Copy)	800-346-6767 (In NY 212-887-8525) 10:00 AM–12:30 PM; 1:30 PM–3:00 PM Monday–Friday
Dealer Sales:	800-638-3822 (In NY 212-887-8566) 9 AM–5 PM Monday–Friday

Coming In Future Issues

The Hermit: A Challenging Text Adventure For Commodore 64, IBM PC, Amiga, Atari, and Apple

Atari Screen Display Toggle

Masked Input For The Amiga

Chrome II For Apple

Graphics Aid For The 64

BASIC Programs Menu For IBM PC/PCjr

Machine Language Routines For The Commodore 64 And 128

Subscription Orders & Inquiries

COMPUTE!
P.O. Box 10954
Des Moines, IA 50340

TOLL FREE Subscription Order Line
1-800-727-6937

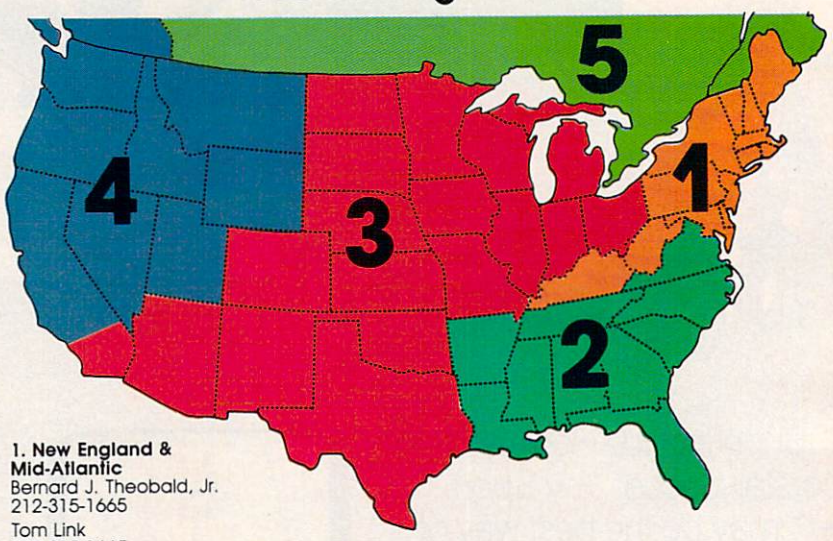
COMPUTE! Subscription Rates (12-Issue Year):

US		Canada & Foreign	
(one yr.)	$24	Surface Mail	$30
(two yrs.)	$45	Foreign Air	
(three yrs.)	$65	Delivery	$65

Member ABC Audit Bureau of Circulations

MPA Magazine Publishers Association

Advertising Sales



1. New England & Mid-Atlantic
Bernard J. Theobald, Jr.
212-315-1665
Tom Link
212-315-1665

2. Southeast & Foreign
Harry Blair
919-275-9809

3. Midwest & Southwest
Jerry Thompson
312-726-6047 (Chicago)
713-731-2605 (Texas)
303-595-9299 (Colorado)
415-348-8222 (California)
Lucille Dennis
415-348-8222

4. West, Northwest & British Columbia
Jerry Thompson
415-348-8222
Lucille Dennis
415-348-8222

5. Canada
Harry Blair
919-275-9809

Director of Advertising Sales:
Peter Johnsmeyer

Advertising Director:
Bernard J. Theobald, Jr.

COMPUTE! Sales Office 212-315-1665

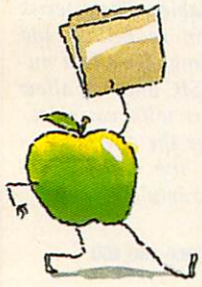
Address all advertising materials to:
Kathleen Hanlon
Advertising Production Coordinator
COMPUTE! Magazine
324 West Wendover Avenue
Suite 200
Greensboro, NC 27408

Now at Your Local Waldenbooks Store!

COMPUTE!'s new *Quick and Easy* series

puts the computer information you need right at your fingertips.

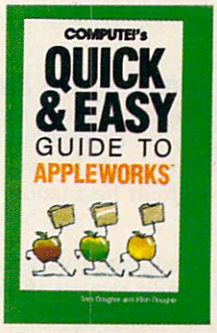
Concise, reasonably priced, hands-on user guides to the most popular and powerful computer applications. You'll find all the easy-to-follow information you need.
Look for these and other COMPUTE! books at your local Waldenbooks Store.

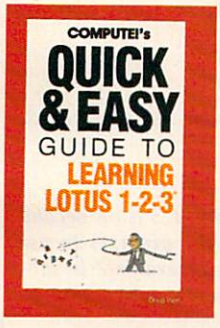


COMPUTE!'s Quick and Easy Guide to *AppleWorks*
Tom and Ellen Dougher
ISBN 0-87455-109-9 *$10.95*

AppleWorks, the unchallenged leader in personal productivity software for the Apple II series, is an integrated package combining a word processor, database, and spreadsheet. Precisely because *AppleWorks* can do so much, it can prove complicated even for the regular user. *COMPUTE!'s Quick and Easy Guide to* AppleWorks solves that problem, putting all the information you're likely to need on a day-to-day basis in an easy-to-use format.

Clear, complete guidance will benefit first-time and experienced users alike. You'll see how to get started with *AppleWorks*, and exactly how to get the most out of each of the three major applications. You'll quickly master the intricacies of this powerful program using the special Command Summary, Keyboard Template, and Quick-Reference Chart.





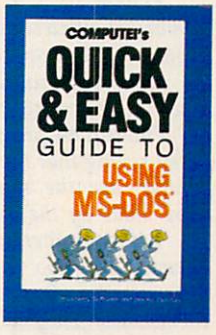
COMPUTE!'s Quick and Easy Guide to Learning *Lotus 1-2-3*
Doug Wolf
ISBN 0-87455-106-4 *$12.95*

Here's the perfect resource for the novice user of *1-2-3*. The book assumes no previous computer experience whatsoever. In fact, *COMPUTE!'s Quick and Easy Guide to Learning* Lotus 1-2-3 covers everything from building a spreadsheet model, setting up a database, and using *1-2-3* for word processing, to creating graphs and powerful macros. Also included are discussions on starting *Lotus 1-2-3* and selecting add-on software to enhance the program.

COMPUTE!'s Quick and Easy Guide to Using MS-DOS
Bonnie Derman and Strawberry Software
0-87455-105-6 *$12.95*

An easy-to-follow reference guide that every user of this powerful and flexible operating system will welcome. A must for new and experienced owners of IBM PCs or compatible computers.

No longer will you need to wade through hundreds of confusing pages in the manual to find just the right DOS command. *COMPUTE!'s Quick and Easy Guide to Using MS-DOS* makes the information clear and instantly accessible. With two indices and an expanded table of contents, simply look up the task you want to accomplish or the DOS command you need—and you'll find exactly what you're looking for.



America finds it at

Waldenbooks®

Books • Audio • Video • Magazines • Book Clubs • Special Orders
Over 1000 stores nationwide.

Check the yellow pages for the Waldenbooks store nearest you.

COMPUTE!™ Publications, Inc. abc
Part of ABC Consumer Magazines, Inc.
One of the ABC Publishing Companies

Readers' Feedback

The Editors and Readers of COMPUTE!

If you have any questions, comments, or suggestions you would like to see addressed in this column, write to "Readers' Feedback," COMPUTE!, P.O. Box 5406, Greensboro, NC 27403. Due to the volume of mail we receive, we regret that we cannot provide personal answers to technical questions.

Forward Two Steps, Back One

Rumors abound that the Amiga is or can be compatible with the Commodore 64. If it were, I would buy one without hestitation.

William F. DeBerg

I maintain a very firm belief that Atari should market an Atari 8-bit emulator for the ST.

Damon Milhem

Can I expand my VIC to run 64, 128, Amiga, or Atari ST programs?

Jeffrey M. Powers

Judging by the letters we receive, and by the questions and comments in the user group newsletters that we read, there is a great deal of interest in emulation—making one computer run programs written for another.

Emulation is a complex subject, so let's begin with a simple question that was popular a few years ago: Why can't my 64 run VIC programs? *(Some people asked the opposite question:* Why can't my VIC run 64 programs?*) They continue by saying that the two computers look similar, use the same peripherals, and were made by the same company.*

Although there are many differences between the two computers, the major difference lies deep within the computers themselves: They have different display hardware. The VIC used a video chip called the VIC; the 64 used a chip called the VIC-II. For a program to work on two computers, it must "see" the same hardware registers in the same memory locations.

With the popularity of the Amiga and the Atari ST, the question is coming up again: Why can't my new computer run my old programs? *And, again, the alternate question:* What can I add to my old computer to gain the benefits of the new?

Adding components to your old computer to gain the speed and graphics capabilities of the new computers is simply not feasible. To add the features of the Amiga to your 64, for example, you would have to replace the microprocessor, keyboard, RGB output, and sound and graphics chips. Indeed, a 64 has almost nothing that an Amiga could use. In essence, you would have to add a whole Amiga to your 64.

Others want their old software to run on their new computer. This approach is only slightly more promising. Programmers are working on an Atari 8-bit emulator for the ST and a Commodore 64 emulator for the Amiga. Unfortunately, this approach is bound to lead to lackluster results. The 16-bit 68000 microprocessor used in the ST and Amiga is simply not fast enough to emulate eight-bit computers at full speed. Although the 6502 is reasonably easy to emulate, duplicating the hardware of the the still-impressive 64 and Atari eight-bit computers would be a monumental task. The emulators we've seen achieve approximately 20–30 percent of the speed of the original computer. Even if you had perfect emulation, you would still need to transfer your programs across incompatible disk formats, or from program cartridges.

The best solution to the problem is this: If you want to run programs written for a specific computer, or if you want to write programs to take advantage of the power of a certain computer, buy that computer. Any other option will lead to frustration.

Variable Operators?

When I type in and execute this line:

```
10 X = 2 * 6
```

the variable X contains a value of 12. But when I type in these lines:

```
10 OP$ = "*"
20 X = 2 OP$ 6
```

I get a syntax error. How can I make it so that I can change the operators in my expressions?

Scott K. Stephen

Line 10 of your second example, OP$="*", is perfectly legal. Any character can be assigned to a string variable. Line 20 is the problem. BASIC is familiar with operators called *, /, +, −, and others, but it doesn't know of one called OP$.

BASIC has a set of rules that determines whether an expression is allowed. Although a BASIC-like language could be devised to allow your statement in line 20, BASIC does not. The numbers 2 and 6 in your example are numeric operands. They can be replaced by variables. The asterisk (used for multiplication in BASIC and most other computer languages) is a numeric operator, and BASIC does not allow you to replace operators with variables.

One way to change the operators in expressions is to use the ON-GOSUB command. Here's an example:

```
10 INPUT B,C,OP
20 ON OP GOSUB 100,200,300,400
30 PRINT A
40 END
100 A=B+C:RETURN
200 A=B-C:RETURN
300 A=B*C:RETURN
400 A=B/C:RETURN
```

When the program asks for input, respond like this:

```
4,5,1
```

The first operand is 4; 5 is the second operand; and 1 tells the program to use operator 1 (addition). Operator 2 is subtraction; operator 3 is multiplication; operator 4 is division.

What's A Blit?

I've heard people talk about a "blitter" in the Amiga. Just what is a blitter and why is it important?

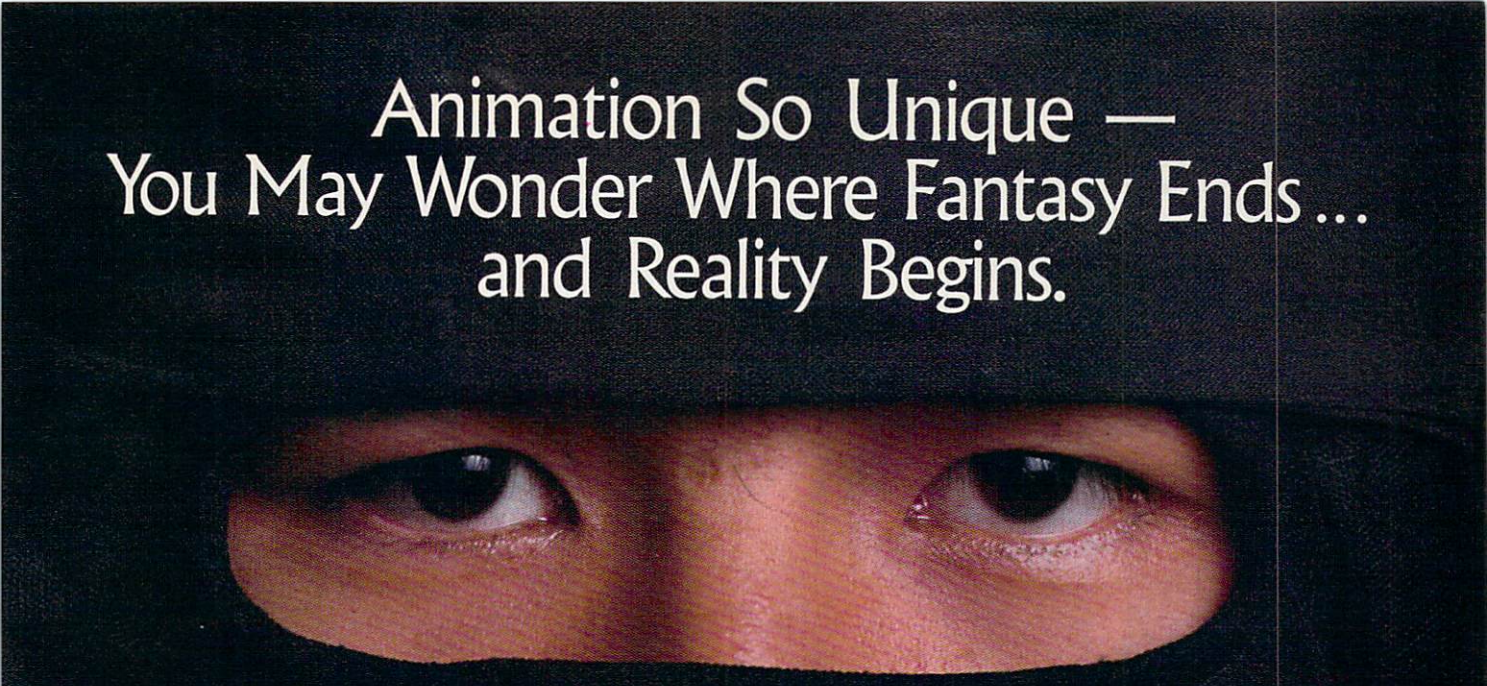
Jeff Murphy

The word blit *is short for bit-block image transfer, so a* blitter *is something which transfers bit-block images. All the images that are displayed are your screen are represented internally as rectangular groups of bits. The blitter's job is to move the collection of bits that make up a graphics image from one part of memory to another at high speed.*

Most computers rely on the microprocessor to perform these transfers. Unfortunately, this slows down the computer to a great degree any time an image needs to be moved. Since the Amiga's blitter operates independently of its 68000 processor, smooth animation is possible.

If your favorite language is C,

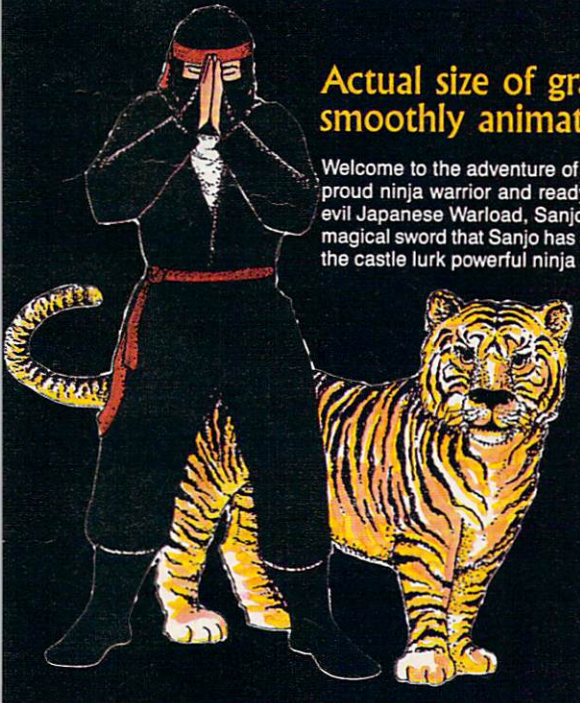
Animation So Unique — You May Wonder Where Fantasy Ends... and Reality Begins.



MASTER NINJA

SHADOW WARRIOR OF DEATH™

You become the "Master Ninja" on his quest to recover the Magical Sword in the Mysterious Emperor's Castle.



Actual size of graphics—large, smoothly animated 3½" characters.

Welcome to the adventure of a lifetime. Assume the identity of a proud ninja warrior and ready yourself to enter the castle of the evil Japanese Warload, Sanjo. Your quest is to regain a precious magical sword that Sanjo has stolen from your ninja sect. Through the castle lurk powerful ninja warriors, Samurai guards, evil mystic priests, and deadly tigers. You must work your way through the many chambers of Sanjo's castle, fight and kill his ninjas, battle his black magic curses, overcome his traps and obstacles and recapture the magic sword. No ninja has ever survived. Are you the one who can finally battle the dark power and bring truth and goodness back to the land? Are you the one who has no fear? Are you the one that will be called . . "MASTER NINJA"?

Paragon's "MASTER NINJA" is the most realistic martial arts adventure game available today.

Game features include:

- 20 Real-to-Life Martial Arts Moves
- Obstacles including Hidden Ninja's, Evil Curses of Black Magic, Hidden Traps, and Even Deadly Tigers
- Over 25 Combat Screens growing more and more difficult
- Historic Ninja Weapons including Knives, Blinding Powder, Throwing Stars, Blow Dart Guns, and Bow and Arrows
- A Historic Story That Unfolds Like an Actual Ninja Movie with Detail so Great That Even the Constellations in the Sky are Accurate
- Realistic Sound

For the distributor nearest you, call

Master Ninja "Hotline"

412-838-1166

Developed exclusively for the IBM PC,® PCXT,® PCAT® and IBM compatibles.

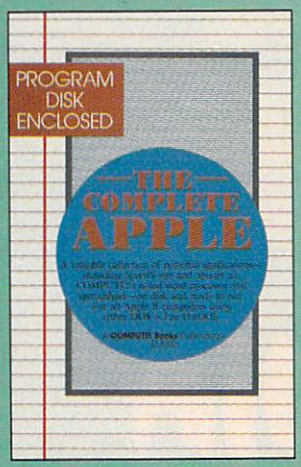
PARAGON

or write:

Paragon Software, Inc.
Plymouth Center
521 Plymouth Street
Greensburg, Pennsylvania 15601

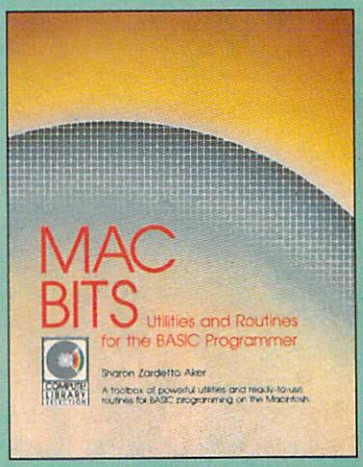
NEW COMPUTE! BOOKS FOR YOUR APPLE II AND MACINTOSH

Look to COMPUTE! Books for the best, most up-to-date information for your Apple II and Macintosh computers. You'll find clear guides to many commercial programs, tutorials to help you design your own programs, ideas for setting up a desktop publishing system, and powerful applications software.



COMPUTE!'s Quick and Easy Guide to *AppleWorks*™
Tom and Ellen Dougher
ISBN 0-87455-109-9 $10.95
The unchallenged leader in personal productivity software for the Apple II series, *Appleworks* is an integrated package combining a word processor, database, and spreadsheet. Precisely because *AppleWorks* can do so much, it can prove complicated even for a regular user. *COMPUTE!'s Quick and Easy Guide to AppleWorks* solves that problem, giving you instant access to all the information you're likely to need on a day-to-day basis.

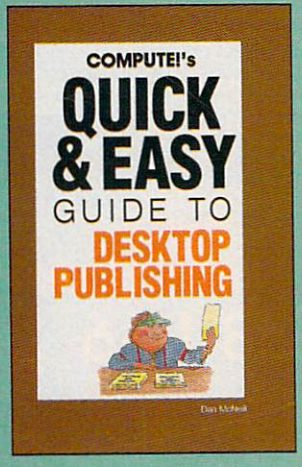
Clear, complete guidance will benefit first-time and experienced users alike. You'll see how to get started with *AppleWorks*, and exactly how to get the most out of each of the three major applications. You'll quickly master the intricacies of this powerful program using the special Command Summary, Keyboard Template, and Quick-Reference Chart.



Making Dollars and Cents with *Dollars and Sense*™
Gia Rozells and James Bartimo
ISBN 0-87455-101-3 $18.95
Take charge of your money with this guide to Monogram's *Dollars and Sense*, the bestselling personal and business financial-management program for the IBM PC and the Apple Macintosh. *Making Dollars and Cents with Dollars and Sense* takes you past the manual to fully exploit this efficient, versatile software. Graph income and expense, analyze cash flow, and see your real financial picture so you can make more informed decisions.

COMPUTE!'s Quick and Easy Guide to Desktop Publishing
Dan McNeill
ISBN 0-87455-112-9 $10.95
Desktop publishing is one of the fastest growing personal computer applications, making room for itself in businesses large and small. With the right computer, software, and printer, you can turn out documents, flyers, brochures, and more—all in the highest quality available outside of a professional print shop. This easy-to-use reference guide explains the ins and outs of desktop publishing, shows what you can do with the variety of available software, and even helps you select your system. If you're tired of wading through long and technical manuals, this guide and tutorial can get you started today with desktop publishing.

MacBits: Utilities and Routines for the BASIC Programmer
Sharon Zardetto Aker
ISBN 0-87455-075-0 $18.95
Book/disk combination
ISBN 0-87455-076-9 $29.95
A collection of valuable utilities and routines for the Macintosh with Microsoft BASIC. Utilities create windows, menus, and much more. Over 80 routines for functions such as refreshing the screen, scanning the keyboard, and keeping time. Information included on using the new libraries to enhance BASIC. *A disk is available which includes all the programs and most of the routines in the book, $15.95 (750BDSK).*

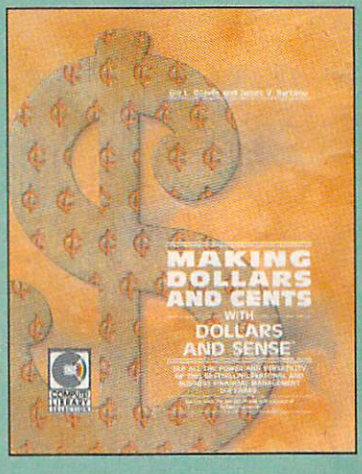


The Complete Apple
Edited
ISBN 0-87455-084-X Book/disk only $18.95
A book/disk combination with a series of powerful applications on disk. The book has complete, easy-to-follow documentation for each of the applications. Featured are *SpeedScript* and *SpeedCalc*, COMPUTE!'s professional-quality word processing and spreadsheet programs. All the applications are on disk ready to load and use on any Apple II computer.

COMPUTE! Books are at your local computer store or bookstore, or you can order directly by calling toll free 1-800-346-6767 (in NY 212-887-8525). Write COMPUTE! Books, Customer Service, P.O. Box 5038, F.D.R. Station, New York, NY 10150.

COMPUTE! books are available outside the United States from subsidiaries of McGraw-Hill International Book Company.

COMPUTE! Publications, Inc. abc
Part of ABC Consumer Magazines, Inc.
One of the ABC Publishing Companies



Please include $2.00 shipping and handling per book for U.S. and surface mail, $5.00 for airmail. NC residents add 5 percent sales tax, and NY residents add 8.25 percent sales tax. Please allow 4–6 weeks for delivery.

Modula-2, or machine language, you can access the blitter directly, creating blazing animation. If you program in Amiga Basic, you can still use the blitter. Amiga Basic has OBJECT commands which control bobs *(blitter objects).*

Right now, the Amiga is the only personal computer that has a blitter, but during the past year Atari has been showing one for its ST line of computers. As of this writing (early July), it is rumored that production problems may keep the blitter from being shipped in the new Mega ST line of computers. Presumably, the blitter will be made available as an upgrade to both the Mega series and the 520 and 1040STs. The older machines will also need new ROM chips to support the blitter.

Changing Chips

I enjoyed reading about the Winter Consumer Electronics Show (April '87 COMPUTE!). I was especially interested in the Mega ST line from Atari. By using 1 megabit chips, these computers provide more memory with less hardware, and they run cooler. Are chips like this available for the Commodore 128? It would be nice to be able to replace the chips in the 128 to give it greater memory capacity.

William Jones

The 128 is not designed to be expanded with 256K or 1 megabit RAM memory chips. The computer's memory management circuitry is designed specifically for the 64K chips currently installed. Furthermore, these chips are soldered in place, so you would have a difficult time removing them without damaging the circuit board. However, you can expand the memory of the 128—after a fashion—with the Commodore 1700 or 1750 expansion modules. These do not directly increase the amount of memory available to the processor, but the 128 provides routines so you can easily copy information to and from the modules.

Some computers do allow memory chips to be upgraded. The Macintosh SE from Apple, for instance, comes with 256K chips and is designed to accommodate 1 meg chips. This will increase the available memory of a Macintosh SE from 1 megabyte to 4 megabytes.

Another Drive

My Atari 130XE's owner's manual says that I can use up to four disk drives with my computer, but when I try to use more than two, I get an ERROR 160 message. Can I use four drives or not?

Lawrence Everett Van Buskirk

Since most people use one or two drives with their computer, Atari DOS is set up to use two. A memory location keeps track of which drives are allowed. The following table will help you determine the value to POKE into location 1802.

Drive	Value
1	1
2	2
3	4
4	8
5	16
6	32
7	64
8	128

On the 130XE, drive number 8 is the ramdisk. Add up all the values from the table above for the drives you are using. For example, if you want to use drives 1, 2, and 3, add 1 + 2 + 4, so you would use this statement:

```
POKE 1802, 7
```

Now, go to the DOS menu and use option H to write your modified DOS to disk. The next time you boot up with this disk, the new drive will be recognized.

Note that each drive you use will take away 128 bytes of your free RAM.

A Mouse In Applesoft

I recently purchased an Apple IIGS system. I am happy with it, but I cannot find out how to use the mouse in my own programs. I would appreciate any information you could give me.

Steve Green

The Apple IIGS and Apple IIc both have a built-in mouse port. (A mouse is also available for the Apple IIe.) The mouse is accessed through slot 4. At the beginning of your program, you should initialize the mouse with the following code:

```
PRINT CHR$(4) "PR#4"
PRINT CHR$(1)
PRINT CHR$(4) "PR#0"
```

This turns on the mouse and enables transparent mode, which sets up an interrupt routine to update the mouse position counters each time the mouse is moved. This is the only mouse mode that works with Applesoft BASIC.

After initializing the mouse, you must set up the mouse for input with the command IN#4. Now, the mouse can be read using a statement of the form

```
INPUT X,Y,ST
```

where the X variable returns a value in the range 0–1023, representing the horizontal coordinate of the mouse; the Y variable returns a value in the range 0–1023, representing the vertical coordinate of the mouse; and the ST variable returns a value representing the status of the mouse button. The button status value will be 1 if the mouse button is currently being pressed, 2 if it was just pressed, 3 if it was just released, and 4 if the button is currently released.

After you are through reading the

mouse, your program must execute an IN#0 statement so that the computer again looks to the keyboard for input.

The following BASIC program lets you draw on the high-resolution screen with a mouse. To exit the program, press the mouse button.

```
10 PRINT CHR$(4) "PR#4"
20 PRINT CHR$(1)
30 PRINT CHR$(4) "PR#0"
40 PRINT CHR$(4) "IN#4"
50 HGR : POKE -16302,0
60 INPUT X,Y,ST
70 X1 = INT(X / 3.65357143):Y1 = INT(Y
   / 5.35602094)
80 HPLOT X1,Y1
90 IF ST = 4 THEN GOTO 60
100 PRINT CHR$(4)"IN#0"
110 HOME : TEXT
```

Amiga Autoboot

Do you have any information about how to make an Amiga program boot itself if the Workbench is not already loaded? This would be useful to a programmer who doesn't want someone else to break into the program while it is loading.

Marc Singal

The answer is to be found in a file named startup-sequence, which is found in the s (sequence) subdirectory of the Workbench disk. If a file of this name is present, AmigaDOS automatically executes all the commands it contains before turning control over to the user. From the AmigaDOS CLI, you can use the TYPE command to examine this file, or ED to modify it. A normal startup-sequence file ends with these two commands:

```
loadwb
endcli > nil:
```

The first command tells AmigaDOS to load the Workbench, and the second terminates the startup sequence, closing the current CLI window and throwing away its output (directing its output to NIL:, the null device).

If the program you wish to run is one that can run from the Workbench or CLI, you can run it from startup-sequence simply by inserting its name before the endcli command. The program will run from startup-sequence just as if you had typed its name at the CLI prompt. For instance, if you want to run a program named myprog, you could edit the last lines of startup-sequence to look like this:

```
loadwb
myprog
endcli > nil:
```

AmigaDOS will still load the Workbench, but before turning control over to the user, it runs the program named myprog. If myprog terminates normally, the startup-sequence continues with the last line, which terminates the CLI and places you on the Workbench as usual. If you delete the endcli command from startup-sequence, you will find yourself at the CLI prompt when myprog terminates.

Of course, it's up to you, as the creator of myprog, to determine when, if ever, you want the user to escape the program. If the program is designed to terminate normally (for instance, when you are using it), but you want others to be barred from examining it, you could delete the loadwb command from the example start-up-sequence file. When you reboot with this file, AmigaDOS does not load the Workbench. When you terminate myprog, AmigaDOS executes the endcli command as before. But when you return to the Workbench, the screen is devoid of any disk icons and contains no menus. Without menus or icons of any sort, the user is precluded from doing anything except turning off the power or rebooting with another disk.

You also can run BASIC programs automatically when the computer boots up. To run the BASIC program named basicprog, for instance, you could edit the last lines of startup-sequence as shown here:

```
loadwb
amigabasic basicprog
endcli < nil:
```

When AmigaDOS executes this file, it loads and starts BASIC, which in turn automatically loads and runs the program basicprog. If you execute a SYSTEM command from BASIC, startup-sequence executes the endcli command and returns you to the Workbench as usual.

It's fairly simple to prevent someone from examining a BASIC program which runs automatically from startup-sequence. All you need to do is disable the normal BASIC menus and use ON BREAK to prevent the user from breaking out of the program with CTRL-C. The following program demonstrates the basic techniques.

```
' Turn off BASIC menus
FOR j=1 TO 4
  MENU j,0,0," "
NEXT J
' Enable break trap
ON BREAK GOSUB Gotcha
BREAK ON
'
Loop:
  x$=INKEY$
  IF UCASE$(x$) = "Q" THEN
       BREAK OFF
       MENU RESET
       STOP
  ENDIF
GOTO Loop
'
Gotcha:
PRINT "You can't break out that way...."
PRINT "Press Q to quit."
RETURN
```

The MENU commands cause all of the normal BASIC menus to disappear, preventing anyone from breaking out by selecting the STOP menu option. The ON BREAK GOSUB command tells BASIC to go to the designated subroutine whenever it detects the CTRL-C key combination, and the BREAK ON command activates the trap. At this point, it's impossible to break out by using the menus or the keyboard.

This program provides a "back door" command (the Q key) that allows the user to exit under controlled circumstances. In this case, we simply disable the break trap, turn the menus back on, and halt the program with STOP. There are a number of practical reasons, apart from secrecy, why you might want to prevent someone from breaking out of a program. A database program, for instance, might need to insure that all disk files are updated and closed properly before it relinquishes control. Custom screens and/or windows should also be closed when no longer needed, to avoid wasting the memory they consume.

If you don't provide a back door, of course, the user has no way to escape the program short of rebooting the computer. That method would be suitable for a case where you wish to allow someone to use the program, but not to examine it in any way. You could also allow an escape from BASIC, but return the user to a "dead" Workbench as described above. If you substitute SYSTEM for STOP in the BASIC example, you will exit BASIC and return to the startup-sequence file. If that file was edited to eliminate the LOADWB command, you will go back to a useless Workbench screen. ©

Nobody's Perfect...

Game	Winner	Time to Move
1	SARGON III	5 seconds
2	SARGON III	5 seconds
3	SARGON III	15 seconds
4	SARGON III	15 seconds
5	SARGON III	30 seconds
6	Chessmaster	30 seconds
7	SARGON III	3 min
8	SARGON III	3 min

YOU CAN'T WIN THEM ALL.
We pitted Sargon III, the world champion, against the challenger. Head to head, with no punches pulled. And believe it or not, the champ lost one game to the challenger. But Sargon III beats Chessmaster 2000 almost everytime.

Sargon III is the result of more than twenty one man years of development. Sargon III will play at precisely the level of difficulty you desire. Ask Sargon III for a hint, take back a move you've already made, replay a sequence, change sides, or set up your own board for customized learning. Sargon III lets you study in detail more than 100 great games of history or review 45 classic chess problems, all on screen. Sargon III also includes the largest library of opening moves available on any chess program.

SARGON III WRITES CHESS HISTORY
In a tournament sanctioned by the U.S. Chess Federation, Sargon III beat a master rated 2209. That makes Sargon III the only microcomputer program ever to defeat a player at that level.

Sargon III has no trouble wiping Chessmaster 2000 off the globe. Someday, maybe a chess playing program will be able to beat Sargon III. Until then, which chess program do you want to own: the world champion or the runner-up?

(Note: The test was run on two Apple II computers. Each program played once as white and once as black at each level. Send us $5.00 for handling, along with a postage paid return envelope and we'll send you a copy of the moves each side played.)

Sargon III is available for: Apple II series, Apple Macintosh, IBM PC and Commodore 64/128 computers. You can find Sargon III at your local retailer or call 1-800-826-0706. Illinois residents call 1-800-826-1330.
Only MasterCard, VISA and American Express accepted for phone orders.

HAYDEN
SOFTWARE
A division of Spinnaker Software

CLOSE UP:

The Amiga 500

Rhett Anderson, Assistant Editor

The Amiga 500 is poised to break into the home market. Here's a firsthand look at the newest version of this powerful machine from Commodore.

Judging by Commodore's resolve at the most recent COMDEX computer show, the company is convinced that it can make the Amiga into the latest Commodore success story. The company is so convinced, in fact, that it is bringing two machines to market. The Amiga 500 is aimed squarely at the home computer market previously dominated by the 64. The Amiga 2000, along with the PC compatibles that Commodore recently brought to the U.S., is part of the company's effort to secure a foothold in the lucrative business market.

The two new computers replace the Amiga 1000, which won rave reviews and sold respectably, but never gained the critical mass necessary to become a runaway hit. Some sources have claimed that fewer than 200,000 Amiga 1000s were ever made. For a company which has sold over six million 64s, that number was not enough.

Fortunately, the new Amigas retain the 1000's star quality that dazzled the critics. And both have acquired a few new tricks of their own. (For a closer look at the 2000, see the March 1987 issue of COMPUTE!.) Amiga's new two-fisted attack on the market should end the criticism that the Amiga was too expensive to be a home computer and lacked the software base to be a business computer.

How did Commodore manage to reduce the price of the computer so dramatically? By taking a close look at the market and at its computer.

More than any other feature, the price is the greatest difference between the Amiga 1000 and the Amiga 500. The original Amiga was initially offered at $1,295 without an RGB monitor. The computer came with 256K bytes of RAM, expandable to 512K with a relatively inexpensive add-on. Further expansion was available, but expensive.

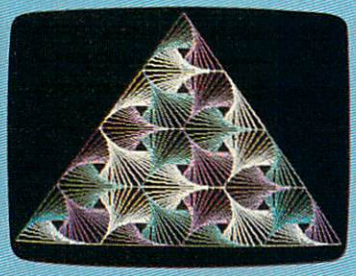
In contrast, a 500 with 512K RAM carries a list price of $595 without a monitor. The new price—plus a bit of advertising—may be the catalyst needed to make the Amiga a major player in the home-computer market.

The Inheritance

Rumors circulated months before the introduction of the machines speculating that the new Amigas would have graphics chips able to provide higher resolution, non-interlaced graphics. These chips are not present in either the 500 or the 2000. Reportedly, the Amiga engineers are still working on new dis-

GET UP TO 200 FUN-FILLED PROGRAMS EACH YEAR—

when you subscribe now to COMPUTE!



Subscribe to COMPUTE! today through this special introductory money-saving offer, and you'll be getting a lot more than just another computer magazine. That's because each issue of COMPUTE! comes complete with up to 20 all-new, action-packed programs.

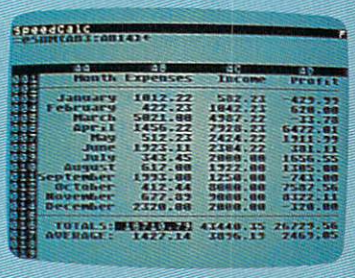
Subscribe now and you can depend on a steady supply of high quality, fun-filled programs like Hickory Dickory Dock, Switchbox, TurboDisk, Home Financial Calculator, Turbo Tape, SpeedScript, SpeedCalc, and hundreds of other educational, home finance, and game programs the entire family can use all year long.

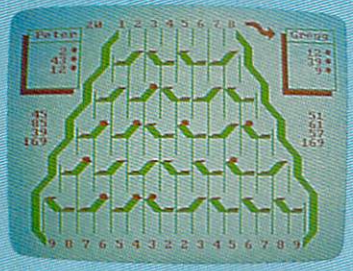
The superb programs you'll find in each issue are worth much, much more than the low subscription price.

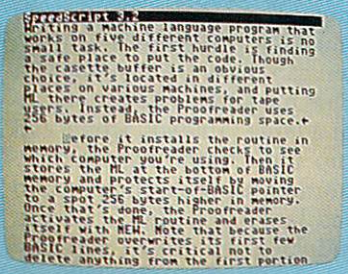
And there's more to COMPUTE! than just exciting new programs. Month after month, COMPUTE!'s superb articles deliver the latest inside word on everything from languages to interfaces...from programming to disk drives.

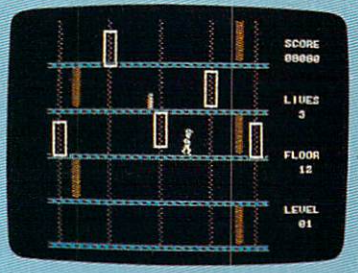
Whether you're a novice or an experienced user, COMPUTE! is the magazine for you. So subscribe today. Return the enclosed card or call 1-800-727-6937.

Do it now.

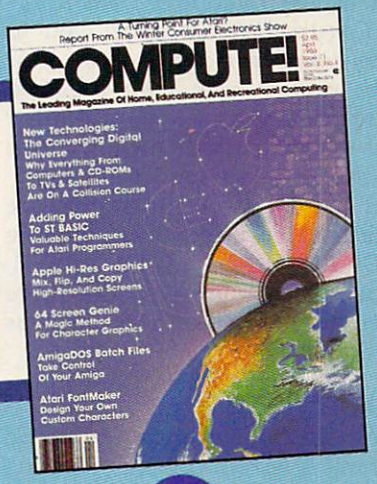








ACT NOW AND SAVE!



COMPUTE!™ Publications, Inc. abc

One of the ABC Publishing Companies

If attached order card is missing, write: COMPUTE! P.O. Box 10955, Des Moines, IA 50950

play technologies. Presumably, any new graphics chips will first appear as enhancements for the 2000 (and possibly the 500), and then as standard equipment in new computers.

As it stands now, the Amiga still has the most impressive graphics of any home computer. With resolutions of up to 640 x 400 pixels and 4096 colors to choose from, the Amiga is capable of stunning displays. One special screen mode even allows all 4096 colors to be displayed at once. Without special programming, 32 colors per screen is the limit.

The Amiga's powerful chips allow large amounts of graphics data to be moved and combined quickly to provide animation that is almost of cartoon quality. For the artistically inclined, the Amiga runs the most spectacular paint programs available.

Commodore recently announced that arcade manufacturers are beginning to use a modified Amiga computer as the engine of their latest arcade games. As part of the agreement, Commodore will market the software produced by the arcade-game publishers. For the first time, you'll be able to play exactly the same game at home that you can at the arcade.

The 500 also inherits the musical virtuosity of its older sibling. The Amiga has a dedicated sound chip with four voices. Two voices each are routed to the two phono-plug outputs for stereo sound. Like the Apple Macintosh, the Amiga is a natural for recording and playing back digitized sound. Programs such as Electronic Arts' *Deluxe Music Construction Set* allow you to write your own songs and hear them played back with digitized trumpets, pianos, flutes, and other instruments. For those who want to use the Amiga for more serious musical work, MIDI interfaces are available for under $50, offering MIDI IN, OUT, and THROUGH. With this interface, you can drive external music machines such as synthesizers.

Multitasking

Another of the Amiga's powerful features is its mulitasking operating system. The computer can run several programs simultaneously. While this may seem like a novelty at first, its usefulness quickly becomes apparent. For instance, suppose you're typing away on your word processor and you discover you don't have enough room on any disks to save your text. Simply pull up the Workbench screen and format a new disk—without ever leaving the word processor.

The Workbench is a mouse- and icon-oriented windowing system like that of the Macintosh. The CLI (Command Line Interface) lets you communicate more directly with the operating system. Using a CLI is similar to using the MS-DOS A> prompt. The most recent version of the Amiga operating system, version 1.2, is nearly bug-free, and it's faster and easier to use than previous versions.

The New, Low Price

In order to make the 500 more cost efficient than the 1000, several design changes had to be made. These changes affect the style and personality of the machine.

The most obvious of the changes is the computer's appearance. The Amiga 1000, with its detachable keyboard, and space atop the computer for a monitor, resembles an IBM PC system. The 500 looks more like an Atari 1040ST. The 500's keyboard is physically part of the computer housing, and there's no space on the computer for a monitor. To make up for its relative immobility, Commodore expanded the keyboard, matching that of the Amiga 2000. The new keyboard has an enhanced keypad, cursor keys in an inverse-T layout, and larger function keys.

In another major change, Commodore placed the Kickstart portion of the operating system into ROM. In the 1000, 256K of protected RAM known as the WCS (Writable Control Store) held the operating system, which had to be loaded from disk. Commodore had originally planned to place the operating system in ROM, but didn't feel the software was sufficiently debugged when the 1000 was released. Using the WCS gave the software designers time to fix bugs and improve the performance of the computer. Commodore has evidently decided that the 1.2 version of the operating system is good enough to put in ROM. This commitment should help convince developers that the computer is "finished."

For additional cost savings, new custom chips and gate arrays were designed to take over functions which were previously performed by more common, off-the-shelf chips. For example, the Agnes chip has grown into Fat Agnes by incorporating much of the support circuitry that previously surrounded the chip. More powerful chips mean less complex, easier to produce motherboards.

More RAM

The Amiga 500 comes with 512K bytes and is easily expandable to one megabyte (1024K). Although the system can be further expanded (up to nine megabytes), users looking for this much power would be better off with the Amiga 2000, which can be expanded to the same amount internally. Readers familiar with eight-bit computers may think that the 256K of RAM in the Amiga 1000 is plenty, but a great deal of the commercially available software for the machine requires at least 512K.

A large amount of memory is especially important in a multitasking computer like the Amiga. The more memory you have, the greater the number of programs and utilities you can have in memory at once. One megabyte of memory allows for a large ramdisk, which can speed up the system a great deal. Programmers using assemblers and compiled languages like C, Pascal, and Modula-2 will almost certainly want at least one megabyte of RAM.

The Manuals

The Amiga 500 comes with two books, *Introduction to the Commodore Amiga 500* and *Amiga Basic*. The first tells you everything you need to know about setting up your computer and using it for the first time. There are chapters on using the Workbench and the CLI, caring for your computer, and adding peripherals to the 500. The BASIC manual is a reference manual that assumes you already know BASIC. Those ready to move beyond BASIC will need the detailed technical reference books available from Commodore.

The Amiga And The Rest Of The World

In general, the 500 is as expandable as the 1000. However, some of the ports have been moved, and two of these have had their genders changed. Here's a quick look at how the 500 gets its power and talks to its peripherals.

The power supply, separate from the computer, contains the power switch. This makes it difficult to turn the machine off accidently. If you do, however, Commodore recommends waiting several seconds before turning it back on again. Resetting the computer by pressing the Control key in conjunction with the two logo keys is preferable to flipping the power switch.

Two game controller ports are located on the back of the computer. These are used to connect joysticks, mice, paddles, and drawing pads to the Amiga. Although the computer can use either analog joysticks (which are standard for the IBM and Apple) and digital joysticks (which are standard for the 64 and Atari computers), the latter have effectively become the Amiga standard.

The 500 has the same RGB monitor output as the 1000. However, unlike the 1000, the 500 cannot use composite color monitors or televisions. It does have a new composite monochrome output which could be used to connect a high-persistance monochrome monitor for high-resolution word processing and graphics.

The genders of RS-232 serial interface and Centronics parallel printer interface have been changed to make them compatible with cables designed for the IBM PC.

The expansion port has moved from the right side of the computer to the left. Although the 500's port is electrically identical to the 1000's, some devices will not connect to the 500. However, it won't be long before manufacturers begin to consider the 500 when they design new products.

One 3½-inch disk drive is built into the 500, and another can be connected to the external disk connector located on the back of the machine. The drives are double-sided and store 880K on each disk.

The Future Of Commodore

The Amiga 500 casing features the embossed name *Commodore*. It also uses a Commodore logo key in place of the left Amiga key of the 1000. Its new sister, the Amiga 2000, lacks any trace of the Commodore name. Apparently, Commodore expects to sell the 500 to the people most familiar with the Commodore name—those who bought the VIC, 64, and 128. The Amiga 2000 is aimed at another market.

The new Amigas are an impressive pair of computers. Home users will appreciate the 500's graphics, sound, and programming power. For the business community, the 2000 brings dazzling graphics and superb sound to the staccato of the MS-DOS world. The combination of developer enthusiasm and market excitement is already resulting in the appearance of innovative and powerful Amiga software and hardware products. If Commodore does everything right, it might just have another 64 on its hands. ©

32-Bit Processing:
Get Ready For The Next Wave

Keith Ferrell, Features Editor

Fast, powerful 32-bit computers are having a big impact on the business market—and it won't be long before that impact hits home as well.

Horsepower.

It's an odd, almost archaic word to apply to the world of computers, but it's the word nearly everybody's using to describe the new 32-bit microcomputers that are setting speed and productivity records on virtually every application. PCs built around the Intel 80386 chip, and micros based on the Motorola 68000 series of chips, are being extolled as having all the horsepower even the most dedicated power user needs to achieve maximum computer speed and efficiency.

What does all that horsepower enable a computer to do? Combined with the right hardware and software, a 32-bit computer can access *gigabytes* of RAM, move huge amounts of data, or accomplish mathematical calculations at blinding speed.

The result is a computer able to complete spreadsheet calculations more quickly, deliver smoother and more detailed graphics and animation, spellcheck a manuscript in seconds, manage artificial intelligence applications of increasing sophistication and complexity, deliver digitized sound, run several programs at once in a process called *multitasking*, and achieve other applications we have not yet imagined.

Essentially, we are seeing the emergence of microcomputers able to do things which previously required expensive minicomputers or even mainframes.

And all of it is a result of the latest strides in microprocessor technology. Chips can now work with 32 bits of data at a time.

Bit By Bit

At the heart of any microcomputer is its microprocessor, the *central processing unit* (CPU). The microprocessor serves as the computer's regulating device, coordinating the instructions of the operating system, data, and programs among the memory, storage, input/output, and peripheral devices of a computer system.

It is the CPU that processes, or moves, data. The data and software—whether a game, a word processor and text files, spreadsheets with cells and columns of numbers, or anything else—must first be broken into manageable units of information. Each of these units is called a *byte*, and is composed of eight binary digits, or *bits*.

The smallest unit of computer memory is the *bit*. Eight bits ganged together are called a *byte*. Four-bit microprocessors can work with four bits of data at a time, eight-bit machines can move eight bits (a full byte), and so on. Put simply, the more bytes that the microprocessor can access, move, and manipulate at one time, the faster the results.

"In general," says Greg Riker, director of Compact Disc Interactive (CDI) technology for Electronic Arts, "the 32-bit machines have the ability to address far larger address space, which means that applications can be much bigger and can have huge amounts of RAM attached to the machine. And this allows programs to run far faster and get to a lot more memory a lot more quickly."

Business Before Pleasure

So far, the impact of the 32-bit machines has been felt most dramatically in business, reflecting a trend that's been dominant since the arrival of the second generation of eight-bit machines early in the 1980s. The driving force behind the first wave of eight-bit computers was the home user, whose enthusiasm for machines, including the Commodore 64, the Apple II, and the Atari eight-bit series, did so much to convince hardware manufacturers that there was, indeed, a market for desktop computers.

Those early enthusiasts carried their very contagious commitment to computers into the workplace, which responded with excitement of its own once eight-bit computers capable of running sophisticated spreadsheets, word processors, databases, and other applications arrived. To be precise, those machines—IBM PCs and their offspring clones and compatibles—are considered 8/16-bit machines: They're able to address eight bits at a time and process 16 bits at a time. That 16-bit processing capability made it possible for PCs to begin to demonstrate speeds and capabilities that, however distant, at least approached those of minicomputers.

With the business market established and growing through the early 1980s, processor manufacturers such as Intel and Motorola, along with companies such as Microsoft (which developed the operating system that runs the IBM PCs and clones), had the capital to apply intensive research and development energy to extending the capabilities of 8/16 computers, and to the development of the next gener-

ation of processors—true 16-bit machines.

Sweet Sixteen

Eight-bit machines, such as the IBM PC and its offspring clones and compatibles, brought word processing, spreadsheets, and databases to every large corporation and to many, if not most, medium and small businesses. With PCs selling in the millions, an impetus existed for the development of the next generation of microprocessors: the true 16-bit chip, such as Intel Corporation's 80286.

If the eight-bit machines elevated computer speed to a level where big business could do something with it, the 16-bit machines delivered speed and capabilities that were nothing less than stunning.

"The 286 now constitutes that mainstream processor of choice for most business," says Bob Beach, manager of corporate relations for Compaq Computer Corporation, a leading manufacturer of business-oriented microcomputers, and the company that introduced the first production-model 32-bit computer.

Just a couple of years old, the 286 machines are being eclipsed by their 32-bit descendants. Does the arrival of 32-bit processing spell the end of 16-bit machines? Not at all, says Beach.

"The 286 is still very much alive. There's still a lot of value left in 286 machines and, for that matter, the eight-bit PCs. As one user upgrades to a 32-bit machine, his 286 machine tends to be passed along to someone else."

Recently we've seen the introduction of 16/32-bit computers, such as the Macintosh, Amiga, and Atari ST, which attempt to offer the best of both worlds. But it is the true 32-bit machine that has caught the attention of the business community.

1987 = 386

Intel's 32-bit chip is the 80386, and excitement gathered around it from the moment it was introduced.

"1987 has been a year in which the 386 machines have gotten a great deal of attention," says Bob Beach. "And I think 1988 will definitely be the year of the 386 processor."

Compaq led the industry with the introduction of a 32-bit machine, putting the Compaq Deskpro 386 on the market in September of 1986—to much market enthusiasm, but also to some skepticism.

"One of the things that we found most interesting when we introduced our 386," Beach says, "was that a lot of articles were published asking who really needed that much power. And a lot of the answers were *nobody*. The 286, people felt, had plenty of speed and memory for the typical user."

32-bit processors... will definitely find their way into entertainment machines.

The success of the Compaq 386—more than 10,000 were sold in the last quarter of 1986—and the quick entry of other compatibles manufacturers, such as PC's Limited (now Dell Computer Corporation), into the 386 market proved the skeptics were wrong.

"Obviously," Beach notes, "there are plenty of people out there who want this speed and power. We have yet to see a level point in the PC industry where users have said, 'This is it. I don't need any more speed, I don't need any more storage.'"

Is there such a point? "I don't have any idea where such a level point is," Beach says, "if it exists at all."

Out Of The Cradle

With 386 machines on the market and in the hands of consumers, the question becomes: How long will it be before we are fully able to utilize their capabilities?

Beach suggests it's too early to say. "Thirty-two-bit processing is still in its infancy. For that matter, the 286 is just now starting to leave its teenage years and enter early adulthood."

For now, owners of 386-based systems are limited to using their machines to run existing IBM-compatible software, albeit at far faster speeds than possible on 286 machines. The current PC operating system, version 3 of Microsoft's MS-DOS, can access only 640K of RAM at a time, and is limited to mass storage devices of 32 megabytes or less. It won't be until Microsoft's new OS/2 operating system is introduced that the full power of the 386 will be unleashed.

The widely-touted OS/2 is still in development at Microsoft, with release expected next year. Simultaneously, Microsoft is working on an extension to DOS 3 that will be able to handle larger amounts of RAM as well as accessing larger storage devices.

"There are two markets [for 386 software]," Bob Beach says, "and it's important to bear both operating systems in mind. Both of them will be important to computer users, depending on the individual user's needs."

The Price Of Heading Home

So far, 32-bit machines have remained high-end items mostly affordable to large businesses. Computer prices, though, have displayed a tendency to decrease with success. Will 32-bit technology touch the home computer and entertainment market in the next few years?

"Definitely," says Electronic Arts's Riker. He points out that while the home user is not likely to purchase a high-end 32-bit machine—such as the Compaq Deskpro 386, PC's Limited 386[16], or IBM's new Personal System/2 Model 80—other companies are already bringing 32-bit machines home.

"In general, the 32-bit processors of both the Intel [386] family and the Motorola 68000 family will, as their costs comes down, find their way into computers intended for the home."

Tandy/Radio Shack celebrated its tenth anniversary in the computer business with the early August introduction of its own 386 machine, the Tandy 4000. Pushing the price for an 80386-based com-

puter down to consumer levels is among Tandy's goals for the new machine, says Ed Juge, the company's director of market planning.

While the Tandy 4000 is capable of running DOS 3 and OS/2, as well as sophisticated business operating systems, such as UNIX, the machine also has a place in the home.

"All the horsepower is great for spreadsheets, databases, and other business applications," Juge says. "But what's even more exciting is the fact that with all that horsepower, you can move the computer a lot closer to consumers who've so far hesitated about getting one."

Because a 32-bit machine can access so much memory so quickly, its operation can be made much more transparent than that of other, less powerful microcomputers. Transparency options like pull-down, "point-and-shoot" operating menus, and simple English interactive instructions consume a lot of memory, but a computer with a memory potential measured in gigabytes can afford to be transparent.

"The power of a 386 machine," Juge says, "means that you can build in all sorts of conveniences that can make the computer a lot less scary to a non-technical consumer."

Electronic Arts's Riker also views the path from business to home as inevitable. "Initially, new hardware gets sold into business," he explains. "For business purposes, it makes sense to make investments in high performance machines.

"The home user tends to piggyback on whatever cost reductions have been created by volume use in other markets. For example, the Atari 2600 had a [Motorola] 6502 chip in it, because the costs of that chip had been amortized in business. Atari was able to get the chips at an affordable price."

The arrival of 32-bit home computers is only a matter of time. "If you look at the curve of the introduction price versus the price it bottoms out at," Riker says, "32-bit processors have not yet come down that curve. When they do, they'll definitely find their way into entertainment machines."

Heaps Of Horsepower

How will 32-bit processing affect the nature of entertainment software?

Riker points out that while the chip brings many capabilities, other factors enter the picture as well. "It's not just the processor," he points out, "it's the overall horsepower of the system. A 32-bit machine implies a faster processor that's able to execute many more instructions per second."

"So for one thing, you have a higher performance, and that translates into better graphics on the screen, better audio accompaniment, and a more realistic simulation of whatever the particular product is trying to simulate."

The extra horsepower can be put to different tasks. "Instead of 4 colors or 16 colors on the screen," Riker says, "we're up to 256 colors, or even more. And instead of beeps and bloops for sound effects, we're using digitized waveforms of real sound and music, as on the Amiga."

Commodore's Amiga is a 16/32-bit machine, as is the Atari ST and Apple's Macintosh. "The Amiga," Riker suggests, "offers a real good clue as to what entertainment machines will be like in the years ahead."

Riker also implies that 16/32 is a bit of a misnomer. "The Amiga, for example, is actually a 24-bit machine in terms of address space, a 32-bit machine in terms of the way data is manipulated internally, and a 16-bit machine in terms of the way it reads and writes memory."

How do we know what type of machine it really is? "Basically, the threshold, as far as I'm concerned, is whether or not the processor can do mathematical operations on 32-bit quantities. In that case, the Amiga, the Mac, and the ST are 32-bit machines."

That's Entertainment

Robert Lindsey, director of creative development at Epyx, thinks it will still be some time before 32-bit machines exert a dramatic effect on the home market. "386 machines were announced just a little over a year ago," he says, "so there hasn't really been time for them to have a huge effect on entertainment software.

"At the same time," Lindsey notes, "it's always exciting for developers to have new technology available, and we can see all sorts of opportunities ahead."

What types of opportunities? "A multitasking computer would let developers put together large realtime adventures where things are happening throughout a larger real world than what the player sees on the screen. Think of the player as in one room in a house filled with many rooms, with something going on in each of them."

Animation, too, is an area where 32-bit processors will have an effect, although Lindsey sees other technology coming home first. "Enhanced Graphics Adapters (EGAs) are already changing the way we think about PC graphics," he says, "changing the standards from CGA and monochrome."

"Improved processing speeds will allow for better animation." Lindsey points out that new computers offer a larger palette of colors as well. "Animation works best when there are more colors present to enhance to illusion," he says, "and the new generations of computers are offering huge color selections."

Another innovation whose arrival will be hastened by the increased availability of 32-bit machines is effective digitized sound. "Sound is the item that completes the illusion," says Bob Lindsey.

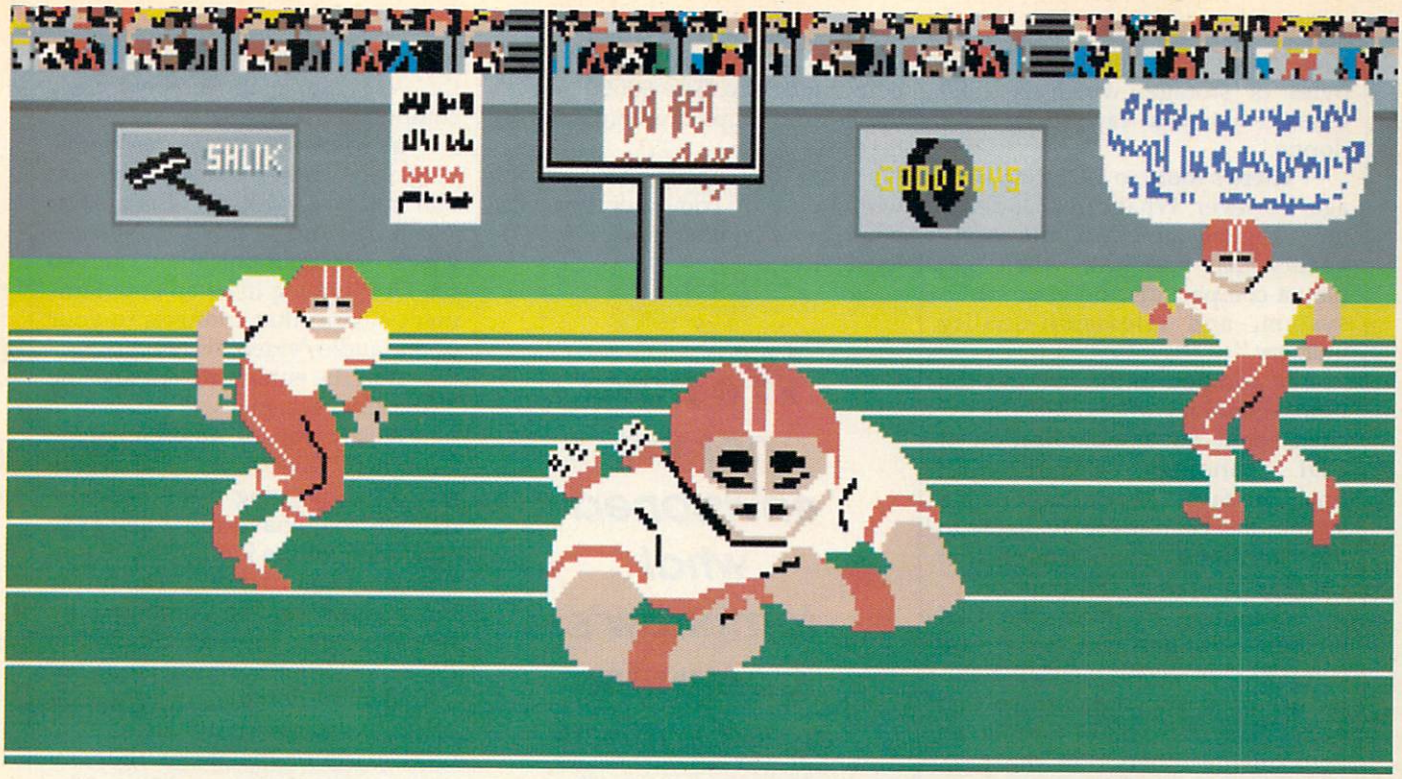
Spinoffs And Peripherals

One thing that's certain about the computer industry is that innovations don't occur one at a time. The 32-bit processors themselves imply that a variety of hardware and software support is required to take maximum advantage of the chip's capabilities.

What other innovations are coming, if not hand in hand with 32-bit processors, then at least hard on their heels?

"We're really excited about CDI," Riker says, referring to Compact Disc Interactive—systems that allow users to interface with the vast storage capabilities achieved by laser-read digital compact discs.

"My group is currently putting together development tools and hardware for the purposes of developing CDI systems. The systems use compact discs as storage media,



Amiga screen

SO REAL IT HURTS

GFL Championship Football™

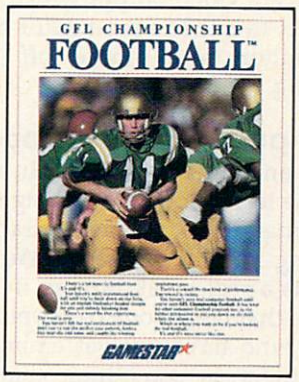
The way computer football *should* be.

Other football games put you in the grandstands, looking down on the action. Now see what it's like from the *player's* perspective—looking out of your helmet at an angry linebacker headed straight for you, and no blockers in sight.

With **GFL Championship Football,**™ you've got the first football simulation that actually takes you down on the field, taking the hits and making the plays. And it's more than just a pretty picture—you really get the feel of *playing* football.

No other football simulation gives you so many features:

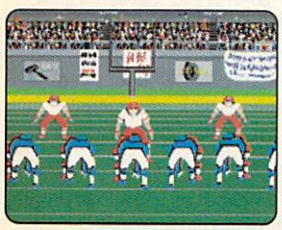
- *In-the-helmet perspective* puts you at ground level on the playing field.
- *Scrolling-screen animation* moves you up and down the playing field.
- *Realistic sound effects* let you hear everything from the quarterback calling the signals to the sound of your own footsteps.
- *Team selection screens* allow you to set the playing style of your team and that of your opponent.



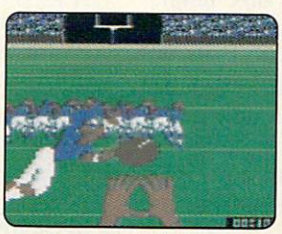
Whether you're taking on bone-crunching action against a friend, or going up against any of the 27 computer-controlled teams in the **GFL**, this is the one that puts you where the action is!

Available now for the Commodore 64/128, IBM PC and Tandy 1000, Apple IIe, IIc, Amiga, Atari ST and 100% compatible computers.

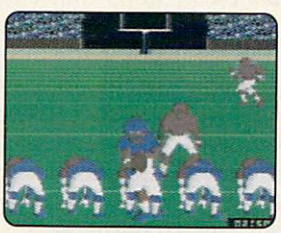
Look for Activision products at your local software dealer.
Or you can buy by mail at suggested list price by calling 1-800-227-6900.



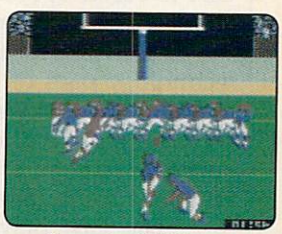
Amiga screen



Commodore 64-128 screen



Commodore 64-128 screen



Commodore 64-128 screen

GAMESTAR™

each having 550 or more megabytes of storage, which is 1500 times what you can stick on a floppy."

There's more to CDI than storage, however. What can CDI deliver? "Digital audio of the same quality that you hear when you play a compact disc on your stereo system, and television-quality imagery."

In addition to the features the new technology offers, Riker is excited about the standardization of CDI technology. "For the first time," he says, "a specification for a system has been made available to any hardware manufacturer that chooses to license and build it."

According to Riker, the CDI builders—Phillips and Sony—are committed to keeping CDI's architecture open and standard so that each CDI system will be compatible with all the others. "In the past, when the Commodore 64 came out, you could only get it from Commodore. When the Apple IIGS came out, you could only get it from Apple. With CDI, Phillips and Sony have made the architecture of the system available to any manufacturer who wants to build it."

This decision, Riker feels, will have a large impact upon consumers. "For the first time, consumers looking for a computer will have the assurance that there is a standard. Any CDI disc can be put into any CDI player. Before, you had to be a pretty aware shopper knowing that Commodore software ran only on Commodore hardware."

Electronic Arts is targeting several products to be released in late 1988—at the same time the hardware will be ready.

A New Generation—Again

Will the new CDI software require a new generation of chips? Not at all. CDI is based upon Motorola's 32-bit 68000 series of microprocessors.

But if this next generation of technology rests on existing chips, the computers themselves will be something quite different from the keyboard/computer/monitor combinations we're familiar with now.

"The CDI player is itself the computer," Riker explains, "but it's not being called a computer. It's a freestanding Compact Disc Interactive system, meaning that the user interacts with the system through a pointing device. The 68000 and all the custom chips are contained in the box."

That box will be linked not with a desktop computer, but with other, more familiar home electronics devices.

We have not yet envisioned what tomorrow's computer applications are going to be like.

"It has left and right audio outputs that connect to your stereo amplifier," Riker says. "There's a composite video output which you connect to either a modulator or to a monitor, and you're off and running."

Computers For Everybody

Riker and others involved in CDI see the technology as opening up a new market as well as attracting existing computer users. "We see CDI as a real breakthrough in getting to that large body of people out there who have not yet embraced computers as something that's a necessary part of their lives."

Because the discs are *read-only*, users won't be able to use them for word processing or other operations that change the disc itself.

"We see these players being put in the living room, and attached to the television set that the family watches," Riker says, "as opposed to being placed in the study and attached to a monitor as a peripheral to a word-processing type computer."

If this approach to CDI succeeds, it will represent a dramatic shift in the marketing of computers. "There's a real divergence that's beginning to occur now in terms of the way people think about their computers," Riker says.

In what way is the market diverging? "There's the kind of system that you work on," Riker says, "such as word processing, spreadsheets, telecommunications, and so on. Or it's going to be an entertainment system that connects to your home audio/video system."

And the entertainment-specific 32-bit CDI systems will bring us a new generation of entertainment? "You bet," Riker says. "The degree of realism is going to be unprecedented. You'll have audio that's the equivalent of the kind of audio you hear on a compact disc. The music for a game can be composed by John Williams, for example, or The Police, or Bruce Springsteen. All of a sudden the doors are thrown wide open for bringing in talent from other parts of the music industry that we never before could consider doing."

What about the pictures that go with the sound? "The incredible audio capability," Riker says, "will be coupled with the ability to present television-quality video on the screen. *Plus* it's interactive, not to mention the incredible amounts of storage space you have for audio, video, and other aspects of entertainment."

These two paths for computers come from the same source: The microprocessor whose evolution from 4 to 8 to 16 and 32 bits has been swift and is, doubtless, not over yet. Few who were involved in the industry a decade ago would have, or could have, predicted the shape of the industry and the part computers play in our lives today.

It's the same now, says Greg Riker. "These are exciting times, and I think we have not yet envisioned what tomorrow's computer applications are going to be like."

Compaq's Bob Beach is equally enthusiastic: "For the first time in the history of man's use of tools, we have a tool that is not created for any one specific task. By combining the right hardware and software, you can choose what task you want the computer to perform."

But while we wait for those applications to take shape, it's exciting—and entertaining—to speculate. ©

SYLVIA PORTER'S PERSONAL FINANCE SERIES MANAGES IT ALL

FREE CUSTOMER TECHNICAL SUPPORT

Your day-to-day finances. Your financial future. And now, your investments, too.

Volume 1

Your Personal Financial Planner™

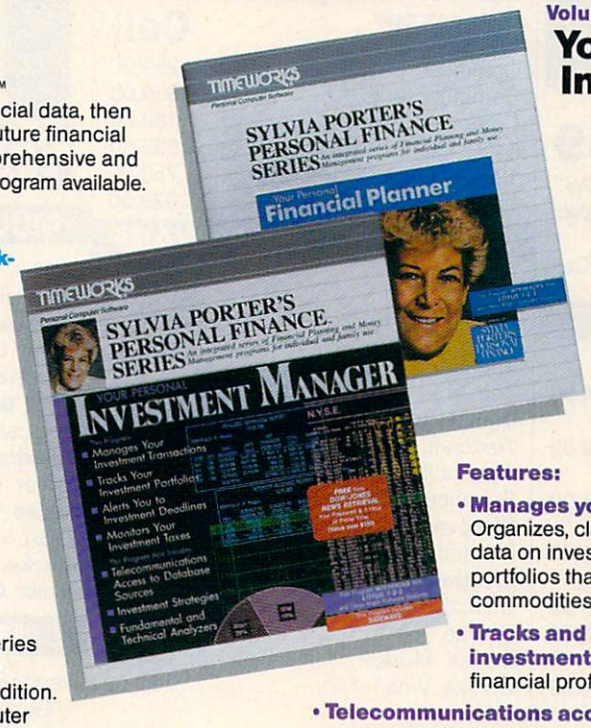
Helps you track your day-to-day financial data, then combines this information with your future financial objectives to produce the most comprehensive and easily-understood financial planning program available.

For Your Day-to-Day Affairs:

- **Maintains your electronic checkbook** and credit card transactions.
- **Writes your checks** and balances your checkbook.
- **Prepares and monitors your budget.**
- **Classifies and tracks your taxable income and expenses.**
- **Calculates your net worth and generates customized personal financial statements.**
- **Tracks your financial assets**—and your insurance policies.

For Your Financial Future:

Leads you step-by-step through a series of questions regarding your financial goals, and your current financial condition. Your answers will enable your computer to determine and print a summary of the amounts you must save each year to meet your financial objectives—in both real and inflated dollars.



Each SYLVIA PORTER program:

- **Interfaces with the others in the Series:** *You need to enter data only once.*
- **Generates unique graphic representations** that display your data in colorful charts.
- **Includes a Customized Report Writer** that prints out any report you want, tailored to your specific needs.
- **Includes Timeworks New Quick-Start Mini Manual:** Gets you up and running in less than 50 minutes!!

Volume 2

Your Personal Investment Manager™

This program combines portfolio management with powerful graphics and communications capabilities, enabling you to manage your *entire* investment program. YOUR PERSONAL INVESTMENT MANAGER is for both beginning and advanced investors—and for both novice and experienced computer users.

Advanced window capability allows you to view information on any transaction—as the need occurs—without leaving the section you're currently working in.

Features:

- **Manages your investment transactions:** Organizes, classifies, and maintains all important data on investment transactions. Manages multiple portfolios that include stocks, bonds, mutual funds, commodities, and much more.
- **Tracks and automatically updates your investments in each portfolio:** Plus, maintains financial profiles on individual companies.
- **Telecommunications access to on-line services:** Lets you automatically update your investments using current and historical prices from Dow Jones, CompuServe, and Warners.**
- **Unique Graphic Analysis and Charting Capabilities:** Creates bar, pie, high/low/close price, volume, relative strength, and comparison charts for up to 520 periods.
- **Imports and exports LOTUS, DIF, and ASCII Files**
- **Alerts you to investment deadlines**
- **Monitors your investment taxes**
- **Investment Strategies and Fundamental and Technical Analyzers**
- **Prints your spreadsheets sideways**
- **SwiftKeys*:** Just press a key and obtain a summary of your current holdings, "Hot Lists," an on-screen calculator, and more.

...PLUS *FREE*
from DOW JONES News/Retrieval*
Your Password and 1 Hour of Prime Time
(Value over $120!)

Suggested Retail List Prices:

IBM PC* & Compatibles (256K)
Vol 1... $99.95
Vol 2...$149.95

Apple* (256K)
Vol 1... $99.95

Commodore 128* (128K)
Vol 1... $69.95

Commodore 64* (64K)
Vol 1... $49.95

At your favorite dealer today, or call Timeworks: **1-312-948-9202**

TIMEWORKS®

MORE POWER FOR YOUR DOLLAR

Timeworks, Inc., 444 Lake Cook Rd., Deerfield, IL 60015
312-948-9200

**Phone modem required

*Registered trademarks of International Business Machines Corp., Apple Computer Systems, Inc., Commodore Electronics, Ltd., Dow Jones Co., Inc., and Timeworks, Inc.

When you want to talk computers..

ATARI COMPUTERS

Atari Computers

800 XL 64K Computer.....Call
65XE 64K Computer.....87.99
130XE 132K Computer.....129.00
520ST Monochrome System.....499.00
520ST Color System.....659.00

Atari 1040 Color System $859

Includes: 1040ST, 1 mb RAM with 3½" drive built-in, 192K ROM with TOS, Basic, ST language and color monitor.
New 520ST FM is in stock.....Call

ATARI SOFTWARE

Access
Leaderboard Golf.....24.99
Accolade
Fight Night.....19.99
Activision
Music Studio.....34.99
Antic
Cad 3-D.....32.99
Batteries Included
Paperclip w/Spellpack.....39.99
Degas Elite.....48.99
Infocom
Zork Trilogy.....44.99
Microprose
Top Gunner.....19.99
Silent Service.....22.99
Optimized Systems
Personal Pascal.....47.99
Origin Systems
Ultima 4 XL/XE.....39.99
Paradox
Wanderer (3-D).....27.99
Psygnosis
Deep Space.....34.99
Timeworks
Wordwriter ST.....48.99
VIP
Professional (GEM).....144.00

COMMODORE COMPUTERS

Amiga 500 System

Includes: Amiga 500 CPU, 1 MB, 1080 RGB Monitor, Amiga DOS, Mouse, Kaleidoscope **Call**

Commodore 128.....259.00
Commodore 128D.....539.00
Commodore 64C.....189.00
64C, 1541C, 1802C Package....599.00
128, 1571, 2002 Package.....759.00
128D, 2002 Package.....829.00

COMMODORE SOFTWARE

Activision
Gamestar Series.....(ea.) 28.99
Broderbund
The Print Shop.....29.99
The Toy Shop.....39.99
Commodore
Textcraft w/Graphic Craft.....59.99
Assembler.....79.99
Enhancer DOS 1.2.....14.99
Discovery Software
Marauder Back-up.....32.99
Electronic Arts
Deluxe Paint II.....97.99
Deluxe Print.....74.99
Instant Music.....34.99
Deluxe Video 1.2.....97.99
Infocom
Hitchhiker's Guide.....28.99
Micro Illusions
Dynamic-Cad.....349.00
Mindscape
Halley Project.....34.99
SAT Prep.....51.99
Micro Systems
Analyze Version 2.0.....119.00
Scribble.....66.99
On-Line/Comm.....46.99
Sublogic
Flight Simulator.....31.99
V.I.P.
V.I.P. Professional.....112.00

MS/DOS SYSTEMS

AT&T 6300.....from $1299.00
Compaq.....from 1699.00
IBM-PS-2 Model 30.....Call
IBM-AT.....from 2499.00
Leading Edge.....from 999.00
NEC Multispeed.....from 1499.00
Panasonic Business Partnerfrom 799.00
Toshiba 1000 Lap Top.....from 999.00

PC-TOO 20 Meg XT-Compatible $999

MULTIFUNCTION CARDS

AST
Six Pak Plus PC/XT.....129.00
Hercules
Color Card.....159.00
Graphics Card Plus.....209.00
Fifth Generation
Logical Connection 256K.....329.00
Quadram
Silver Quadboard.....119.00
Video 7
EGA Video Deluxe.....339.00
Zuckerboard
Color Card w/Parallel.....89.99

MS/DOS SOFTWARE

Ashton-Tate
d-Base III+.....399.00
5th Generation
Fastback Utility.....89.99
IMSI
Optimouse w/Dr. Halo.....99.99
Lotus
Lotus 1-2-3.....329.00
MicroPro
Professional 4.0 w/GL Demo.....239.00
Microstuf
Crosstalk XVI.....89.99
P.F.S.
First Choice (Premium).....119.00
Word Perfect Corp.
Word Perfect 4.2.....209.00

COMPUTER MAIL ORDER

.....When you want to talk price.

DRIVES

Atari
AA314 DS/DD Disk (ST).....$209.00
AA354 SS/DD Disk (ST).....139.00
SHD204 20 Meg ST Hard Drive579.00
Commodore
Amiga 1010 3½".....219.00
1541C.....179.00
1571.....239.00
1581 3½" External.....229.00
CSI
10 mb (64-128).....1099.00
Indus
GT Disk Drive Atari XL/XE.....179.00
Microbotics
20 mb Hard Drive (Amiga).....1299.00
Racore
Jr. Expansion Chassis.....299.00
Seagate
20 mb ST-225 Hard Drive Kit....369.00
Supra
20 Meg Hard Drive (ST).....559.00
Xebec
20 mb (Amiga).....899.00

MONITORS

Amdek
Video 300 Amber Comp.....139.00
Commodore
Commodore 2002.....319.00
Amiga 1080 Hi-Res Color.....299.00

Magnavox 8502 RGB/Composite $179

Magnavox
8505 RGB/Composite.....199.00
NEC
12" TTL Green or Amber.....109.00
JC-1401P3A Multi-Sync.....549.00
Princeton Graphics
MAX-12 12" Amber TTL.....149.00
Taxan
Model 124 12" Amber.....119.00
Zenith
ZVM 1220/1230.....(ea.) 99.99

MODEMS

U.S. Robotics 1200 Baud Direct $99.99

Anchor
Volksmodem 1200.....99.99
6480 C64/128 1200 Baud.....$119.00
VM520 ST520/1040 1200 Baud.129.00
Atari
XM301 300 Baud.....42.99
Best
1200 Baud External.....119.00
Commodore
Amiga 1680-1200 BPS.....169.00
CBM 1670 & C-128).....99.99
Everex
Evercom 1200 Baud Internal.....109.00
Hayes
Smartmodem 300 External.....139.00
Smartmodem 1200 External.....319.00
Novation
Parrot 1200.....119.00
Practical Peripherals
1200 BPS External.....159.00
Supra
MPP-1064 AD/AA C64.....69.99
1200AT 1200 Baud Atari.....139.00
U.S. Robotics
2400 Baud Internal.....$189.00

DISKETTES

Maxell
MD1-M SS/DD 5¼".....9.99
MD2-DM DS/DD 5¼".....10.99
MF1-DDM SS/DD 3½".....12.99
MF2-DDM DS/DD 3½".....21.99
Sony
MD1D SS/DD 5¼".....8.99
MD2D DS/DD 5¼".....10.99
MFD-100 SS/DD 3½".....13.99
MFD-200 DS/DD 3½".....20.99
Hewlett-Packard Calculators
28C Scientific Pro.....199.99
18C Business Consultant.....139.95
12C Slim Financial.....74.99

PRINTERS

Atari
1020 XL/XE Plotter.....$30.99
1025 XL/XE Dot Matrix.....129.00
XDM121 Letter Quality.....199.00
XMM801 XL/XE Dot Matrix.....189.00
XMM804ST Dot Matrix.....179.00
Brother
M-1109 100 cps, 9 pin.....199.00
M-1409 180 cps, 9 pin.....339.00
Citizen
MSP-10 160 cps, 80-Column.....299.00
Premier 35 cps Daisywheel.....499.00
C.Itoh
315P 132 Column Prowriter.....449.00

Epson LX-800 150 cps Dot Matrix $169

Epson
FX-86E 240 cps, 80-column.....329.00
FX286E 240 cps, 132-column....459.00
EX-800 300 cps, 80-column.....399.00
LQ-800 180 cps, 24-Wire.....459.00
Hewlett Packard
Thinkjet.....379.00
NEC
Pinwriter 660 24 Wire.....489.00
Pinwriter 760 24 Wire.....689.00
Okidata
Okimate 20 Color Printer.....129.00
ML-182 120 cps, 80-column.....239.00
ML-192+ 200 cps, 80-column...369.00
ML-193+ 200 cps, 132-column.549.00
Panasonic
KX-1080i 120 cps, 80-column....179.00
KX-1091i 180 cps, 80-column....189.00
KX-P3131 22 cps Daisywheel....289.00
Star Micronics
NX-10 120 cps, 80-column.....179.00
NX-10C 120 cps, C64 Interface.219.00
NX-15 120 cps, 132-column.....359.00
Toshiba
P321SL 216 cps, 24-Pin Head...539.00

In the U.S.A. and in Canada

Call toll-free: 1-800-233-8950.

Outside the U.S.A. call 717-327-9575 Telex 5106017898 Fax 717-327-1217

Educational, Governmental and Corporate Organizations call toll-free 1-800-221-4283

CMO. 477 East Third Street, Dept. A210, Williamsport, PA 17701

ALL MAJOR CREDIT CARDS ACCEPTED.

POLICY: Add 3% (minimum $7.00) shipping and handling. Larger shipments may require additional charges. Personal and company checks require 3 weeks to clear. For faster delivery use your credit card or send cashier's check or bank money order. Pennsylvania residents add 6% sales tax. All prices are U.S.A. prices and are subject to change and all items are subject to availability. Defective software will be replaced with the same item only. Hardware will be replaced or repaired at our discretion within the terms and limits of the manufacturer's warranty. We cannot guarantee compatibility. All sales are final and returned shipments are subject to a restocking fee.

World Class Leader Board

James V. Trunzo

Requirements: Commodore 64, Apple II series (with a minimum of 64K), and IBM PC and compatibles (with a minimum of 256K).

If, by the time this is published, playing 18 holes is just a memory of last summer, you'll be thrilled to know that you can vicariously exercise your passion for golf through the use of a new software simulation. On the other hand, if you are reading this in a place where the sun is scorching, you can delight in the knowledge that you can beat the heat, stay indoors, and still "hit the links" each day. Either of the above scenarios can be accomplished simply by loading *World Class Leader Board* into your computer.

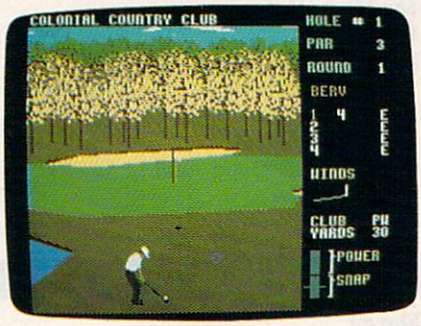
World Class Leader Board from Access Software isn't the first arcade-style golf simulator, but it's got to be one of the best. A simulation of this type needs to possess three key ingredients if it expects to gain recognition in the competitive field of sports simulations: It must be challenging, yet fair; it must faithfully simulate the sport; and it must be aesthetically pleasing. *World Class Leader Board* exceeds the norm in all three categories.

Like The Real Thing—Tough

To begin with, the game starts as a challenge and remains one in such a manner as to parallel the real sport of golf. Being an arcade game, *World Class Leader Board* requires good hand/eye coordination and concentration (the two really go together). So does the real game. And like the real game, this simulation rewards and punishes each drive, each approach shot, and each putt in such a gradient manner that if your timing on any given day is off just a little, the result is, at best, a par, but more likely a bogey. Therefore, even after playing hundreds of holes, you no more master the game than does a Greg Norman or an Arnold Palmer. If your hand/eye coordination is off more than a little, well, head for the nineteenth hole.

On the other hand, when it all comes together—when your backswing generates maximum power and your wristsnap guarantees a straight flight—*World Class Leader Board* provides you with such a sense of accomplishment that you want to throw your imaginary golf club high in the air and shake hands with your caddie. The game provides this thrill, in a large part, through its excellent graphics and animation.

The graphics in *World Class Leader Board* are a delight. Two views are available as you play each hole. A Top View provides you with an overview of the entire hole, citing your current position. The most commonly used view, however, is a full-screen view seen from behind a highly detailed golfer figure.

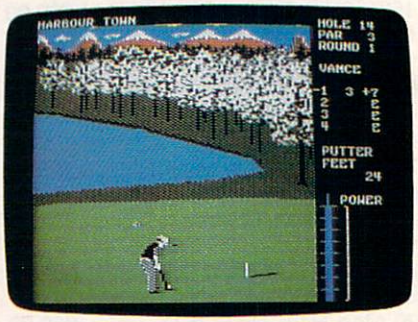


Commodore 64 version of World Class Leader Board.

The latter view clearly delineates fairways, rough, greens, and obstacles such as trees, water, and sandtraps in a realistic and functional manner. Furthermore, considering how much detail has gone into displaying each hole (up to 196 trees per hole, for example), the screens are quickly drawn and redrawn when shifting from view to view or moving from stroke to stroke. This is a bonus, because some simulations tend to get bogged down because of the time delay between actions.

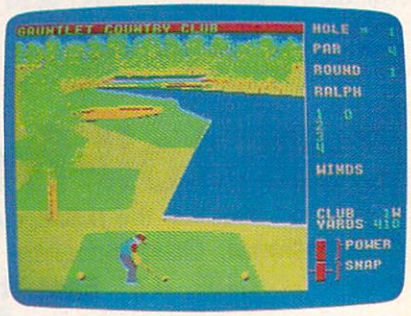
Fore!

As mentioned, the graphics are not only aesthetically pleasing, but also functional, a factor which cannot be underestimated—in a game of this type, part of the vicarious experience is "seeing" what you have done as opposed to just knowing you drove the ball 250 yards or missed a putt by inches. In *World Class Leader Board*, every hook, slice, chip shot, and putt are accurately displayed. You literally see a putt slowly curving towards the hole—or popping over it if you've putted too hard. You see the ball bounce off the flag when you've hit a perfect 9-iron shot—or bounce off of a tree trunk when your shot leaves the fairway. You hear the plop when the putt sinks into the hole—or into the water. No detail has been ignored by Access in designing this game.



World Class Leader Board—*Apple II version.*

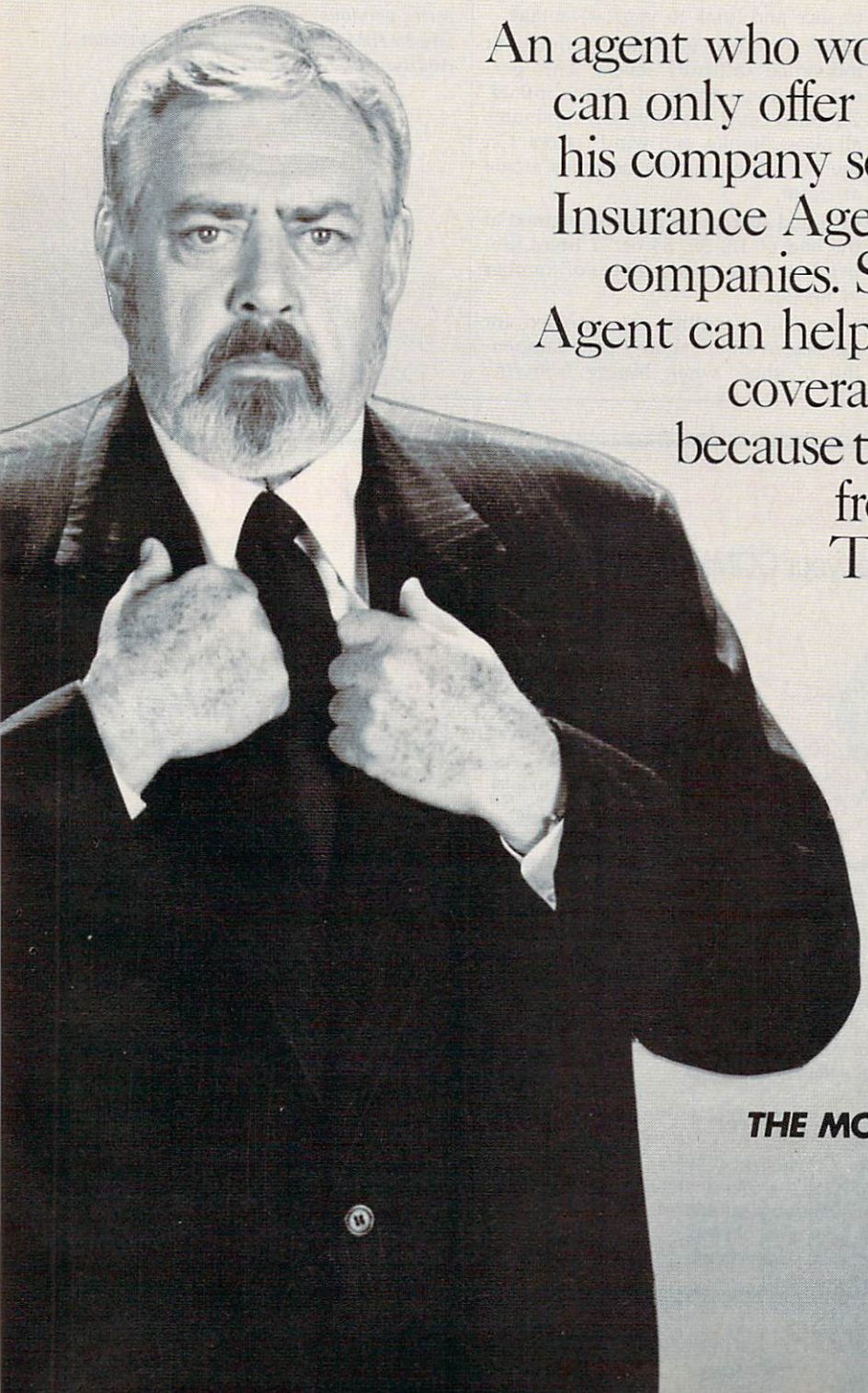
The golfer figure itself is also worth mentioning. Full-figured, the golfer is an integral part of the game. You measure his backswing and follow through in two ways. The first, obviously, is by watching the animated figure; the second way is by watching sidebars that gauge power, snap, and distance. These sidebars work in sync with the golfer to produce all shots taken during the



World Class Leader Board—*IBM PC and compatibles version.*

"Before you buy insurance, examine the evidence."

RAYMOND BURR



An agent who works for one company can only offer you the policies that his company sells. An Independent Insurance Agent represents several companies. So your Independent Agent can help you select the right coverage at the right price because there are more policies from which to choose. The evidence is clear.

Independent Insurance Agent.

THE MORE-THAN-ONE-COMPANY INSURANCE AGENT.

You'll find the Independent Insurance Agent nearest you in the Yellow Pages.

course of play. Incidentally, all choices and actions can be controlled through a joystick or the keyboard.

Simulated Clubs

World Class Leader Board provides the player with a complete choice of clubs: 3 woods, 9 irons, a pitching wedge and a putter. Each club produces a range of results realistic to the type of club used and dependent upon how cleanly the ball is hit. You can also choose various power levels to alter the distance produced by each club within its given parameters.

Actual play mirrors reality almost too closely when you have a stroke like mine. Four courses are available for play: St. Andrews, the Doral Country Club, Cypress Creek, and a course known forebodingly as the Gauntlet. As expected, each course is an exact replica of the original, with obstacles and hazards placed in frustratingly challenging (but realistic) positions. Putting greens are sloped, and players must correctly read the break as well as recognize whether or not he is putting uphill or down. As if these factors weren't enough, Access has added a wind factor that can range from a refreshing breeze to a gale that affects all shots.

Extra Features

It's obvious that *World Class Leader Board* can claim excellence in all the crucial areas needed to make it a top-flight game. Nevertheless, Access goes that extra step to make the game even better. Consider the following features:

- A course editor which allows you to create your own course by choosing from among the 72 holes present in the game.
- Practice driving ranges and putting greens to sharpen your game before competition. This not only gives you the opportunity to learn how to drive the ball straight, it also allows you to learn to slice and hook to your advantage.
- One to four players may compete at individual difficulty levels. This permits various types of handicapping. The Kids Level, for example, eliminates all slices, hooks, and wind factors. On the Pro Level, putting becomes a true art form.
- Printed scorecards can be obtained at the end of 18, 35, 54 or 72 holes (depending upon user's choice of number of holes).
- Realistic penalties are assessed for landing in water, there's true dampening of shots when hitting from the rough, and you encounter life-like difficulty when blasting out of a sandtrap.

Worth noting, too, is the fact that Access is intent on supporting *World Class Leader Board*. Additional 18-hole course layouts for use with this game are available.

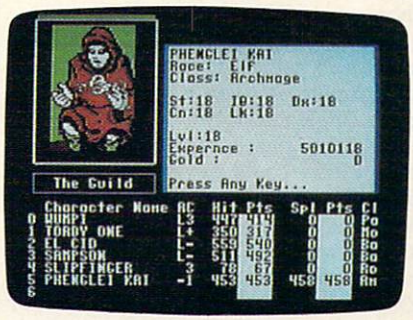
World Class Leader Board is an excellent buy for both the casual sports enthusiast and the avid golfer. It scores an eagle in all departments.

World Class Leader Board
Access Software
2561 S. 1560 West
Woods Cross, Utah 84087
$39.95 Commodore 64 and Apple II series versions
$44.95 IBM PC and compatibles version
Additional course disks: $19.95

The Bard's Tale II: The Destiny Knight

James V. Trunzo

Requirements: Apple II series and Commodore 64.



Electronic Arts' first venture into computerized fantasy role-playing games resulted in the highly popular and critically acclaimed *The Bard's Tale I: Tales of the Unknown*. The product immediately set new standards for game play, graphics, and sophistication. The long awaited sequel—entitled *The Bard's Tale II: The Destiny Knight*—has arrived, and it lives up to its predecessor's excellent reputation.

The plot is typical of a fantasy game: Evil in the form of the Archmage Lagoth Zanta threatens to spread its dominion over the entire Realm, and only the Destiny Wand has the power to end the threat. Unfortunately, this magical artifice has been shattered into seven pieces, each piece now hidden in one of seven different locations within the Realm. Your group of adventurers must locate the pieces, reconstruct the wand, and destroy the wicked Zanta.

New Features

Bard's Tale II plays almost identically to *Bard's Tale I*—which isn't a fault. Changes come in the form of additions rather than in alterations of basic gaming procedures. There are six complete cities instead of one, though the cities themselves are basically the same, with only their layout being different. There is a small but interesting area of wilderness in *Bard's Tale II*, where none had existed in *Bard's Tale I*, and there are new spells and new animated monsters. All in all, *The Destiny Knight* is about 50 percent larger in code size, according to the people at Electronic Arts.

Banks and casinos can now be found in the cities, also. The former is a real plus because it provides you with a place to stash your money, making it less likely to lose it to some ultimate evil—like a power outage or stupidity (such as turning off your computer before saving your characters).

Creating characters requires the same steps in *Bard's Tale II* as it did in *Bard's Tale I*. You can also import your characters from *Bard's Tale I* (or *Wizardry* or *Ultima*, for that matter) if you so desire, and a third option allows you to simply use a set of pregenerated characters. Be forewarned, however, that when transferring characters, certain items might be lost in the process, though not enough to make you forsake your favorite Hobbit.

The character classes and types are standard: warriors, thieves, wizards, humans, elves, dwarves, and the like. However, the Bard character—unique to EA's adventure games—is back, and a new character class debuts: the Archmage. This character class could be achieved in *Bard's Tale I* in the form of a conglomerate of the other magical types; in *Bard's Tale II*, the Archmage has its own set of very powerful spells.

Several other new features help make *The Destiny Knight* different from its predecessor.

First of all, range factors in combat add a new twist to the hacking and slaying that occurs with such frequency in this program. Missile weapons may now be employed to attack enemies outside melee distance. Even magic spells are given a range of effectiveness, losing power if cast at distances beyond their given scope. The range feature requires the gamer to employ more strategy than before, making combat a little more interesting.

Combat, however, reveals a problem in the program. Early on, your fighters are key characters, bearing the brunt of the fighting while magic users develop their skills and learn new spells. Later on, though, the opposition becomes so powerful that even high-level fighters can't seem to score against them with any frequency, and only high-level spellcasters can save your group of adventurers at this point. There is something vaguely unsatisfying about this, especially if your favorite character, carefully nurtured, is a warrior.

Perplexing Puzzles

The realtime puzzles, better known as Snares, are a second major feature not found in *Bard's Tale I*. These puzzles, seven in all, must be solved because each one houses a piece of the Destiny Wand. The Snares are deadly. Virtually no magic works in these areas, and the game's pause feature is inoperative there. If the puzzle isn't solved in the allotted time (usually everything you need is contained within the Snare it-

self), your party perishes. This concept isn't inherently bad—as a matter of fact, several of the Snares are really enjoyable, but there is a down side to this feature, too.

I found some of the Snares to be not only deadly but also very frustrating. The fact that some of the puzzles are difficult doesn't bother me. What does bother me is the need to repeat certain actions over and over again when those actions add nothing to the information needed to solve the puzzle. For example, in one case, you literally must run back and forth between two walls a number of times to solve the puzzle. That's the type of "action" most gamers can do without.

A third area that can both delight and annoy is the interaction between the wise Sage, who is an indispensible personage, and your characters. The Sage possesses knowledge about the location of the dungeons. By conversing with him, you can gain the information you need. Be forewarned, though, that you must be very precise in what you ask and in how you spell the words that comprise your questions and responses. Errors in spelling or omitted words in a request can thwart your attempt to get answers from the Sage. This type of problem exists in many adventure games, especially text adventures; however, since it costs gold to talk to the Sage, errors are quite expensive in *The Destiny Knight*.

Expanded Challenges

Before this review takes on too negative a tone, understand that I found the game to be a delight. It improves on *Bard's Tale I* in many areas, and it does what a sequel is meant to do: It expands the theme and adds features to enhance that expansion. The new monsters, graphics, traps and puzzles, and so on, make *Bard's Tale II* a real challenge. Furthermore, the graphics are, once again, done well, and there is more animation in *Destiny Knight* than in *Bard's Tale I*.

Also, powerful monsters can now be summoned to be a permanent part of your adventuring group. In *Bard's Tale I*, creatures can be summoned, but they vanish after the adventuring party returns to the Guild. Additionally, because a guild exists in each city in the new adventure, games can be saved at any one of six different locations. Finally, a beginner's dungeon exists to help new parties reach a competitive level prior to venturing forth in search of the pieces of the Destiny Wand.

This last feature, the starter dungeon, is a nice piece of planning. Many adventure games are difficult to get absorbed in because new characters get killed so quickly. While *Bard's Tale II* is no less deadly (more on this in a moment), it at least gives you a chance—not to mention some game play—to get characters off the ground.

Bard's Tale II is a game for the true adventure gamer. It is a very difficult and challenging game, and it requires great intestinal fortitude. Not that I should be the barometer, but quite truthfully, I could never have made it through the game without continuous help from the clue book (available separately from Electronic Arts). While I do not hesitate to recommend *The Destiny Knight* to experienced gamers, I'm not sure that it is the product on which inexperienced players should cut their fantasy teeth.

New Life For An Old Favorite

I would be remiss, if, at this point, I failed to mention that *Bard's Tale I: Tales of the Unknown* is now available for the Apple IIGS for $49.95.

The game itself is nearly identical to the Apple IIe version; however, the GS adaptation makes use of the machine's Macintosh-like interface, employing pull-down menus and the mouse to activate commands, and so on. And, from the moment the program is booted, it becomes obvious that the overwhelming features of *Bard's Tale I* for the Apple IIGS are, not surprisingly, its graphics and sound. It is unquestionably the most graphically stunning product that I have seen on any Apple computer.

From the title page, when a full-screen Bard strums his mandolin, to the Latin chants of the temple monks as they heal an adventurer, *Bard's Tale I* for the GS is one fascinating delight after another. Each instrument played by the Bard has its own sound; each monster and character possesses a different form of animation.

The game is so impressive that I have begun to replay *Bard's Tale I* on the GS, even though I have completed it on the IIe. I am willing to invest the time so I can discover the rest of the three-dimensional, animated delights in the GS version.

A program with no problems? Not quite. The graphics presentation of *Bard's Tale I* for the Apple IIGS is so good that I have become jaded. Other games have lost some of their appeal now that I have experienced the capabilities of a GS program—and that's a problem.

Bard's Tale II: The Destiny Knight
Electronic Arts
1820 Gateway Dr.
San Mateo, CA 94404
$49.95 Apple II-series version
$39.95 Commodore 64 version
Clue book: $12.95

President Elect: 1988 Edition

Ervin Bobo

Requirements: Apple II series (with a minimum of 48K), Commodore 64, and IBM PC and compatibles (with a minimum of 256K).

In order to fully enjoy *President Elect*, you must first decide what you want it to do for you. You could use it to find the strongest candidate for 1988 and place your election bets accordingly—though we'll accept no responsibility for an upset and your loss of the rent money.

Students or hobbyists in political science might wish to use the program to recreate campaigns of the past 28 years; others may wish to see the results of campaigns that never were; and still others will want to test their own campaigning skills by running against the computer or a human opponent.

President Elect will allow you to do all these things—and to do them with what we suspect is more than reasonable accuracy.

Past, Present, Or Future

To begin a contest, you first select a presidential election year between 1960 and 1988. Should you choose the future, you'll be asked a series of questions concerning unemployment, inflation, national mood, and so on—the stuff of which all campaigns are made. For past years, these parameters have already been factored in, but you'll have to make other choices, including Historical or Ahistorical, the former allowing you to place your own candidates within the framework of real history. You also must choose whether one of the candidates is an incumbent.

In an ahistorical simulation, you are not bound by the constraints of history: You may have Hubert Humphrey run as a Republican, Barry Goldwater run as a liberal, Lyndon Johnson run against Teddy Kennedy, or Gerald Ford run as a third-party candidate. These and many other permutations are possible. Additionally, you can play by "name-only," where you and a friend do the campaigning using the candidates' names; or you can run campaigns as fully computerized simulations in which your computer knows the strengths and weaknesses of each candidate and determines the outcome of the match.

The campaign itself is composed of nine turns, each corresponding to one of the nine weeks between Labor Day and Election Day. During each turn,

candidates do the usual things candidates do—such as calling for debates, spurning invitations to debates, deciding whether or not to visit a foreign country, making questionable statements, and spending campaign money (in this case, spending PAPs, or Political Action Points).

Each week begins with the results of the latest poll, showing what percent of the voters favor each candidate and then using this figure to determine the number of electoral votes each would secure if the election were held immediately. Following this update, there is a Current Events phase, and, if you are actively playing (as opposed to watching a computer simulation), you'll have to decide what you might say or do about the poll's results. Next comes the week's campaigning, during which you'll spend your PAPs where you think they'll do the most good—on a national or regional basis. You'll also decide which states might be swayed by a personal visit.

In playing against a human opponent, this part of the campaign is kept secret, each player using the computer long enough to enter his or her decisions while the other player leaves the room for coffee and doughnuts.

This sequence of plays occurs in each of the nine turns. At the end of each turn, a map displays the states favoring each candidate and the states still undecided (the only graphics display in the game), giving you a chance to formulate strategy for the next turn.

Should a debate occur, you'll be given a question which, when answered, will be rebutted by your opponent. You do not have to be a scholar to answer; you must decide only how much weight to give the five possible responses: Discuss Relevant Considerations, State Position, Attack Opponent, Kill Time, and Criticize with Witticisms. Following the debate, the media declares one of you the winner.

Then, once the campaign is over and the polls close, the preliminary results and updated projections arrive, much as they do on television. You can let this take its normal "realtime" course, which takes a few hours; or you can speed the results by a factor of ten—or you may jump to an immediate conclusion.

In setting up a simulation, demographics play an important role, and while we do not know the extent to which they are used (demographics are as endless as pi), we are told in the documentation that there are different parameters used for each election year. For example, Florida, which today has 21 electoral votes, possessed only 10 in 1960. Individual states are also biased according to history, and such history becomes probability when dealing with the election of 1988.

Perhaps the most important factor is the political profile of a candidate—his stance on various issues, his poise and magnetism, his ability to speak. These things have been factored into *President Elect* and seem to have a different meaning in different election years.

Reagan Versus Kennedy

One of our tests involved a fully computerized 1988 matchup between Ronald Reagan and John Kennedy. Perhaps surprisingly for those of us who remember the aura of the Kennedy era, Reagan not only won the single debate but also the election—perhaps proving that communication is tougher than charisma.

However, a similar matchup against the temper of the times of 1960 had Kennedy winning by a landslide that was so great as to be embarrassing. What these simulations prove is that the database from which *President Elect* works is both extensive and accurate.

However the simulation is used, in historical or ahistorical mode, it is fun. And it is helped along by good documentation, including a pull-out score card (of which you'll want to make copies for successive campaigns) and a listing of recent political front runners with numerical weighting of aspects of their personal political profiles.

But where your election bet is concerned, it should be noted that neither COMPUTE! nor SSI will advance you the money.

President Elect: 1988 Edition
Strategic Simulations
1046 N. Rengstorff Ave.
Mountain View, CA 94043
$24.95

Mentor

Keith Ferrell

Requirements: For the IBM PC and compatibles with a minimum of 256K and a CGA (Color Graphics Adaptor) card.

Have you ever wondered what your IQ is? Or have you finished an IQ test at school or work and wished you had a chance to take the test on your own time? We encounter many different types of psychological tests at various points in our education and careers, and many of us are frustrated at their impersonality. Now there's a piece of software that lets you test your IQ, and other psychological performances, on your PC.

Mentor, the first product released by Heuristic Research, is a disk packed with more than 50 separate tests aimed at examining and improving your memory, reaction time and coordination, perception and balance, aural pitch, as well as your general mathematical and verbal intelligence. Thrown in for good measure is a biorhythm chart.

The program contains instructions, information, and tutorials on disk. *Mentor*'s manual announces itself as the smallest set of instructions in the world, and it just may be. Simply insert the disk in drive A:, type *mentor*, and begin. *Mentor* will run on monochrome composite monitors, however, a color graphics adaptor (CGA) is required. Its setup features include user-specified resolution, text size, and colors.

Start Testing

The first thing you encounter after booting the program and supplying it with your name and birth date, is a menu offering your choice of psychological tests, IQ tests, biorhythm, and information about the program and screen setup options.

The psychological test section offers further choices. You may be tested for memory, reaction, coordination, or general perception. In turn, each of these options narrows the parameters of your test even more. For example, as well as offering three difficulty levels, the memory test section offers you the option of being tested for memory of color patterns, sound and pitch, numbers, or letters.

Mentor suggests that you run the program's introduction before undertaking any of the actual tests, and that's good advice. Part of doing well on IQ and similar tests is knowing how the tests work; *Mentor* gives you a glimpse of its operations before you start testing. Forewarned is forearmed.

It's also advisable to start each testing sequence at the easiest level, however confident you are of your abilities. When testing your memory of numbers, for example, the program flashes seven digits, then prompts you to enter them in the correct order. At the easiest level, you see the digits for one second; at the most advanced level, the numbers are onscreen for only a quarter of a second.

Once the digits disappear, it's up to you to enter them, in the correct sequence, as quickly as you can. *Mentor* lets you know whether you are right or wrong and how long your attempt took. It also keeps track (on the screen) of the number of correct attempts you've made, as well as showing how many of the digits you got right for each example. The disk includes similar tests for letters, color, and sound.

IQ Tests

Mentor's general IQ tests are as tough and thorough as any I care to attempt. A typical test gives you 50 questions to be answered in 40 minutes. The questions and exercises stretch your verbal, numerical, and visual skills and perception. In addition to general IQ tests, there are specific verbal and numerical ability tests.

Verbal questions take several forms. One question might ask you to enter the word that matches *both* of two dissimilar words outside the parentheses; an example is: ARROW (_____) FASTENER. That is one of the simpler questions (the answer is *bolt*). Other verbal ability tests request that you complete a sequence of letters or supply a word based on others in a group.

Numerical ability tests likewise take several forms. You are asked to provide the right number for a sequence or to fill in a blank in a relation of numbers.

Readers who have taken IQ tests will recall the sections where a series of similar geometric shapes are presented, with the final space blank, to be filled in with the proper selection. That type of test, too, is included in *Mentor*. As the requirement of a CGA card implies, the program's graphics are excellent, with detailed resolution that gives you a fair chance to study the relationships among the figures so you may select the correct answer.

When your 40 minutes are up, the program evaluates your performance, presents you with its measurement of your IQ, and offers you the chance to go through the information section once more. It's worthwhile to read through the information more than once—this section gives good, general advice on the nature of psychological tests. In fact, I'd like to see the information section expanded, as it would be nice to learn more about the nonstandard tests that *Mentor* includes.

Think Fast!

Mentor also provides reaction, perception, and coordination tests. You can check your eye-hand, ear-hand, and eye-ear-hand coordination. My favorite from this section of the program presents you with a pattern of dots scattered across the screen, challenging you to position the cursor at the center of the pattern. It's harder than it sounds, and much harder than it looks.

Other general psychological tests include perception of quantities. In one timed test, you are given a few seconds to ascertain the number of dots on the screen. As with all of the other tests in *Mentor*, these are challenging and informative.

Just for fun, the program also includes a biorhythm generation chart that produces a graphics representation of your emotional, intellectual, and physical cycles over the month of your choice. There is also an information section that discusses, without endorsing, biorhythm.

Mentor is an impressive package. Its on-disk documentation suggests that it can be used for self-improvement, for monitoring the intellectual development of children, and for preparation for actual psychological tests. All true, but the program can also serve as an effective mental "exerciser," just the sort of thing to sharpen thinking skills and hone analytical instincts and abilities. The program is packed with so many tests and configurations of tests that it should be hard to exhaust its challenges.

If you are curious about your intellectual and psychological abilities, *Mentor* is an affordable, provocative, and entertaining way to learn about yourself while improving your ability to succeed at such tests.

Mentor: *Psychometric Software*
Heuristic Research
3112-A West Cuthbert Ave.
Midland, TX 79701
$49.95

Marbles

Stephen Stout

Act quickly—and don't panic—in this clever arcade-style game for the Atari, Commodore 64, Amiga, Apple II, and IBM PC/PCjr and compatibles. The Atari, 64, and Amiga versions each require a joystick. The Apple version requires paddles. The IBM PC/PCjr version requires BASICA or GW-BASIC and a color/graphics adapter for the PC and compatibles, and Cartridge BASIC for the PCjr. The Apple version runs under DOS 3.3 or ProDOS.

In two weeks the Statewide Marbles Championship will be held in Localsville. You've been practicing for nearly a year, and now, at the last minute, you find you've lost your favorite marble. You're going to have to collect every marble you can to find one as special as the one you lost.

"Marbles" is a hypnotic arcade-style game in which you try to catch marbles in a bucket. Originally written for the Atari eight-bit computer series, we have added versions for the Commodore 64, PC/PCjr and compatibles, Apple, and Amiga computers. Each version of the game is unique, so be sure to read the instructions for your computer for the details of the game.

Atari 400, 800, XL, And XE Version

The Atari version of Marbles (Program 1) is written entirely in BASIC. Type it in and save it to tape or disk. The program listing contains many special graphics characters. Refer to the "Guide To Typing In Programs" article elsewhere in this issue for details on typing these characters. To play the game, simply load and run the program.

Use a joystick plugged into port 1 to control the buckets. Press fire to start the game. The marbles fall, one at a time, down the pipes. When a marble comes to an intersection, it may go in any direction.

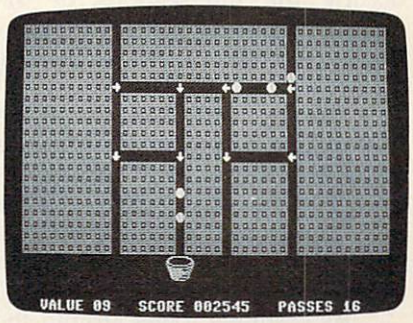
There are seven screens in this version of the game. Your task is to catch 20 marbles on each screen without dropping even one. If you do drop a marble, you must start over with the same screen. To further complicate matters, you must catch red marbles only in the red bucket and blue marbles only in the blue bucket. Catching a marble in the wrong bucket counts as a miss.

The game ends when you've managed to catch all the marbles on all seven screens.

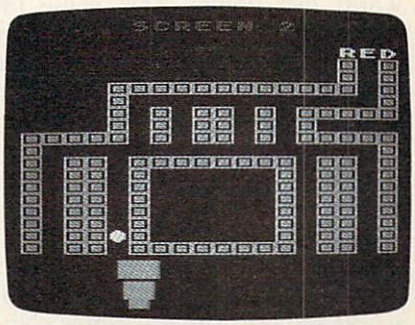
Commodore 64 Version

This version of Marbles (Program 2) is written in machine language, so you'll need to type it in using the 64 version of "MLX," the machine language entry program found elsewhere in this issue. Be sure you're familiar with MLX before you begin typing in Marbles. When MLX asks for starting and ending addresses, respond with the following:

Starting address: 0801
Ending address: 1500

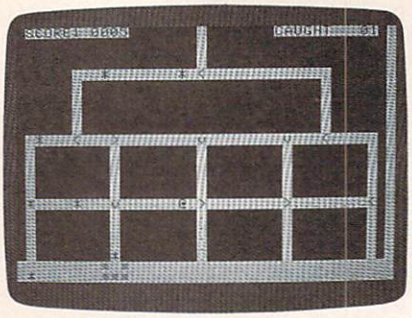


"Marbles," a challenging action game for the Atari 400, 800, XL, and XE.

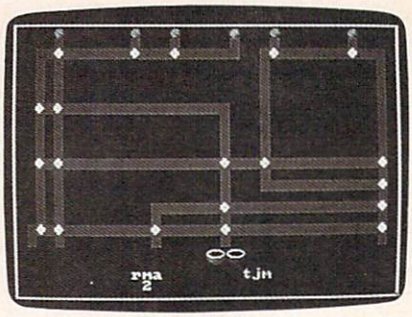


The Commodore 64 version of "Marbles."

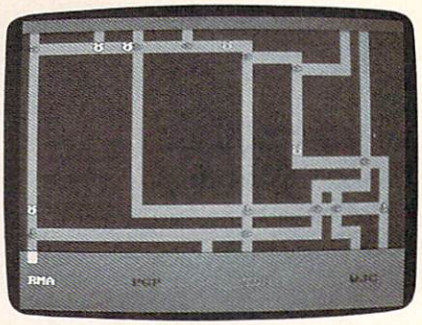
Type in the data, and be sure to save a copy to tape or disk before leaving MLX. Although the program is in machine language, it can be loaded and run like a BASIC program. When you're ready to play the game, simply load it and type RUN.



A paddle is used to control the bucket in the Apple II version of "Marbles."



The Amiga version of "Marbles" features random mazes.



"Marbles" for the IBM PC/PCjr.

Use a joystick plugged into port 2 to control a bucket at the bottom of the screen. The marbles in this game follow the arrows that occupy the intersections of the pipes. Press fire to change the direction in which the arrows point. (The arrows move clockwise.)

A feature unique to this version of the game is the gremlin, which is created whenever two different colored marbles collide. (Both marbles are destroyed when the gremlin is created.) If a gremlin collides with a marble, both are destroyed.

As gremlins travel around the maze, they sometimes change the direction of the arrows they cross.

You advance to the next screen (there are a total of nine) by scoring 1000 points. Each ball you catch is worth five points. This value increases whenever a gremlin drops through. If you catch a gremlin, the ball value goes back to five. Each screen allows you to miss a certain number of marbles (displayed on the screen as *passes*). If you drop any more than the limit, the game ends.

Apple II Version

Marbles for the Apple (Program 3) works on any Apple II-series computer, under either DOS 3.3 or ProDOS. A pair of paddles is required. Because the program is written entirely in machine language, you'll need to type it in using the Apple version of "MLX." When MLX asks for a starting and an ending address, respond with these values:

```
STARTING ADDRESS?   2000
ENDING ADDRESS?     2A17
```

Type in the data, and be sure to save a copy to disk before leaving MLX. When you're ready to play Marbles, enter BRUN *filename*, where *filename* is the name you used for the Marbles program.

This version of Marbles can be played as either a one- or two-player game. Exactly 40 marbles drop on each level. To make it to the next level, you must catch at least 20 of the marbles. If you do not advance to the next level, the game ends. Use the paddle to move the bucket. When you play the two-player game, players alternate turns. When one fails to advance, the other keeps playing.

When marbles drop into the maze, they appear as asterisks (*). When an asterisk travels over a dot, it turns into an *at* sign (@) and the dot disappears. You get points only for the latter type of marbles—one point for each dot the marble travels over. Each intersection has an arrow that determines the direction in which the marbles travel. Press the paddle button to change the direction of all the arrows on the screen. The arrows move clockwise. (Note: On Apple II and II+ models, alphabetic characters will appear in place of the arrows and buckets. This does not affect the operation of the game.)

Press the space bar to pause the game. Press it again to resume play.

Amiga Version

The Amiga version of Marbles (Program 4) is written in BASIC. Type it in and save it to disk. To play the game, simply load and run the program.

To play Marbles, you must have a Workbench 1.1 disk or the most recent version of Workbench 1.2. The first release of 1.2 does not properly allocate sprites 6 and 7.

A joystick plugged into port 1 controls the buckets at the bottom of the screen. You must catch the red marbles in the red bucket and the blue marbles in the blue bucket. The arrows at the intersections of the pipes control the flow of marbles. Press the fire button to change the direction of all the arrows on the screen. The arrows move counterclockwise.

This version of Marbles can be played by up to five players. The one-player game starts with three marbles on the screen at once. After every ten points, another marble is added until there are six marbles on the screen at once. The game ends when 40 marbles have been caught. Your score is the percentage of marbles you caught.

In the multiplayer game (two to five players), all contestants play the same level. After each has had a turn, the player with the lowest score is dropped from the game. When only one player remains, he or she is declared the winner. If two or more players tie for low score, the level is played again.

IBM PC/PCjr Version

This version of the game (Program 5) has the same rules as the Amiga version (described above), but there is only one bucket. However, you must change the color of the bucket to match the color of the ball you're trying to catch. Use the cursor keys to move the bucket left and right. The cursor-up key changes the color of the bucket. Press the space bar to change the directions of all the arrows on the screen.

For instructions on entering these programs, please refer to "COMPUTE!'s Guide to Typing In Programs" elsewhere in this issue.

Program 1: Atari 400, 800, XL, and XE Marbles

```
NO 1  REM COPYRIGHT 1987  COM
      PUTE! PUBLICATIONS, INC
        ALL RIGHTS RESERVED.
GB 10 GOTO 1510
FL 20 POKE 77,0
JE 30 ST=STICK(0):X=X+((ST=6
      OR ST=7 OR ST=5)-(ST=
      11 OR ST=9 OR ST=10))*
      8
BB 40 IF X<12 THEN X=12
IO 50 IF X>196 THEN X=196
KE 60 POKE 53248,X:POKE 5324
      9,X+24
AE 70 IX=SGN(XX(L)-MX):IY=SG
      N(YY(L)-MY):IF IX=0 AN
      D IY=0 THEN L=L+1:GOTO
       70
GG 80 POSITION MX,MY:? #6;"
      ":MX=MX+IX:MY=MY+IY:PO
      SITION MX,MY:? #6;M$
KG 90 SOUND 0,0,12,15:SOUND
      0,0,0,0
CP 100 IF MY=19 THEN 120
CN 110 GOTO 20
CN 120 IF M$="■" AND PEEK(53
       252)<>0 OR M$="■" AND
       PEEK(53253)<>0 THEN
       190
FO 130 FOR I=15 TO 0 STEP -1
       :SOUND 0,0,0,I:FOR D=
       1 TO 35:NEXT D:POKE 7
       09,PEEK(20):NEXT I
DP 140 POKE 709,6:IF SCORE=0
       THEN 170
DE 150 FOR I=SCORE TO 1 STEP
       -1:SOUND 0,I+100,10,
       15:POSITION I-1,1:? #
       6;" "
OE 160 SOUND 0,0,0,0:FOR D=1
       TO 35:NEXT D:NEXT I:
       SCORE=0
NF 170 POSITION MX,MY:? #6;"
       "
GK 180 GOTO 260
LL 190 SCORE=SCORE+1:POSITIO
       N SCORE-1,1:? #6;M$:F
       OR I=15 TO 0 STEP -1:
       SOUND 0,100,10,I
IC 200 FOR D=1 TO 5:NEXT D:N
       EXT I
NA 210 POSITION MX,MY:? #6;"
       "
AJ 220 IF SCORE=20 AND SCREE
       N=7 THEN 410
GE 230 IF SCORE=20 THEN FOR
       I=250 TO 50 STEP -1:S
       OUND 0,I,10,15:NEXT I
       :SOUND 0,0,0,0
BJ 240 IF SCORE=20 THEN FOR
       I=1 TO 250:NEXT I:GOT
       O 490
GI 250 GOTO 260
OM 260 POSITION 0,2:? #6;"
       {20 SPACES}"
CG 270 FOR I=1 TO 10:XX(I)=A
       SC(X$(I,I)):YY(I)=ASC
       (Y$(I,I))
LC 280 ST=STICK(0):X=X+((ST=
       7)-(ST=11))*8
EI 290 IF X<12 THEN X=12
LM 300 IF X>196 THEN X=196
NC 310 POKE 53248,X:POKE 532
       49,X+24
AC 320 IF XX(I)<100 THEN 350
KH 330 IF XX(I)=200 THEN XX(
       I)=XX(I-1):GOTO 350
DK 340 D=INT(RND(0)*(XX(I)-1
       00)):XX(I)=ASC(R$((I-
       1)*10+D+1))
CA 350 NEXT I
LD 360 D=INT(RND(0)*2):M$="■
       ":IF D=1 THEN M$="■"
BA 370 L=2:MX=XX(1):MY=YY(1)
       :POKE 53278,0:IF M$="
       ■" THEN POSITION MX-1
       ,2:? #6;"■■■"
CH 380 IF M$="■" THEN POSITI
       ON MX-1,2:? #6;"■■■"
AG 390 POSITION MX,MY:? #6;M
       $
CP 400 GOTO 20
MP 410 POSITION 3,10:? #6;"■
       ■■■■■■■■■■■■■■"
JL 420 POSITION 3,11:? #6;"■
          ■■■■■■■■■■■■■■"
ML 430 POSITION 3,12:? #6;"■
       {12 SPACES}■"
EG 440 POSITION 3,13:? #6;"■
       ■■■■■■■■■■■■■■"
NH 450 POSITION 3,14:? #6;"■
       ■■■■■■■■■■■■■■"
PG 460 FOR I=50 TO 250:SOUND
       0,I,10,15:NEXT I:SOU
       ND 0,0,0,0
JP 470 POKE 77,0:IF STRIG(0)
       =1 THEN 470
CI 480 POKE 53248,0:POKE 532
       49,0:RUN

LG 490 SCREEN=SCREEN+1
FH 500 POKE 559,0:POSITION 0
       ,0:? #6;"{CLEAR}":POS
       ITION 6,0:? #6;"SCREE
       N ";SCREEN:POSITION 0
       ,3
GP 510 READ I:IF I=-1 THEN 5
       30
DD 520 READ S$:FOR D=1 TO I:
       ? #6;S$:NEXT D:GOTO 5
       10
AD 530 READ X$,Y$:R$="{,}":R
       $(100)="{,}":R$(2)=R$
HF 540 READ I:IF I=-1 THEN 5
       60
DM 550 READ S$:R$((I-1)*10+1
       ,(I-1)*10+1+9)=S$:GOT
       O 540
IE 560 SCORE=0
IG 570 X=108:POKE 53248,X:PO
       KE 53249,X+24:POKE 53
       256,1:POKE 53257,1:PO
       KE 559,46:GOTO 260
AH 580 DATA 4, x x
JF 590 DATA 1, x xxxxxxxxxxx
       xxxx
PN 600 DATA 1, x{15 SPACES}x
CG 610 DATA 1, xxx xxx xxx x
       xx x
NH 620 DATA 9,{3 SPACES}x x
       x x x x x x
BB 630 DATA -1
EG 640 DATA {2 B}h■AAAAAA,
       {C}{2 H}{S}AAAAAA
DM 650 DATA 3,{D}{H}{L}{P}
       {6 ,}
BE 660 DATA -1
AF 670 DATA 2,{16 SPACES}x x
CN 680 DATA 1,{5 SPACES}xxxx
       xxxxxxxx x
AG 690 DATA 1,{5 SPACES}x
       {12 SPACES}x
LO 700 DATA 1,{5 SPACES}x x
       xx x xxxxx
NP 710 DATA 1,{5 SPACES}x x
       xx x x
KA 720 DATA 1, xxxxx x xx x
       xxxxx
AB 730 DATA 1, x{16 SPACES}x
KC 740 DATA 1, x xx xxxxxxxx
        xx x
NI 750 DATA 6, x xx x
       {6 SPACES}x xx x
KE 760 DATA 1, x xx xxxxxxxx
       xx x
BG 770 DATA -1
JN 780 DATA {2 Q}h■h■{4 ,},
       {C}{2 F}{2 J}{S}{4 ,}
PC 790 DATA 3,{F}{H}{K}{M}
       {6 ,},5,{B}{E}{N}{Q}
       {6 ,}
BA 800 DATA -1
PB 810 DATA 2,{5 SPACES}x x
       {4 SPACES}x x
LB 820 DATA 1,{3 SPACES}xxx
       xxxxxx xxx
AC 830 DATA 1,{3 SPACES}x
       {12 SPACES}x
MD 840 DATA 1,{3 SPACES}x xx
       x xx xxx x
NF 850 DATA 2,{3 SPACES}x x
       x xx x x x
KF 860 DATA 1, xxx xxx xx xx
       x xxx
AG 870 DATA 1, x{16 SPACES}x
LN 880 DATA 7, x xx xx xx xx
       xx x
BJ 890 DATA -1
KH 900 DATA f■h■j■{4 ,},{C}
       {2 F}{2 K}{S}{4 ,}
NF 910 DATA 1,{F}{M}{8 ,},3,
       {D}{H}{K}{O}{6 ,}
EP 920 DATA 5,{B}{E}{H}{K}
       {N}{Q}{4 ,}
BE 930 DATA -1
OH 940 DATA 4, x x
       {3 SPACES}x x
       {6 SPACES}x x
BN 950 DATA 1, x xxxxx xxxxx
       xxx x
AG 960 DATA 1, x{16 SPACES}x
LH 970 DATA 1, x xx xx xx xx
       xx x
AI 980 DATA 1, x{16 SPACES}x
KJ 990 DATA 1, xx xxx xxxx x
       xx xx
CI 1000 DATA 1, x{16 SPACES}x
NO 1010 DATA 6, x xx xx xx x
        x xx x
DL 1020 DATA -1
BP 1030 DATA g■j■h■j■{2 ,},
        {C}{2 H}{2 J}{2 L}
        {S}{2 ,}
BL 1040 DATA 1,{B}{H}{Q}{7 ,}
        ,3,{B}{E}{H}{K}{N}
        {Q}{4 ,}

CP 1050 DATA 5,{C}{G}{L}{P}
        {6 ,},7,{B}{E}{H}
        {K}{N}{Q}{4 ,}
DP 1060 DATA -1
OA 1070 DATA 2, x xx xx xx x
        x xx x
DA 1080 DATA 1, x{16 SPACES}x
OD 1090 DATA 3, x xx xx xx x
        x xx x
CJ 1100 DATA 1, x{16 SPACES}x
NM 1110 DATA 3, x xx xx xx x
        x xx x
CL 1120 DATA 1, x{16 SPACES}x
OA 1130 DATA 5, x xx xx xx x
        x xx x
DO 1140 DATA -1
CB 1150 DATA j■j■j■j■{2 ,},
        {C}{2 E}{2 I}{2 M}
        {S}{2 ,}
HI 1160 DATA 1,{B}{E}{H}{K}
        {N}{Q}{4 ,}
HL 1170 DATA 3,{B}{E}{H}{K}
        {N}{Q}{4 ,}
HO 1180 DATA 5,{B}{E}{H}{K}
        {N}{Q}{4 ,}
IB 1190 DATA 7,{B}{E}{H}{K}
        {N}{Q}{4 ,}
DL 1200 DATA -1
NM 1210 DATA 2, x xx xx xx x
        x xx x
CM 1220 DATA 1, x{16 SPACES}x
```

We Won't Be UNDERSOLD

1st In Price, 1st In Support, 1st In Warranty

15 Day Free Trial • 90 Day Immediate Replacement Policy • Free Catalogs

Call before you order, our prices may be lower • All sale prices expire 10-31-87

80 Column Printer• 8½" Letter Size

Big Blue Printer

Dot Matrix
Heat Transfer
Upper Case
Lower Case
Underline
Enlarged
Affordable
Plus More

No One Sells This Printer For Less!

Sale $29.95

List $199

(Add $7.50 Shipping*)

This printer was made by **Canon®** for IBM. The Big Blue printer comes ready to hook up to the serial port of the **IBM®** PC jr, XT, AT & Compatibles. Plus with our low cost adapter cables you can connect this printer to the **Apple®** II, IIe, IIc and Compatibles, **Atari®** computers, **Commodore®** 64, 128, SX-64, Vic 20, Plus 4, and more.

RS-232 Adapter - Adapter for IBM® PC, AT, XT & Apple® II series RS-232 port. (Specify male or female). . . List $49.95 **Sale $19.95**
Apple® IIc & Laser 128 Interface - With printer driver program for graphics and text. . . . . List $49.95 **Sale $19.95**
Intelligent Commodore® Interface - Prints graphics and text. Use Print Shop and more. . . . . List $49.95 **Sale $19.95**
Intelligent Atari® Interface - (Excludes Atari 1200) Prints graphics and text. Use Print Shop and more. . . . . List $49.95 **Sale $19.95**
Paper (2 Rolls) . . . . . List $19.95 **Sale $ 5.95** **Single Sheet Paper** (Qty. 500) . . . . . List $29.95 **Sale $12.95**

IBM, Apple, Canon, Commodore, Atari & Laser are registered trademarks of International Business Machines, Apple Computer, Canon Inc, Commodore Business Machines, Atari Inc. & Video Technologies Respectively.

Hi-Speed (160-180 CPS) Printer

NLQ-180 Printer

Near Letter Quality

** Lifetime Warranty

No One Sells This Printer For Less!

Sale $199

List $499

(Add $10.00 Shipping*)

• Near Letter Quality Selectable From Front Panel Controls • High Speed Dot Matrix
• Letter Quality Modes • 8K Buffer frees up computer 4-times faster • Super Graphics
• Pica, Elite, Italics, Condensed • Business or Personal • Tractor/Friction • 15 Day Free Trial
• ** Lifetime Warranty on Print Head • 6 Month Immediate Replacement Policy

INTERFACES

IBM $24.95 Apple II $44.95 Laser 128 $19.95 Commodore $29.95 Atari $39.95 Macintosh $49.95

COMPUTER DIRECT (A Division of **PROTECTO**)
22292 N. Pepper Rd., Barrington, IL. 60010
312/382-5050 or 312/382-5244
We Love Our Customers

* Illinois residents add 6½% sales tax. All orders must be in U.S. Dollars. We ship to all points in the U.S., CANADA, PUERTO RICO & APO-FPO. Please call for charges outside continental U.S. or C.O.D. **MAIL ORDERS** enclose cashier check, money order or personal check. Allow 14 days delivery, 2 to 7 for phone orders and 1 day express mail. Prices and availability subject to change without notice. (Monitors only shipped in continental U.S.)

VISA — MASTERCARD — C.O.D.

We Won't Be UNDERSOLD

15 Day Free Trial • 90 Day Immediate Replacement Policy • Free Catalogs

Call before you order, our prices may be lower • Sale prices expire 10-31-87

Complete Apple® Compatible System

Computer, Drive, Monitor, Printer & Software

Sale Price

$499

640K System Option $599

List $1228 (Add $35.00 shipping.*)

- **Laser 128K Computer**
  Video, parallel & serial ports included
  Runs virtually all Apple II/IIc/IIe software
- **Hi Res 12" Monochrome Monitor**
- **Big Blue 8½" Printer** with interface and paper
- **Software Package:** word processor, spreadsheet & data entry

Apple® is the registered trademark of Apple Computers Inc.

Hi-Speed NEQ Printer and Color Monitor Options Available

No One Sells These Systems For Less!

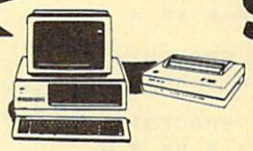
Professional IBM® XT Compatible System

Computer, Drive, Monitor, Printer & Software

SALE

Sale Price

$649



List $2495 (Add $35.00 shipping.*)

- **4.77/8 MHz / 512K Super Turbo XT Computer**
  Clock/calendar; Parallel, serial and game ports
- **MS DOS 3.2 & G.W. Basic** included
- **Hi-Res 12" Monochrome Monitor**
- **Big Blue 8½" Printer** with interface and paper
- **Software Package:** word processor, spreadsheet & data base

IBM® is the registered trademark of International Business Machines, Inc.

Complete C-64 System

Computer, Drive, Monitor & Printer

Sale Price

$359

List $1045 (Add $35.00 shipping.*)

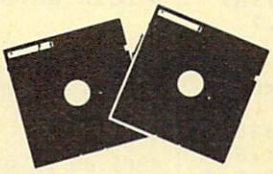
Sale

All this for $399 List

• **Commodore 64 Computer**	$299.00
• **Accelerator Disk Drive**	$229.00
• **Hi-Res 12" Monochrome Monitor**	$249.00
Monitor Cable	$ 19.95
• **Big Blue 8½" Printer** with	$199.00
Interface and 2 rolls of paper	$ 49.95
Total Prices	$1045.90

5¼" Floppy Disk Sale

Double-Sided • Double-Density



As Low As

24¢ each

100% Certified
Each disk is individually checked. (Add $3.00 shipping.*)

Lifetime Free Replacement Warranty
Against failure to perform due to faulty materials or workmanship.

Automatic Dust Remover
Makes sure dust and dirt are constantly removed during operation.

1 Box of 10	**$ 3.90 (39¢ each)**
1 Box of 50	**$14.50 (29¢ each)**
1 Box of 100	**$24.00 (24¢ each)**

Box of 100 comes without disk sleeves.
Paper Economy sleeves ..... 5¢ each.

Call

(312) 382-5244
For Atari & Commodore

(312) 382-5050
For IBM & Apple

Mail

COMPUTER DIRECT
22292 N. Pepper Road
Barrington, IL. 60010
We Love Our Customers

```
DO 1230 DATA 2, x x x x xx x
        x x x
CO 1240 DATA 1, x{16 SPACES}x
DA 1250 DATA 2, x xx xx xx x
        x xx x
DA 1260 DATA 1, x{16 SPACES}x
PC 1270 DATA 2, x x x x xx x
        x x x
DC 1280 DATA 1, x{16 SPACES}x
OG 1290 DATA 4, x xx xx xx x
        x xx x
DM 1300 DATA -1
BL 1310 DATA j[H]k[H]l[H]m[H]n,{C}
        {2 E}{2 H}{2 K}{2 N}
        {S}
EN 1320 DATA 1,{B}{E}{H}{K}
        {N}{Q}{4 ,},3,{B}
        {D}{F}{H}{K}{M}{O}
        {Q}{2 ,}
FG 1330 DATA 5,{B}{E}{H}{K}
        {N}{Q}{4 ,},7,{B}
        {D}{F}{H}{K}{M}{O}
        {Q}{2 ,}
IA 1340 DATA 9,{B}{E}{H}{K}
        {N}{Q}{4 ,}
EB 1350 DATA -1
PD 1360 DATA 3, x x x x xx x
        x x x
DC 1370 DATA 1, x{16 SPACES}x
PE 1380 DATA 2, x x x x xx x
        x x x
DE 1390 DATA 1, x{16 SPACES}x
ON 1400 DATA 2, x x x x xx x
        x x x
CN 1410 DATA 1, x{16 SPACES}x
OP 1420 DATA 2, x x x x xx x
        x x x
CP 1430 DATA 1, x{16 SPACES}x
PC 1440 DATA 3, x x x x xx x
        x x x
EC 1450 DATA -1
HP 1460 DATA l[H]m[H]n[H]o[H]p,{C}
        {2 F}{2 I}{2 L}{2 O}
        {S}
GG 1470 DATA 1,{B}{D}{F}{H}
        {K}{M}{O}{Q}{2 ,},3,
        {B}{D}{F}{H}{K}{M}
        {O}{Q}{2 ,}
GP 1480 DATA 5,{B}{D}{F}{H}
        {K}{M}{O}{Q}{2 ,},7,
        {B}{D}{F}{H}{K}{M}
        {O}{Q}{2 ,}
JJ 1490 DATA 9,{B}{D}{F}{H}
        {K}{M}{O}{Q}{2 ,}
DO 1500 DATA -1
BP 1510 GRAPHICS 0:POKE 559,
        0:POKE 16,64:POKE 53
        774,64
AF 1520 FOR I=53248 TO 53255
        :POKE I,0:NEXT I:FOR
         I=704 TO 707:POKE I
        ,0:NEXT I
CJ 1530 CLR :DIM FIL1$(1),FI
        L2$((INT(ADR(FIL1$)/
        1024)+1)*1024-ADR(FI
        L1$)-1)
FN 1540 DIM BUF$(384),MI$(12
        8),P0$(128),P1$(128)
        ,P2$(128),P3$(128)
JL 1550 BUF$(1)=CHR$(0):BUF$
        (384)=CHR$(0):BUF$(2
        )=BUF$:MI$=BUF$:P0$=
        BUF$:P1$=BUF$:P2$=BU
        F$:P3$=BUF$
EO 1560 DIM MOVE$(56):MOVE$=
        "hh[E]{B}h[E]{A}h
        [E]{D}h[E]{C}h[E]
        {F}h[E]{E}[,][E][P][1
        {A}[Q]{C}[G]{A}[F]{B}[G
        {B}[G]{C}[F]{B}[G]{D}[F]{E}[K
        {E}[I]{INSERT}[PHF]{F}[K
        {F}[I]{INSERT}[F][.]{.}"
OB 1570 CHSET=144*256
DL 1580 ML=USR(ADR(MOVE$),5.
        344,CHSET,1024)
LN 1590 RESTORE 1630
MJ 1600 GOTO 1620
CN 1610 FOR I=0 TO 7:READ D:
        POKE CHSET+A*8+I,D:N
        EXT I
DI 1620 READ A:IF A<>-1 THEN
         GOTO 1610
IN 1630 DATA 54,255,255,255,
        255,255,255,255,255
KI 1640 DATA 56,255,129,189,
        189,189,189,129,255
KP 1650 DATA 58,24,60,126,12
        6,126,126,60,24
EF 1660 DATA -1
LJ 1670 GRAPHICS 1+16:POKE 5
        59,0:POKE 16,64:POKE
         53774,64
FD 1690 ? :? :? :? "
        {6 SPACES}
        {5 SPACES}COPYRIGHT
        1987{8 SPACES}":POSI
        TION 8,4:? "COMPUTE!
        PUBLICATIONS, INC."
CO 1700 POSITION 8,5:? "
        {3 SPACES}ALL RIGHTS
        RESERVED.
        {4 SPACES}";:POSITIO
        N 10,12:? "PRESS BU
        TTON TO BEGIN";
DA 1730 POKE 559,34
AB 1740 POKE 77,0:IF STRIG(0
        )=1 THEN 1740
LI 1750 GRAPHICS 1+16:POKE 5
        59,0:POKE 16,64:POKE
         53774,64
KD 1760 POKE 709,6:POKE 756,
        144:POKE 708,196
FL 1770 POKE 54279,ADR(BUF$)
        /256:POKE 53277,3:PO
        KE 623,1
DD 1780 DIM IM$(16):IM$="
        {6 INSERT}{,}
        {6 BACK S}<<<"
JO 1790 P0$(95)=IM$:P1$(95)=
        IM$:POKE 559,46:POKE
         704,PEEK(710)-2:POK
        E 705,PEEK(711)-2
EL 1800 DIM XX(10),YY(10),S$
        (19),X$(10),Y$(10),R
        $(100),M$(1)
PC 1810 SCREEN=0:RESTORE 580
KB 1820 GOTO 490
```

Program 2: 64 Marbles

Version by Bill Chin, Editorial Programmer

```
0801:0B 08 0A 00 9E 32 30 36 2E
0809:31 00 00 00 4C AE 0A A2 85
0811:18 A9 00 9D 00 D4 CA 10 6A
0819:FA A9 0F 8D 18 D4 A9 FF 33
0821:8D 0F D4 A9 80 8D 12 D4 24
0829:60 8D 03 18 8A 48 98 48 9D
0831:AE 03 18 BC 4E 11 A9 00 32
0839:99 04 D4 BD 47 11 99 05 44
0841:D4 BD 55 11 99 06 D4 BD 33
0849:5C 11 D0 03 AD 1B D4 99 33
0851:01 D4 BD 63 11 99 04 D4 D0
0859:49 01 99 04 D4 68 A8 68 C3
0861:AA 60 A9 00 85 FC BD 20 CF
0869:C8 0A 0A 0A 85 FB 0A 26 98
0871:FC 0A 26 FC 18 65 FB 85 EA
0879:FB A5 FC 69 00 85 FC A5 DC
0881:FB 18 7D 29 C8 85 FB 85 B1
0889:FD 85 3B A9 04 65 FC 85 31
0891:FC 18 69 BC 85 3C 18 69 D5
0899:18 85 FE A0 28 60 30 01 25
08A1:60 49 FF 85 02 E6 02 A5 E1
08A9:02 60 A9 00 8D 00 C8 8D 93
08B1:08 C8 8D 07 18 AD 00 DC 6E
08B9:8D 04 18 4A B0 03 CE 08 70
08C1:C8 4A B0 03 EE 08 C8 4A 82
08C9:B0 03 CE 00 C8 4A B0 03 A0
08D1:EE 00 C8 AD 04 18 29 10 30
08D9:D0 03 EE 07 18 AD 07 18 FE
08E1:CD 08 18 D0 01 60 8D 08 97
08E9:18 AD 07 18 D0 F7 AC F3 87
08F1:C8 8C 09 18 AC 09 18 B9 9F
08F9:B7 C8 8D 00 18 EE 00 18 5E
0901:AD 00 18 29 03 AA B9 D5 8B
0909:C8 3D 09 12 F0 EF 8A 99 07
0911:B7 C8 98 AA 20 17 11 CE 3D
0919:09 18 10 D8 60 BD 20 C8 48
0921:CD 1A 12 90 05 A0 01 4C E4
0929:2F 09 BD 5A C8 A8 B9 20 EE
0931:12 9D 00 C8 B9 24 12 9D 60
0939:08 C8 BD 29 C8 9D 42 C8 D5
0941:BD 20 C8 9D 4A C8 BD 18 36
0949:C8 9D 3A C8 BD 10 C8 9D 58
0951:32 C8 BD 00 C8 18 7D 18 20
0959:C8 9D 18 C8 BD 00 C8 30 76
0961:0B BD 29 C8 69 00 9D 29 C9
0969:C8 4C 75 09 BD 29 C8 E9 40
0971:00 9D 29 C8 BD 08 C8 18 54
0979:7D 10 C8 9D 10 C8 BD 08 68
0981:C8 30 0B BD 20 C8 69 00 38
0989:9D 20 C8 4C 97 09 BD 20 CC
0991:C8 E9 00 9D 20 C8 BD 20 1C
0999:C8 CD 19 12 90 48 A9 FF C0
09A1:9D 20 C8 9D 4A C8 BD 62 D0
09A9:C8 30 24 A9 04 20 2A 08 48
09B1:CE E4 07 AD E4 07 C9 2F 26
09B9:D0 08 A9 39 8D E4 07 CE DB
09C1:E3 07 CE 20 18 D0 1F A9 4F
09C9:FF 8D F7 C8 4C E7 09 AC 8B
09D1:22 18 C0 63 F0 10 C8 8C 2F
09D9:22 18 A9 01 A0 09 20 F4 A6
09E1:09 A9 01 20 2A 08 BD 29 1B
09E9:C8 C9 28 90 05 A9 FF 9D 4D
09F1:20 C8 60 8D 00 18 B9 C0 BF
09F9:07 18 6D 00 18 99 C0 07 F2
0A01:A9 00 8D 00 18 B9 C0 07 CB
0A09:C9 3A 90 0C 38 E9 0A 99 7A
0A11:C0 07 EE 00 18 4C 06 0A 2D
0A19:88 B9 C0 07 C9 20 F0 05 1E
0A21:AD 00 18 D0 D1 60 A2 07 78
0A29:A0 0E BD 20 C8 10 0C BD 27
0A31:11 12 2D 15 D0 8D 15 D0 01
0A39:4C 94 0A BD 09 12 0D 15 75
0A41:D0 8D 15 D0 BD 10 C8 8D 1E
0A49:00 18 BD 20 C8 0E 00 18 B3
0A51:2A 0E 00 18 2A 0E 00 18 21
0A59:2A 69 2F 99 01 D0 BD 18 3B
0A61:C8 8D 00 18 BD 29 C8 0E F0
0A69:00 18 2A 0E 00 18 2A 18 76
0A71:69 06 8D 01 18 0E 00 18 8E
0A79:2A 99 00 D0 90 0C BD 09 4F
0A81:12 0D 10 D0 8D 10 D0 4C 8B
0A89:94 0A BD 11 12 2D 10 D0 69
0A91:8D 10 D0 88 88 CA 10 92 35
0A99:60 A9 07 8D 86 02 18 A2 11
0AA1:18 A0 02 20 F0 FF A9 B3 BA
0AA9:A0 11 4C 1E AB 20 10 08 C3
0AB1:20 CC 0B A9 6D A0 11 20 35
0AB9:1E AB 20 31 0E A9 00 8D 83
0AC1:7A C8 A9 0B 8D 20 D0 8D 47
0AC9:21 D0 A9 0F 8D 27 D0 20 93
0AD1:5F 0B 20 97 0F 20 9A 0A 0E
0AD9:20 2E 0C 20 AB 08 CE 05 2D
0AE1:18 D0 16 A9 04 8D 05 18 0C
0AE9:AD 00 C8 18 6D 29 C8 C9 DA
0AF1:27 90 03 AD 29 C8 8D 29 A9
0AF9:C8 20 A4 0C 20 27 0A AD 2F
0B01:8D 02 29 01 D0 F9 AD F7 55
0B09:C8 F0 D0 30 16 AC 7A C8 FD
0B11:C8 C0 09 90 02 A0 00 8C 05
0B19:7A C8 20 97 0F 20 2E 0C 7D
0B21:4C DC 0A 18 A2 15 A0 0F 11
0B29:20 F0 FF A9 D7 A0 11 20 A9
0B31:1E AB AD 15 D0 29 FE 8D FE
0B39:15 D0 20 E4 FF C9 43 D0 DE
0B41:0C 20 9A 0A 20 97 0F 20 F6
0B49:2E 0C 4C DC 0A C9 52 D0 BD
0B51:08 A9 00 8D 7A C8 4C 42 80
0B59:0B C9 51 D0 DD 00 A9 80 61
0B61:8D 00 18 A9 20 8D 72 0B 03
0B69:A2 00 8E 71 0B A9 00 9D 56
```

```
ØB71:ØØ 18 E8 DØ FA EE 72 ØB 3B
ØB79:AC 72 ØB CØ 25 DØ FØ AØ DE
ØB81:22 AD ØØ 18 C9 86 9Ø Ø2 21
ØB89:AØ 16 2Ø AB ØB EE ØØ 18 6Ø
ØB91:AD ØØ 18 C9 8E DØ E8 A9 51
ØB99:8D 8D F8 Ø7 A2 ØØ BD F6 FØ
ØBA1:12 9D 4Ø 23 E8 EØ 33 DØ 64
ØBA9:F5 6Ø 48 A9 ØØ 8D C1 ØB 3B
ØBB1:68 4A 6E C1 ØB 4A 6E C1 98
ØBB9:ØB 8D C2 ØB BD A6 12 99 Ø8
ØBC1:ØØ 18 E8 C8 C8 CØ 2C 9E
ØBC9:9Ø F2 6Ø AD ØE DC 29 FE Ø1
ØBD1:8D ØE DC A5 Ø1 29 FB 85 52
ØBD9:Ø1 AØ ØØ B9 ØØ DØ 99 ØØ AA
ØBE1:3Ø B9 ØØ D1 99 ØØ 31 C8 93
ØBE9:DØ F1 A5 Ø1 Ø9 Ø4 85 Ø1 ØE
ØBF1:AD ØE DC Ø9 Ø1 8D ØE DC C5
ØBF9:A2 ØØ A9 ØØ 85 FC BD 43 75
ØC01:12 ØA ØA 85 FB Ø6 FB 26 54
ØC09:FC A5 FC 69 3Ø 85 FC AØ 71
ØC11:ØØ E8 BD 43 12 91 FB C8 E6
ØC19:E8 CØ Ø8 DØ F5 EØ 59 9Ø 5A
ØC21:D9 AD 18 DØ 29 FØ 18 69 48
ØC29:ØC 8D 18 DØ 6Ø A2 Ø7 A9 FF
ØC31:FF 9D 2Ø C8 A9 8Ø 9D 1Ø DB
ØC39:C8 9D 18 C8 A9 ØØ 9D 29 5E
ØC41:C8 CA 1Ø EB 8D F7 C8 A9 B8
ØC49:14 8D 2Ø C8 8D 29 C8 A9 AB
ØC51:Ø5 8D 22 18 A9 35 8D C9 1C
ØC59:Ø7 A9 3Ø 8D C8 Ø7 8D E3 9F
ØC61:Ø7 8D E4 Ø7 AD D5 Ø7 8D CD
ØC69:23 18 AD 2Ø 18 C9 ØA 9Ø 5D
ØC71:Ø9 38 E9 ØA EE E3 Ø7 4C 5B
ØC79:6E ØC 18 6D E4 Ø7 8D E4 E8
ØC81:Ø7 6Ø BD 42 C8 9D 29 C8 E8
ØC89:BD 4A C8 9D 2Ø C8 BD 3A DF
ØC91:C8 9D 18 C8 BD 32 C8 9D EA
ØC99:1Ø C8 A9 ØØ 4C 2A Ø8 4C 88
ØCA1:45 ØE ØØ A2 Ø7 8E F6 C8 33
ØCA9:2Ø BA ØC EC F6 C8 DØ F2 4Ø
ØCB1:CA 8E F6 C8 EØ ØØ DØ FØ D7
ØCB9:6Ø BD 2Ø C8 3Ø E1 CD 1A CØ
ØCC1:12 9Ø ØA FØ Ø8 2Ø EØ ØD E6
ØCC9:BD 2Ø C8 3Ø D2 BD 62 C8 FF
ØCD1:1Ø Ø3 2Ø 64 ØF 2Ø 63 Ø8 C4
ØCD9:BD 52 C8 91 3B 2Ø 1E Ø9 37
ØCE1:BD 2Ø C8 3Ø 5B BD 1Ø C8 B7
ØCE9:C9 8Ø DØ 31 BD 18 C8 C9 DD
ØCF1:8Ø DØ 2A 2Ø 54 ØD BD 62 7A
ØCF9:C8 C9 81 FØ Ø5 A9 Ø1 8D 86
ØD01:1D 18 2Ø 7A ØD AD 1E 18 CE
ØD09:C9 ØF B0 11 29 Ø7 A8 B9 63
ØD11:62 C8 DD 62 C8 DØ Ø6 AD B3
ØD19:ØØ 18 9D 5A C8 2Ø 63 Ø8 28
ØD21:2Ø 42 ØD B1 3B 2Ø B0 ØE 62
ØD29:AE F6 C8 BD 2Ø C8 3Ø 1Ø E1
ØD31:AØ 28 BD 6A C8 91 3B 6Ø 67
ØD39:BD 5A C8 A8 B9 76 ØD A8 D6
ØD41:6Ø B1 3B C9 ØF B0 Ø8 29 7Ø
ØD49:Ø7 A8 B9 52 C8 AØ 28 9D 24
ØD51:52 C8 6Ø 8A 48 A9 ØØ 8D F1
ØD59:1F 18 A2 Ø3 A9 Ø1 9D 1A 34
ØD61:18 BD 76 ØD A8 B1 3B 3Ø 49
ØD69:Ø6 DE 1A 18 EE 1F 18 CA F1
ØD71:1Ø EA 68 AA 6Ø 29 5Ø 27 75
ØD79:ØØ BD 5A C8 49 Ø2 8D ØØ 48
ØD81:18 A9 Ø3 8D Ø1 18 AD 1B 2A
ØD89:D4 8D Ø2 18 AD Ø2 18 29 Ø2
ØD91:Ø3 A8 B9 1A 18 FØ 11 EE C5
ØD99:Ø2 18 CE Ø1 18 1Ø ED AD 2F
ØDA1:ØØ 18 9D 5A C8 4C B2 ØD Ø5
ØDA9:98 CD ØØ 18 FØ E9 9D 5A C9
ØDB1:C8 AØ 28 B1 3B C9 1Ø 9Ø 2A
ØDB9:1D C9 2D B0 19 38 E9 1Ø 13
ØDC1:A8 BD 62 C8 1Ø ØA 8A 48 7E
ØDC9:2Ø Ø3 11 68 AA 4C D7 ØD AØ
ØDD1:B9 B7 C8 9D 5A C8 2Ø 39 19
ØDD9:ØD B1 3B 8D 1E 18 6Ø 2Ø 59
ØDE1:63 Ø8 A9 7F 91 3B AD 29 DA
ØDE9:C8 38 FD 29 C8 2Ø 9F Ø8 D6
ØDF1:C9 Ø2 B0 3B BD 62 C8 3Ø 74
ØDF9:1D AØ 18 AD 22 18 2Ø F4 4F
ØE01:Ø9 A9 Ø3 2Ø 2A Ø8 AD D5 11
ØE09:Ø7 CD 23 18 FØ 1C A9 Ø1 4E
ØE11:8D F7 C8 4C 2B ØE A9 Ø5 B9
ØE19:8D 22 18 A9 3Ø 8D C8 Ø7 72
ØE21:A9 35 8D C9 Ø7 A9 Ø2 2Ø B0
ØE29:2A Ø8 A9 FF 9D 2Ø C8 6Ø FØ
ØE31:A4 A2 88 8C ØØ 18 A4 A2 6E
ØE39:CD ØØ 18 DØ F5 6Ø AØ ØØ BE
ØE41:C8 DØ FD 6Ø AD 1B D4 FØ 3Ø
ØE49:64 AD 1B D4 C9 27 BØ EE EE
ØE51:A8 B9 28 CØ C9 7F DØ E6 16
ØE59:98 9D 29 C8 A9 ØØ 9D 2Ø 83
ØE61:C8 A9 Ø1 9D 5A C8 A9 8Ø 1Ø
ØE69:9D 18 C8 9D 1Ø C8 A9 ØØ 44
ØE71:9D ØØ C8 A9 7F 9D 52 C8 EF
ØE79:A9 1Ø 9D Ø8 C8 A9 8C 9D 46
ØE81:F8 Ø7 8A 9D 6A C8 AD 17 EF
ØE89:18 9D 27 DØ AD 1B D4 29 B7
ØE91:Ø1 9D 62 C8 FØ Ø6 AD 18 81
ØE99:18 9D 27 DØ AD 1B D4 C9 68
ØEA1:1Ø BØ ØA A9 8Ø 9D 62 C8 D5
ØEA9:A9 8Ø 9D F8 Ø7 6Ø 6Ø 3Ø A8
ØEB1:FD DD 6A C8 FØ F8 C9 ØF 2C
ØEB9:BØ F4 29 Ø7 8D ØE 18 8E 64
ØEC1:ØB 18 BD 62 C8 8D ØC 18 F3
ØEC9:1Ø ØB AE ØE 18 BD 62 C8 AC
ØED1:3Ø DC 4C 26 ØF AE ØE 18 9Ø
ØED9:BD 62 C8 1Ø Ø3 4C 26 ØF 2C
ØEE1:CD ØC 18 DØ Ø3 4C EC ØE 29
ØEE9:4C 42 ØF AE ØB 18 BD 5A 18
ØEF1:C8 8D ØF 18 49 Ø2 9D 5A 21
ØEF9:C8 AE ØE 18 2Ø 63 Ø8 2Ø 28
ØF01:54 ØD AD ØF 18 A8 B9 1A 24
ØF09:18 DØ ØE AD ØF 18 9D 5A 72
ØF11:C8 A9 8Ø 9D 18 C8 9D 1Ø 17
ØF19:C8 AE ØB 18 2Ø 83 ØC 2Ø 71
ØF21:63 Ø8 4C 42 ØD AE ØE 18 F7
ØF29:A9 FF 9D 2Ø C8 AE ØB 18 Ø1
ØF31:A9 FF 9D 2Ø C8 BD 52 C8 84
ØF39:AØ 28 91 3B A9 Ø6 4C 2A BF
ØF41:Ø8 AE ØB 18 A9 FF 9D 2Ø 9A
ØF49:C8 A9 Ø5 2Ø 2A Ø8 AE ØE B5
ØF51:18 A9 81 9D 62 C8 A9 8Ø F9
ØF59:9D F8 Ø7 BD 6A C8 AØ 28 21
ØF61:91 3B 6Ø DE 72 C8 DØ 1A 83
ØF69:A9 ØC 9D 72 C8 BD F8 Ø7 7Ø
ØF71:C9 8B DØ ØB A9 89 9D F8 C9
ØF79:Ø7 2Ø 84 ØF 9D 27 DØ FE CE
ØF81:F8 Ø7 6Ø AD 2Ø DØ 29 ØF 6A
ØF89:8D 19 18 AD 1B D4 29 ØF 2Ø
ØF91:CD 19 18 FØ F6 6Ø A9 29 A4
ØF99:8D F4 ØF A9 13 8D F5 ØF Ø2
ØFA1:AC 7A C8 FØ 14 A9 34 18 A4
ØFA9:6D F4 ØF 8D F4 ØF AD F5 AB
ØFB1:ØF 69 ØØ 8D F5 ØF 88 DØ 58
ØFB9:EC A9 ØØ 8D 29 C8 8D 2Ø 39
ØFC1:C8 8D ØØ 18 A9 13 8D 1Ø ED
ØFC9:18 AC 7A C8 B9 31 12 8D 3F
ØFD1:16 18 B9 28 12 8D 15 18 C3
ØFD9:B9 3A 12 8D 2Ø 18 2Ø 84 A4
ØFE1:ØF 8D 17 18 2Ø 84 ØF CD 4E
ØFE9:17 18 FØ F8 8D 18 18 AC FØ
ØFF1:ØØ 18 B9 ØØ 18 8D ØE 18 78
ØFF9:A9 Ø9 8D ØF 18 CE ØF 18 Ø4
1ØØ1:DØ Ø6 EE ØØ 18 4C FØ ØF CB
1ØØ9:EE 1Ø 18 AD 1Ø 18 C9 14 ØB
1Ø11:DØ 13 A2 ØØ 8E 1Ø 18 2Ø B7
1Ø19:63 Ø8 EE 2Ø C8 AD 2Ø C8 D2
1Ø21:CD 1B 12 FØ 43 AC 1Ø 18 45
1Ø29:ØE ØE 18 9Ø ØC 2Ø 46 1Ø 5D
1Ø31:2Ø 5E 1Ø 2Ø 46 1Ø 4C FE Ø7
1Ø39:ØF 2Ø 55 1Ø 2Ø 5E 1Ø 2Ø 4F
1Ø41:55 1Ø 4C FE ØF A9 FF 91 3A
1Ø49:3B AD 15 18 91 FB AD 16 84
1Ø51:18 91 FD 6Ø A9 7F 91 3B 51
1Ø59:A9 2Ø 91 FB 6Ø 8C 11 18 B7
1Ø61:A9 27 38 ED 11 18 A8 6Ø AØ
1Ø69:CE 2Ø C8 A2 ØØ 2Ø 63 Ø8 8B
1Ø71:AØ 77 2Ø 55 1Ø 88 1Ø FA D6
1Ø79:A9 Ø1 8D F4 C8 8D F5 C8 EØ
1Ø81:A9 ØØ 8D F3 C8 AD F4 C8 17
1Ø89:8D 29 C8 AD F5 C8 8D 2Ø BC
1Ø91:C8 A2 ØØ 2Ø 63 Ø8 B1 3B 9A
1Ø99:C9 FF FØ 48 2Ø 54 ØD AD 5B
1ØA1:1F 18 C9 Ø3 9Ø 3E AC F3 8B
1ØA9:C8 AD 29 C8 99 7B C8 AD 45
1ØB1:2Ø C8 99 99 C8 A9 ØØ 99 67
1ØB9:D5 C8 A2 ØØ BD 1A 18 DØ A2
1ØC1:Ø9 B9 D5 C8 1D Ø9 12 99 E6
1ØC9:D5 C8 E8 EØ Ø3 DØ ED 2Ø 89
1ØD1:Ø3 11 AD 14 18 18 69 1Ø B2
1ØD9:91 3B EE F3 C8 AD F3 C8 5C
1ØE1:C9 1D FØ 1A EE F4 C8 AD 78
1ØE9:F4 C8 C9 27 DØ 97 A9 Ø1 9B
1ØF1:8D F4 C8 EE F5 C8 AD F5 42
1ØF9:C8 CD 1A 12 DØ 87 CE F3 8C
1101:C8 6Ø AD 1B D4 29 Ø3 AA Ø3
1109:B9 D5 C8 3D Ø9 12 FØ F2 CF
1111:8A 99 B7 C8 98 AA BD 7B C8
1119:C8 8D 31 C8 BD 99 C8 8D 29
1121:28 C8 BD B7 C8 8D 13 18 77
1129:8E 14 18 A2 Ø8 2Ø 63 Ø8 54
1131:AE 13 18 BD 6A 11 91 FB Ø5
1139:A9 Ø1 91 FD 6Ø AD 14 18 7C
1141:18 69 1Ø 91 3B 6Ø 11 45 A7
1149:45 ØØ ØØ BB 29 ØØ ØØ ØØ 13
1151:ØØ ØØ Ø7 Ø7 ØØ ØØ ØØ F2 B7
1159:F1 ØØ ØØ Ø4 ØØ ØØ 1C 16 Ø3
1161:Ø5 ØA 8Ø 1Ø 8Ø 11 11 1Ø 14
1169:8Ø 28 29 2A 93 11 2Ø 2Ø DE
1171:2Ø 4Ø 31 39 38 37 2Ø 43 8F
1179:4F 4D 5Ø 55 54 45 21 2Ø 1Ø
1181:5Ø 55 42 4C 49 43 41 54 5C
1189:49 4F 4E 53 2C 2Ø 49 4E E5
1191:43 2E ØD ØD 2Ø 2Ø 2Ø 2Ø 35
1199:2Ø 2Ø 2Ø 2Ø 2Ø 2Ø 41 4C 2A
11A1:4C 2Ø 52 49 47 48 54 53 28
11A9:2Ø 52 45 53 45 52 56 45 B3
11B1:44 ØØ 56 41 4C 55 45 2Ø 37
11B9:3Ø 3Ø 2Ø 2Ø 2Ø 53 43 4F 2A
11C1:52 45 2Ø 3Ø 3Ø 3Ø 3Ø 3Ø 38
11C9:3Ø 2Ø 2Ø 2Ø 5Ø 41 53 53 93
11D1:45 53 2Ø 58 58 ØØ 47 41 87
11D9:4D 45 2Ø 4F 56 45 52 ØD 66
11E1:1D 43 2Ø 54 4F 2Ø 43 4F 7D
11E9:4E 54 49 4E 55 45 2C 2Ø 8E
11F1:52 2Ø 54 4F 2Ø 52 45 53 EC
11F9:54 41 52 54 2C 2Ø 51 2Ø CA
1201:54 4F 2Ø 51 55 49 54 ØØ B4
1209:Ø1 Ø2 Ø4 Ø8 1Ø 2Ø 4Ø 8Ø 31
1211:FE FD FB F7 EF DF BF 7F 31
1219:16 13 15 12 27 14 EC 1Ø 44
1221:ØØ FØ ØØ ØØ 1Ø ØØ FØ 23 Ø7
1229:22 24 25 26 27 ØØ 26 22 16
1231:Ø3 Ø4 Ø5 Ø7 ØF ØA ØD Ø8 AB
1239:ØØ 32 28 1E 14 ØA Ø5 Ø4 A7
1241:Ø3 Ø2 ØØ 3C 42 99 A1 A1 88
1249:99 42 3C 22 DB C3 18 BD 5Ø
1251:BD 18 C3 DB 23 FF 44 FF 32
1259:91 FF 22 FF 91 24 FF FF A7
1261:C3 DB DB C3 FF FF 25 FF 6Ø
1269:18 FF 66 FF 18 FF 66 26 1A
1271:E7 DB A5 5A 5A A5 DB E7 E3
1279:27 5A A5 5A A5 5A A5 5A 5E
1281:A5 28 ØØ Ø8 ØC 7E 7E ØC 66
1289:Ø8 ØØ 29 ØØ 18 18 18 7E A6
1291:3C 18 ØØ 2A ØØ 1Ø 3Ø 7E 9B
1299:7E 3Ø 1Ø ØØ 2B ØØ 18 3C DØ
12A1:7E 18 18 18 ØØ 38 74 7C D5
12A9:38 38 7C 74 38 38 7C 6C D6
12B1:38 38 7C 5C 38 38 5C 7C 2D
12B9:38 38 6C 7C 38 ØØ ØØ ØØ 1F
12C1:38 7C 74 38 48 ØØ ØØ 38 AD
12C9:7C 74 38 48 24 ØØ 38 7C E2
12D1:74 38 48 48 44 38 7C F2 BA
12D9:7C 28 44 E2 Ø3 38 7C C6 B5
12E1:7C 28 44 42 E7 38 7C 9E B2
12E9:7C 28 44 27 7Ø 3C 7E FF C2
12F1:FF FF 7E 3C ØØ ØF FF EØ C6
12F9:38 ØØ 3C 6Ø ØØ Ø6 CØ ØØ 61
1301:Ø3 EØ ØØ Ø7 B8 ØØ 1D C7 19
1309:FF E3 78 ØØ 1E 7F FF F2 19
1311:3F FC ØC 3F FF F4 1F FF 9D
1319:D8 1F FF E8 ØF FC 1Ø ØF 9D
1321:FF DØ Ø7 EØ 2Ø Ø1 FF 8Ø EF
1329:FF BF FF FB FF FF 81 FF Ø2
1331:FF DF FF FD FE ØØ ØØ EF 17
1339:FF FE FF FF EØ ØØ FF FF 26
1341:EF FØ ØØ 3F 7F FB F7 FF 6B
1349:BF 7F FB F7 FF BF ØØ Ø3 3Ø
1351:FF FF BF FF FB FF FF BF ØF
1359:FF FB ØØ ØØ FF DF FF FD FB
1361:FF FF DF FF FD FF FF DF 53
```

```
1369:FF FC 00 FF DB FF FD BF 69
1371:FF DB FF FC 3F FF DB FF 10
1379:FD BF FF DB FF FD BF FF C3
1381:C3 FF FF BF FF FB FF FF 75
1389:BF FF FB FF FF BF 00 00 0E
1391:FF DF FF FD FF FF DF FF 4F
1399:FD FF FF DF FF FC 00 FF B0
13A1:DF BF FD FB FF DF BF FD 24
13A9:FB FF DF BF FC 03 FF DF 99
13B1:BF FD FB FF DF BF FD FB AC
13B9:FF DF BF FD FB FF DF BF 0F
13C1:FD FB FE 00 FF DF FF FD 43
13C9:FF FF DF FF 01 FF F7 FF E3
13D1:FF 7F FF F0 00 3F FF FB DF
13D9:FF FF BC 00 00 DF BF BD 54
13E1:FB FB C0 00 0F FB FB FF 7D
13E9:BF BF F8 00 FF BF BF FB 79
13F1:FB FF BF BF FB FB 00 00 D9
13F9:FD FB FF DF BF FD FB FF 0A
1401:DF 80 FD FB FF DF BF FD B5
1409:FB FF DF BF FD FB FF C0 C7
1411:03 FD FB BF DF BB FD FB 9B
1419:BF DF BB FC 03 BF DF FB 33
1421:FD FF BF DF FB FD FF BF D5
1429:DF FB 00 00 BF FB FB FF 26
1431:BF 80 00 0F 7F BF F0 00 27
1439:0F EF BF FE 00 0F FE BF C6
1441:80 00 0B FF BF BF FB F8 F8
1449:00 00 FB FB FF BF BF F8 28
1451:00 0F FB BF C0 3B FD FF A7
1459:BF DF FB FD FF BF 00 00 B7
1461:FF 7F FF F7 FF FF 3F FF 67
1469:F9 FF FF CF FF FE 7F FF 86
1471:F3 FF FF 9F FF FC FF FF 81
1479:E7 FF FF 3F FF FB F8 00 6B
1481:0F BF BF FB FB FF BF BF F7
1489:F8 03 FF BF BF FB FB FF D0
1491:BF BF F6 F9 FE FF 7F EF EE
1499:F7 FE FF 7F EF F7 FE FC CF
14A1:0F EF DD FE 01 DF EE FD 5C
14A9:FE EF DF 00 01 F7 EF DF F0
14B1:7E FD F0 00 1F 7E FD F7 9D
14B9:EF DF 70 FD F7 7F DF 77 B4
14C1:FD F7 7F DF 77 FD 00 00 88
14C9:FF FF EF FF FE FF FF 0F F6
14D1:FF E5 FF FC DF FF 9D FF 7D
14D9:F3 DF FE 71 FF CF 7F F9 23
14E1:F7 FF 3F 0F E7 E7 FC 00 B7
14E9:FF DB EF F1 BE FF 78 0F 1C
14F1:C7 BE FD FB EF DF 80 0D 3A
14F9:FB EF 00 00 FF FF 00 00 1C
```

Program 3: Apple II Marbles

Version by Tim Victor, Editorial Programmer

```
2000: A9 91 20 ED FD 20 CE 25 8F
2008: A9 00 8D A3 29 8D A8 29 03
2010: 8D A9 29 8D AD 29 8D AC 59
2018: 29 A9 01 8D AA 29 AC A4 48
2020: 29 F0 03 8D AB 29 8D AE 36
2028: 29 8D A2 29 A2 03 A9 B0 6C
2030: 9D CC 28 9D DA 28 CA 10 6E
2038: F7 20 06 26 AD A2 29 D0 BA
2040: 06 20 B8 26 4C B0 20 20 8A
2048: 58 FC A9 00 8D 9B 29 AE 05
2050: AE 29 BD A3 28 85 EC BD 13
2058: AA 28 85 ED 20 51 25 A9 C1
2060: 00 8D 9C 29 A9 26 8D 9D C8
2068: 29 A0 00 B1 EC 8D 9E 29 84
2070: E6 EC D0 02 E6 ED A9 08 E3
2078: 8D 9F 29 CE 9F 29 30 E9 65
2080: A9 AE 0E 9E 29 90 02 A9 26
2088: 20 AC 9C 29 91 EE AC 9D 69
2090: 29 91 EE EE 9C 29 CE 9D 5B
2098: 29 AD 9C 29 C9 14 D0 DB 1B
20A0: EE 9B 29 AD 9B 29 C9 16 6A
20A8: D0 B2 20 D5 23 20 9A 26 54
20B0: AD A3 29 D0 0A A9 28 A2 CC
20B8: C3 20 00 25 4C CB 20 A9 B0
20C0: 01 8D D2 28 A9 28 A2 D1 C6
20C8: 20 00 25 A9 28 A2 B5 20 AF
20D0: 00 25 20 27 24 A2 0F A9 44
20D8: FF 9D D4 29 20 47 25 9D B3
20E0: E4 29 CA 10 F2 A9 28 8D 54
20E8: A5 29 8D A6 29 A9 00 8D DF
20F0: A7 29 A9 00 8D B3 29 20 32
20F8: CE 21 AD A6 29 F0 52 AD 68
2100: 00 C0 10 16 2C 10 C0 C9 C2
2108: B3 F0 40 C9 A0 D0 0B 2C 77
2110: 10 C0 AD 00 C0 10 FB 2C AA
2118: 10 C0 AE A3 29 2C B2 29 2B
2120: 10 09 BD 61 C0 8D B2 29 45
2128: 4C 36 21 BD 61 C0 10 06 52
2130: 8D B2 29 20 03 23 A9 14 19
2138: 20 A8 FC 20 69 23 EE B3 BF
2140: 29 AD B3 29 C9 10 D0 AF 6B
2148: 4C F2 20 20 58 FC 4C D0 93
2150: 03 A9 08 A8 20 A8 FC 88 30
2158: D0 FA A2 03 AD A3 29 D0 65
2160: 0B BD 08 04 9D CC 28 CA 14
2168: 10 F7 30 09 BD 08 04 9D FA
2170: DA 28 CA 10 F7 AD A7 29 73
2178: C9 14 B0 17 AE A3 29 A9 2C
2180: 01 9D A8 29 A9 00 9D AA 85
2188: 29 CA 10 02 A2 01 BD AA 73
2190: 29 F0 28 A9 00 8D A2 29 E7
2198: AD A3 29 49 01 AA BD AA 2D
21A0: 29 F0 06 8E A3 29 8A D0 05
21A8: 0F EE AE 29 AD AE 29 C9 DA
21B0: 08 F0 08 A9 01 8D A2 29 7B
21B8: 4C 39 20 A9 01 8D AD 29 D0
21C0: 20 06 26 AD AC 29 D0 03 E2
21C8: 4C 4B 21 4C 05 20 AE B3 A6
21D0: 29 BD D4 29 10 33 AD A5 92
21D8: 29 F0 2D DE E4 29 D0 28 15
21E0: A9 00 9D B4 29 8D 9B 29 D6
21E8: 9D 04 2A A9 13 9D C4 29 9C
21F0: 8D 9C 29 20 82 24 B0 10 5E
21F8: AC 9C 29 B1 EE 9D F4 29 F9
2200: A9 01 9D D4 29 CE A5 29 53
2208: 60 BD C4 29 8D 9A 29 BD FD
2210: B4 29 8D 99 29 C9 16 B0 91
2218: 78 BD D4 29 20 65 24 B0 C4
2220: 19 BD F4 29 20 A2 24 10 75
2228: 0B AE B3 29 BD D4 29 49 83
2230: 02 20 C2 23 AE B3 29 9D 3C
2238: D4 29 BD D4 29 20 D7 24 D3
2240: 20 82 24 90 09 BD D4 29 D4
2248: 49 02 9D D4 29 60 AD 9B 74
2250: 29 9D B4 29 AD 9C 29 9D 89
2258: C4 29 AD 99 29 8D 9B 29 78
2260: 20 51 25 AC 9A 29 BD F4 62
2268: 29 C9 AE D0 05 FE 04 2A EC
2270: A9 A0 91 EE BD B4 29 8D 73
2278: 9B 29 10 03 4C E5 22 C9 0F
2280: 16 B0 46 20 51 25 BC C4 24
2288: 29 B1 EE 9D F4 29 4C BC 27
2290: 22 20 F4 22 90 4F AD 99 65
2298: 29 8D 9B 29 20 51 25 AC 18
22A0: 9A 29 A9 A0 91 EE EE 99 7B
22A8: 29 AD 99 29 C9 18 F0 35 78
22B0: 9D B4 29 8D 9B 29 20 51 02
22B8: 25 AC 9A 29 BD 04 2A F0 E3
22C0: 03 A9 C0 2C A9 AA 91 EE D5
22C8: 60 20 F4 22 B0 E8 EE A7 B4
22D0: 29 20 BC 25 AE B3 29 BC E8
22D8: 04 2A F0 09 20 71 25 88 F1
22E0: D0 FA AE B3 29 CE A6 29 58
22E8: A9 FF 9D D4 29 20 47 25 80
22F0: 9D E4 29 60 38 AD 9A 29 3F
22F8: ED AF 29 90 04 C9 03 90 2C
2300: 01 38 60 A9 01 8D 99 29 16
2308: A9 01 8D 9A 29 20 94 24 D5
2310: 30 17 20 C2 23 AD 99 29 90
2318: 8D 9B 29 20 51 25 AC 9A 46
2320: 29 AE A0 29 BD B2 28 91 E7
2328: EE EE 9A 29 AD 9A 29 C9 7B
2330: 26 D0 DA EE 99 29 AD 99 6E
2338: 29 C9 15 D0 CB A0 10 8C C2
2340: A1 29 B9 B4 29 8D 99 29 FF
2348: B9 C4 29 8D 9A 29 B9 F4 7C
2350: 29 20 A2 24 30 0F 20 C2 8A
2358: 23 AE A0 29 BD B2 28 AC 38
2360: A1 29 99 F4 29 88 10 D7 A7
2368: 60 A2 00 20 25 25 48 18 EF
2370: 6D B0 29 6A 20 A8 25 CD 21
2378: AF 29 D0 02 68 60 48 A9 1A
2380: A0 20 92 23 68 8D AF 29 A5
2388: A9 FF 20 92 23 68 8D B0 57
2390: 29 60 8D B1 29 A9 16 8D F9
2398: 9B 29 20 51 25 AC AF 29 74
23A0: AD B1 29 91 EE C8 AE 9B FB
23A8: 29 E0 17 F0 02 A9 A0 91 37
23B0: EE AD B1 29 C8 91 EE EE FB
23B8: 9B 29 AD 9B 29 C9 18 D0 F7
23C0: D9 60 38 E9 01 10 02 A9 A7
23C8: 02 8D A0 29 20 C6 24 AD 2C
23D0: A0 29 B0 EF 60 A9 01 8D FF
23D8: 99 29 A9 01 8D 9A 29 AD 52
23E0: 99 29 8D 9B 29 20 51 25 3B
23E8: AC 9A 29 B1 EE C9 20 F0 3C
23F0: 21 20 4B 24 C9 03 90 1A 11
23F8: 20 AE 24 8D A0 29 AD 99 F6
2400: 29 8D 9B 29 20 51 25 AC 83
2408: 9A 29 AE A0 29 BD B2 28 95
2410: 91 EE EE 9A 29 AD 9A 29 C2
2418: C9 26 D0 C3 EE 99 29 AD 03
2420: 99 29 C9 15 D0 B4 60 A9 CD
2428: 00 8D AF 29 8D B0 29 A9 87
2430: 16 8D 9B 29 20 51 25 A0 1E
2438: 02 A9 FF 91 EE 88 10 FB BA
2440: EE 9B 29 AD 9B 29 C9 18 14
2448: D0 EA 60 A2 00 8E A0 29 8E
2450: AD A0 29 20 C6 24 B0 01 E7
2458: E8 EE A0 29 AD A0 29 C9 83
2460: 04 D0 ED 8A 60 4A B0 0D DF
2468: A9 01 20 C6 24 90 12 A9 67
2470: 03 20 C6 24 60 A9 00 20 27
2478: C6 24 90 05 A9 02 20 C6 EB
2480: 24 60 20 51 25 AC 9C 29 4A
2488: B1 EE C9 AA F0 05 C9 C0 39
2490: F0 01 18 60 AD 99 29 8D 4E
2498: 9B 29 20 51 25 AC 9A 29 4C
24A0: B1 EE A2 02 DD B2 28 F0 EC
24A8: 03 CA 10 F8 8A 60 20 47 14
24B0: 25 2A 2A 2A 29 03 C9 03 E9
24B8: F0 F4 8D A0 29 20 C6 24 ED
24C0: B0 EC AD A0 29 60 20 D7 3F
24C8: 24 20 51 25 AC 9C 29 B1 83
24D0: EE C9 20 F0 01 18 60 AC EB
24D8: 9A 29 4A 90 13 4A B0 04 32
24E0: A9 01 D0 02 A9 FE 6D 99 36
24E8: 29 8D 9B 29 8C 9C 29 60 B8
24F0: 4A B0 02 88 24 C8 8C 9C 4D
24F8: 29 AD 99 29 8D 9B 29 60 94
2500: 8D 23 25 20 21 25 8D 9B D4
2508: 29 20 51 25 20 1B 25 A8 CB
2510: 10 03 91 EE C8 20 1B 25 66
2518: D0 F8 60 E8 D0 03 EE 23 37
2520: 25 BD 00 FF 60 AD A3 29 96
2528: 18 69 64 8D 3D 25 AD 70 88
2530: C0 A2 7F A9 00 A0 01 85 18
2538: FC 4C 3C 25 EC 64 C0 65 CD
2540: FC C8 D0 F8 49 FF 60 A5 95
2548: 4E 0A 0A 38 65 4E 85 4E BE
2550: 60 AD 9B 29 29 07 18 69 3B
2558: 08 4A 85 EF A9 00 6A AC B7
2560: 9B 29 C0 08 90 08 69 27 F9
2568: C0 10 90 02 69 27 85 EE 2B
2570: 60 AD AE 29 18 A2 03 7D 8D
2578: 08 04 C9 BA 90 0A E9 0A 37
2580: 9D 08 04 CA A9 00 F0 EF E7
2588: 9D 08 04 60 A0 B0 AD A7 F4
2590: 29 C9 0A 90 05 C8 E9 0A 55
2598: B0 F7 69 B0 8D 25 04 98 13
25A0: D0 02 A9 A0 8D 24 04 60 78
25A8: 8D B3 25 A9 00 85 FC A2 98
25B0: 04 18 69 00 6A 66 FC 4A 61
25B8: 66 FC 4A 66 FC CA D0 F1 CB
25C0: AA A5 FC 4A 4A 4A 4A 4A 68
25C8: C9 07 8A 69 00 60 20 58 BB
25D0: FC A9 28 A2 DF 20 00 25 D7
25D8: A9 28 A2 E9 20 00 25 A9 E9
25E0: 29 A2 11 20 00 25 A9 29 9D
25E8: A2 27 20 00 25 2C 10 C0 0D
25F0: AD 00 C0 10 FB 2C 10 C0 9C
25F8: C9 B1 F0 04 C9 B2 D0 F0 9E
2600: E9 B1 8D A4 29 60 20 58 0D
2608: FC A9 28 A2 C3 20 00 25 30
2610: AD A4 29 F0 0C A9 1B 8D 5B
2618: D2 28 A9 28 A2 D1 20 00 2C
2620: 25 A2 00 BD A8 29 D0 06 15
2628: E8 BD A8 29 F0 13 A9 00 27
2630: 9D A8 29 8A 18 69 B1 8D 9A
2638: 78 29 A9 29 A2 6F 20 00 E5
2640: 25 AD AD 29 D0 29 AD AE 08
2648: 29 18 69 B0 8D 4B 29 A9 FC
```

From the publishers of *COMPUTE!*

October 1987 *COMPUTE! Disk*

All the exciting programs from the past three issues of *COMPUTE!* are on timesaving, error-free, floppy disks that are ready to load on your IBM PC and PCjr or Commodore 64 and 128. The October 1987 *COMPUTE! Disks* contain the entertaining and useful Commodore or IBM programs from the August, September, and October 1987 issues of *COMPUTE!*.

The October 1987 *COMPUTE! Disk* costs $12.95 plus $2.00 shipping and handling and is available only from COMPUTE! Publications. Please specify whether you need a Commodore or IBM disk.

For added savings and convenience, you may also subscribe to the *COMPUTE! Disk.* At a cost of only $39.95 a year (a $12.00 savings), you'll receive four disks, one every three months. Each disk will contain all the programs for your machine from the previous three issues of *COMPUTE!*. To order a subscription, call toll free 1-800-727-6937.

This is an excellent way to build your software library while you enjoy the quality programs from *COMPUTE!*.

Disks and subscriptions are available for Apple, Atari, Commodore 64 and 128, and IBM personal computers. Call for details.

For more information or to order individual issues of the October 1987 *COMPUTE! Disk,* call toll free 1-800-346-6767 (in NY 212-887-8525) 10:00 AM–12:30 PM or 1:30 PM–3:00 PM EST, or write *COMPUTE! Disk*, P.O. Box 5038, F.D.R. Station, New York, NY 10150.

COMPUTE! Publications, Inc. abc

Part of ABC Consumer Magazines, Inc.
One of the ABC Publishing Companies
825 7th Avenue, 6th Floor, New York, NY 10019

Publishers of COMPUTE!, COMPUTE!'s Gazette, COMPUTE!'s Gazette Disk, COMPUTE! Books, COMPUTE!'s Apple Applications, and COMPUTE!'s Atari ST Disk & Magazine.

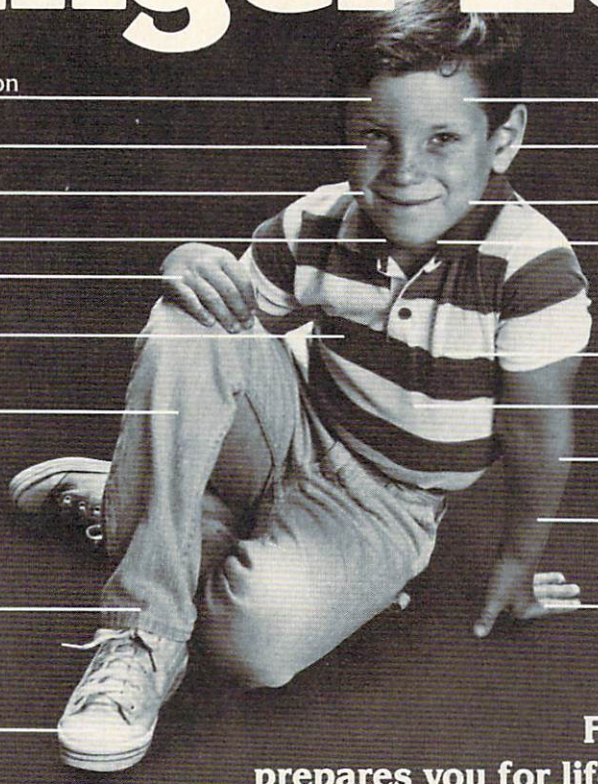
```
2650: 29 A2 43 20 00 25 AD A3 D7
2658: 29 18 69 B1 8D 56 29 A9 49
2660: 29 A2 4D 20 00 25 AE A3 2B
2668: 29 BD 61 C0 10 FB 60 A9 CB
2670: 29 A2 85 20 00 25 2C 10 A9
2678: C0 AD 00 C0 10 FB 2C 10 75
2680: C0 C9 F9 F0 0F C9 D9 F0 32
2688: 0B C9 EE F0 04 C9 CE D0 6F
2690: E8 A9 00 2C A9 01 8D AC 97
2698: 29 60 A2 00 BD 00 04 9D 79
26A0: 00 0C BD 00 05 9D 00 0D 53
26A8: BD 00 06 9D 00 0E BD 00 22
26B0: 07 9D 00 0F E8 D0 E5 60 8F
26B8: A2 00 BD 00 0C 9D 00 04 E8
26C0: BD 00 0D 9D 00 05 BD 00 F6
26C8: 0E 9D 00 06 BD 00 0F 9D 8D
26D0: 00 07 E8 D0 E5 60 FF FF B9
26D8: E0 FF FF E0 FF E0 00 FF 27
26E0: EF F0 FF EF F0 FC 00 70 4C
26E8: FD FF 70 FD FF 70 FD FC DC
26F0: 00 FD FD F0 80 1D F0 BF A5
26F8: DD F0 BF DD F0 BF C0 00 4E
2700: BF DF E0 BF DF E0 BF DF 20
2708: E0 80 00 00 BE FB E0 BE 4D
2710: FB E0 BE FB E0 BE FB E0 07
2718: FF FF E0 FF FF E0 FF C0 C6
2720: 00 FF DF F0 FF DF F0 F8 D3
2728: 00 F0 FB FE F0 FB FE F0 88
2730: 80 78 00 BF 7B F0 83 7B FA
2738: F0 FB 7B F0 83 78 00 BF 3A
2740: 7B F0 83 7B F0 FB 7B F0 10
2748: 83 40 70 BF 5F 70 BF 5F 0E
2750: 70 80 00 00 BE FB E0 BE 5D
2758: FB E0 FF FF E0 FF FF E0 C4
2760: FF F0 00 FF F7 F0 FF F7 66
2768: F0 FF F7 F0 F0 00 00 F7 BC
2770: DF 70 F7 DF 70 F7 DF 70 5B
2778: 80 00 00 BB BB B0 BB BB 96
2780: B0 BB BB B0 80 00 00 AA 47
2788: AA A0 AA AA A0 AA AA A0 F9
2790: 80 00 00 BF 7E F0 BF 7E D0
2798: F0 BF 7E F0 FF FF E0 FF EF
27A0: FF E0 80 00 00 BF 7E F0 24
27A8: BF 7E F0 BF 7E F0 BF 7E 46
27B0: F0 80 00 00 BD EF 70 BD E3
27B8: EF 70 BD EF 70 80 00 00 57
27C0: B6 DB 60 B6 DB 60 B6 DB 82
27C8: 60 80 00 00 BB BB B0 BB 51
27D0: BB B0 80 00 00 BD EF 70 80
27D8: BD EF 70 BD EF 70 FF FF 2D
27E0: E0 FF FF E0 FF FE 00 FF A9
27E8: FE E0 FF FE E0 FF FE E0 C4
27F0: FF E0 00 FF EE E0 FF EE 61
27F8: E0 FF EE E0 FF 80 00 FF A5
2800: BB B0 FF BB B0 FF BB B0 C3
2808: E0 00 00 EE EE E0 EE EE 7F
2810: E0 EE EE E0 80 00 00 BB 38
2818: BB B0 BB BB B0 BB BB B0 42
2820: FF FF E0 FF FF E0 FF 80 90
2828: 00 FF BF F0 FF BF F0 F0 51
2830: 01 F0 F7 FD F0 F7 FD F0 70
2838: 80 E0 30 BE EF B0 BE EF A2
2840: B0 BE EF B0 80 E0 30 B6 40
2848: ED B0 B6 ED B0 80 00 00 F8
2850: BE FB E0 B8 3B 80 BB BB B5
2858: B0 80 00 00 BE FB E0 BE 87
2860: FB E0 FF FF E0 FF FF E0 CE
2868: FF FF E0 FF FF E0 FC 00 52
2870: 00 FD FF F0 FD FF F0 FD 1F
2878: FF F0 FD FF F0 FD FF F0 35
2880: 80 00 00 BF DF E0 BF DF EE
2888: E0 BF DF E0 BF DF E0 BF 42
2890: DF E0 80 00 00 BF DF E0 B8
2898: BF DF E0 BF DF E0 BF DF BA
28A0: E0 BF DF E0 62 D6 18 20 1A
28A8: DE 9C 5A 28 26 27 28 27 A2
28B0: 27 27 BC F6 BE 00 1B C3 55
28B8: C1 D5 C7 C8 D4 A0 A0 A0 EF
28C0: A0 B0 00 00 01 D3 C3 CF 3C
28C8: D2 C5 B1 A0 B0 B0 B0 B0 8E
28D0: 00 00 1B D3 C3 CF D2 C5 8A
28D8: B2 A0 B0 B0 B0 B0 00 03 17
28E0: 10 CD C1 D2 C2 CC C5 D3 BA
28E8: 00 04 01 C3 CF D0 D9 D2 DE
28F0: C9 C7 C8 D4 A0 B1 B9 B8 76
28F8: B7 AC A0 C3 CF CD D0 D5 CD
2900: D4 C5 A1 A0 D0 D5 C2 CC 9C
2908: C9 C3 C1 D4 C9 CF CE D3 B4
2910: 00 05 0A C1 CC CC A0 D2 AE
2918: C9 C7 C8 D4 D3 A0 D2 C5 34
2920: D3 C5 D2 D6 C5 C4 00 0C E2
2928: 07 CF CE C5 A0 CF D2 A0 B2
2930: D4 D7 CF A0 D0 CC C1 D9 FD
2938: C5 D2 D3 BF A0 BC B1 AC A0
2940: B2 BE 00 09 10 CC C5 D6 42
2948: C5 CC A0 B0 00 0C 04 D0 D8
2950: CC C1 D9 C5 D2 A0 B0 BA 46
2958: A0 D0 D2 C5 D3 D3 A0 C2 D7
2960: D5 D4 D4 CF CE A0 D4 CF DC
2968: A0 C2 C5 C7 C9 CE 00 0E 88
2970: 0A D0 CC C1 D9 C5 D2 A0 DD
2978: B0 BA A0 C7 C1 CD C5 A0 D3
2980: CF D6 C5 D2 00 14 0B D0 8D
2988: CC C1 D9 A0 C1 C7 C1 C9 71
2990: CE BF A0 BC D9 AC CE BE F7
2998: 00 00 00 00 00 00 00 00 EA
29A0: 00 00 00 00 00 00 00 00 F2
29A8: 00 00 00 00 00 00 00 00 FA
29B0: 00 00 00 00 00 00 00 00 03
29B8: 00 00 00 00 00 00 00 00 0B
29C0: 00 00 00 00 00 00 00 00 13
29C8: 00 00 00 00 00 00 00 00 1B
29D0: 00 00 00 00 00 00 00 00 23
29D8: 00 00 00 00 00 00 00 00 2B
29E0: 00 00 00 00 00 00 00 00 33
29E8: 00 00 00 00 00 00 00 00 3B
29F0: 00 00 00 00 00 00 00 00 43
29F8: 00 00 00 00 00 00 00 00 4B
2A00: 00 00 00 00 00 00 00 00 54
2A08: 00 00 00 00 00 00 00 00 5C
2A10: 00 00 00 00 49 53 4D 41 D7
```

Program 4: Amiga Marbles

Version by Tim Midkiff, Editorial Programmer

```
'Marbles
'Copyright 1987 COMPUTE! Publications, Inc.
'All Rights Reserved.
CLEAR ,25000:CLEAR ,50000&
DEFINT a-z:RANDOMIZE TIMER
LIBRARY "graphics.library"
DECLARE FUNCTION GetSprite&() LIBRARY
DECLARE FUNCTION ALLocRaster&() LIBRARY
DIM sc(21,40),pno(21,40),pipmap(21,40),arrow(24,2),xi(30),yi(30)
DIM di(30),ti(30),fbb(12),fbr(12),w(255)
SCREEN 1,320,200,4,1:WINDOW 3,,(0,0)-(311,186),16,1:WINDOW OUTPUT 3
RESTORE CoLors:FOR i=0 TO 15:READ r!,g!,b!:PALETTE i,r!,g!,b!:NEXT
CoLors:
DATA 0,0,0,.7,.7,.7,.3,.3,.3,.8,0,.8,.6,0,.6,.2,.2,.2,0,0,.8,0,0,.65
DATA 0,0,.5,0,0,.4,.65,0,0,.5,0,0,.4,0,0,.25,0,0,.5,.5,.5,.6,.6,.6
PALETTE 22,0,0,1:PALETTE 23,0,0,.6:PALETTE 26,1,0,0
PALETTE 27,.6,0,0:PALETTE 30,0,0,1:PALETTE 31,0,0,.6
COLOR 1,5:CLS:GOSUB InitSprites
ON BREAK GOSUB CLoseSprites:BREAK ON
DirDat:RESTORE DirDat:FOR i=0 TO 2:READ dx(i),dy(i):NEXT:DATA -1,0,0,1,1,0
RESTORE LeftArrow:FOR i=0 TO 24:READ a$:arrow(i,0)=VAL("&H"+a$):NEXT
RESTORE DownArrow:FOR i=0 TO 24:READ a$:arrow(i,1)=VAL("&H"+a$):NEXT
RESTORE RightArrow:FOR i=0 TO 24:READ a$:arrow(i,2)=VAL("&H"+a$):NEXT
RESTORE CupData:FOR i=1 TO 202:READ a:cup$=cup$+CHR$(a):NEXT
OBJECT.SHAPE 1,cup$
LINE(2,0)-(6,0),7:LINE(0,1)-(8,1),8:GET(0,0)-(8,1),fbb
LINE(2,0)-(6,0),11:LINE(0,1)-(8,1),12:GET(0,0)-(8,1),fbr
NewGame:COLOR 1,5:CLS:PRINT TAB(12) "Copyright 1987":
PRINT TAB(7) "COMPUTE! Publications, Inc.":PRINT TAB(10) "All Rights Reserved."
PRINT:PRINT:PRINT"Number of Players (1-5)"
PlNum:k$="":WHILE k$="":k$=INKEY$:WEND:IF k$<"1" OR k$>"5" THEN PlNum
CLS:numpL=VAL(k$)-1:pts=0:pn=numpL
FOR i=0 TO 255:w(i)=INT(RND*255-128):NEXT:WAVE 3,w
IF numpL=0 THEN
LOCATE 23,6:PRINT"Marbles:":LOCATE 23,25:PRINT"Rating:":npip=3
ELSE
LOCATE 10,1:PRINT"Enter the initials of player"
FOR i=0 TO numpL:LOCATE 10,29:PRINT LEFT$(STR$(i+1),2);:INPUT nam$(i)
nam$(i)=LEFT$(nam$(i)+SPACE$(3),3):LOCATE 10,29:PRINT SPACE$(11):NEXT
CLS:ofst=INT(34/numpL):npip=7-numpL
FOR i=0 TO numpL:cp(i)=i:LOCATE 23,i*ofst+2:PRINT nam$(i):NEXT
FOR i=0 TO numpL:j=INT(RND*(numpL+1)):SWAP cp(i),cp(j):NEXT
END IF
NewMap:
FOR i=0 TO 2:ck(i)=0:NEXT
FOR i=2 TO 20:FOR j=2 TO 38:sc(i,j)=0:pno(i,j)=0:pipmap(i,j)=0:NEXT j,i
IF numpL>0 THEN
LOCATE 23,cp(pn)*ofst+2:COLOR 4,5:PRINT nam$(cp(pn))
LOCATE 21,1:PRINT SPACE$(40):IF pn=numpL THEN LOCATE 24,1:PRINT SPACE$(39);
END IF
pipe=0:LINE(0,8)-(312,159),0,bf
NewPipe:
row=2:pipe=pipe+1:COLOR 1,2
Start:
coL=INT(RND*37)+2:IF (sc(2,coL-1) OR sc(2,coL) OR sc(2,coL+1))>0 THEN Start
GOSUB Down:IF Ln<3 THEN Start
ballx(pipe-1)=coL:baLLy(pipe-1)=row:pipmap(2,coL)=2
LOCATE 2,coL:PRINT" ";:sc(2,coL)=pipe:Ln=2:pdir=1:dir=1:GOTO PLot
GetDir:
dir=INT(RND*3):ck(dir)=ck(dir)+1
IF ck(0)>1 AND ck(1)>1 AND ck(2)>1 THEN
IF pno(row,coL)=pipe THEN
LOCATE row,coL:COLOR ,0:PRINT" ";:COLOR ,2
sc(row,coL)=0:pno(row,coL)=0:pipmap(row,coL)=0
END IF
im=im+1:IF im>10 THEN NewMap
row=row-dy(pdir):coL=coL-dx(pdir)
ck(pdir)=2:ck((pdir+1) MOD 2)=0:ck((pdir+2) MOD 2)=0
END IF
IF sc(row+dy(dir),coL+dx(dir))=pipe OR ck(dir)>1 THEN GetDir
IF dir=0 THEN GOSUB Left
IF dir=1 THEN GOSUB Down
```

Danger Zone.



FamilyCare™ Software prepares you for life's little emergencies.

Illness and injury. It's every parent's fear, but you don't have to worry. Thanks to FamilyCare™ software you can now handle medical problems with confidence. Developed by a team of expert pediatricians, FamilyCare is a revolutionary new computer program that will tell you exactly what to do if your child gets sick or injured.

With FamilyCare, you can now get the medical answers you need. Fast. And FamilyCare is as easy to use as talking to your doctor. By asking you simple questions about your child's symptoms, the FamilyCare program will interact with you to help determine the nature and seriousness of the problem. You will then receive professional advice on how to get proper and effective medical help.

Thanks to FamilyCare, you will now know. . .

- when, and when not to see a doctor
- how to treat minor problems at home
- how to spot the early symptoms of serious illness
- what to do in a medical emergency

FamilyCare will teach you the basics of first aid as well as provide you with a health maintenance schedule so that you can help prevent illness and injury throughout your child's development. And FamilyCare is comprehensive in its coverage. You will receive top pediatric advice on everything from diaper rash to serious illness.

The remarkable FamilyCare program is now available for only $180. Imagine, for the cost of a few, and sometimes unnecessary doctor's visits, you can now receive expert medical advice whenever you need it. Any time, day or night. And FamilyCare comes with a 30-day money-back guarantee. If you are not completely satisfied, you may return the program for a full refund.

Your family depends on you, so be prepared. Order FamilyCare™ software today and you will rest assured that you're doing the best for your child.

Family CARE™ SOFTWARE

Order FamilyCare Today. (For Newborns to Age 15)
Telephone orders: Call 1-800-426-8426
In Michigan call collect at (313) 559-4561.
Mail Orders: (Please fill out and send to FamilyCare™ Software, 29451 Greenfield Rd., Suite 216, Southfield, MI 48076-2251)
Method of Payment: ☐ VISA ☐ MC ☐ Check or Money Order

Credit Card # _____

Signature _____ Exp. Date _____
(Make checks payable to FamilyCare—Include $2.00 for postage and handling. Canadian and overseas orders add $5.00.)

Name _____

Address _____

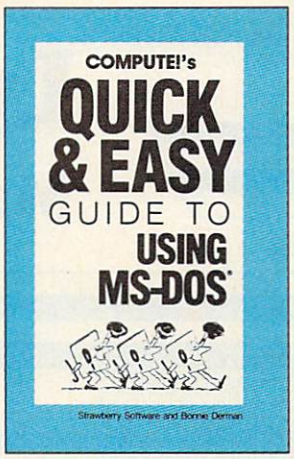
Phone _____

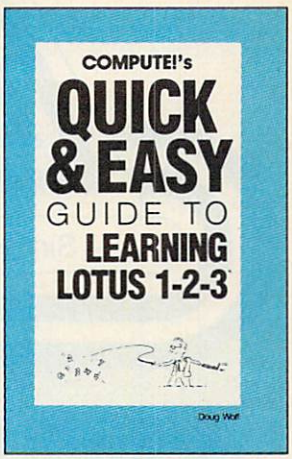
Indicate Computer: ☐ Macintosh® ☐ IBM PC® and compatibles

```
IF dir=2 THEN GOSUB Right
IF Ln>1 THEN ck(0)=0:ck(1)=0:ck(
2)=0:IF Ln>2 THEN im=0
Ln=INT(RND*(Ln+1))
PLot:                    
IF pdir<>dir AND Ln>1 THEN pipma
p(row,coL)=dir+1:pdir=dir
WHILE Ln>1
row=row+dy(dir):coL=coL+dx(dir):
LOCATE row,coL:PRINT" ";
sc(row,coL)=pipe:pno(row,coL)=pn
o(row,coL)+pipe
Ln=Ln-1:WEND
IF row<20 THEN GetDir
IF pipe<npip THEN NewPipe
GOSUB GetIntersect:IF ni<1 OR ni
>30 THEN NewMap
GOSUB MoveBaLLs:COLOR ,5
IF numpL=0 THEN
pts=pts+ct-ms:IF pts<0 THEN pts=
0
LOCATE 23,14:PRINT pts"    ":tot=
tot+ct+ms:ctot=ctot+ct
LOCATE 23,32:PRINT INT(ctot/tot*
100)"    "
npip=INT(pts/10)+3:IF npip>6 THE
N EndGame
ELSE
LOCATE 23,cp(pn)*ofst+2:PRINT na
m$(cp(pn))
ppts(cp(pn))=ppts(cp(pn))+ct:LOC
ATE 24,cp(pn)*ofst+2:PRINT ppts(
cp(pn));
IF pn=0 THEN
FOR i=numpL TO 1 STEP-1:FOR j=0
TO i-1
IF ppts(cp(j))<ppts(cp(j+1)) THE
N SWAP cp(j),cp(j+1)
NEXT j,i
IF ppts(cp(numpL))<ppts(cp(numpL
-1)) THEN
npip=npip+1:COLOR 10,9:LOCATE 21
,1
PRINT SPACE$(40):LOCATE 21,13:PR
INT nam$(cp(numpL))" Eliminated"
LOCATE 23,cp(numpL)*ofst+2:COLOR
0,5:PRINT nam$(cp(numpL))
FOR i=0 TO numpL:ppts(cp(i))=0:N
EXT:numpL=numpL-1:IF numpL=0 THE
N EndGame
END IF
pn=numpL
ELSE
pn=pn-1
END IF
END IF
GOTO NewMap

EndGame:
FOR i=0 TO 5000:NEXT:COLOR 10,9:
LOCATE 21,1:PRINT SPACE$(40)
LOCATE 21,5:PRINT"End of Game.
Play Again (Y/N)?"
k$="":WHILE k$="":k$=UCASE$(INKE
Y$):WEND
IF k$="Y" THEN CLS:GOTO NewGame
IF k$="N" THEN GOSUB CLoseSprite
s
GOTO EndGame

Left:
i=coL:Ln=0
LeftCk:
Ln=Ln+1
IF sc(row+1,i)>0 THEN IF sc(row+
1,i-1)>0 THEN RETURN
IF sc(row-1,i)>0 THEN IF sc(row-
1,i-1)>0 THEN RETURN
i=i-1:IF i<2 THEN RETURN
GOTO LeftCk

Right:
i=coL:Ln=0
RightCk:
Ln=Ln+1
IF sc(row+1,i)>0 THEN IF sc(row+
1,i+1)>0 THEN RETURN
IF sc(row-1,i)>0 THEN IF sc(row-
1,i+1)>0 THEN RETURN
i=i+1:IF i>38 THEN RETURN
GOTO RightCk

Down:
i=row:Ln=0
DownCk:
Ln=Ln+1
IF sc(i,coL+1)>0 THEN IF sc(i+1,
coL+1)>0 THEN RETURN
IF sc(i,coL-1)>0 THEN IF sc(i+1,
coL-1)>0 THEN RETURN
i=i+1:IF i>20 THEN RETURN
GOTO DownCk

GetIntersect:
ni=0
FOR row=3 TO 19:FOR coL=1 TO 39
IF sc(row,coL)>0 AND sc(row-1,co
L)>0 THEN
sdn=ABS(sc(row+1,coL)>0):slt=ABS
(sc(row,coL-1)>0):srt=ABS(sc(row
,coL+1)>0)
IF sdn+slt+srt>1 THEN
xi(ni)=(coL-1)*8:yi(ni)=(row-1)*
8
IF sdn=0 THEN
ti(ni)=0:di(ni)=INT(RND*2)*2
ELSEIF slt=0 THEN
ti(ni)=1:di(ni)=INT(RND*2)+1
ELSEIF srt=0 THEN
ti(ni)=2:di(ni)=INT(RND*2)
ELSE
ti(ni)=3:di(ni)=INT(RND*3)
END IF
pipmap(row,coL)=di(ni)+1:PUT(xi(
ni),yi(ni)),arrow(0,di(ni)),PSET
:ni=ni+1
END IF
END IF
NEXT coL,row
ni=ni-1:RETURN

RotateArrows:
FOR i=0 TO ni
ON ti(i) GOTO rot1,rot2,rot3
di(i)=2-di(i):GOTO 10
rot1:di(i)=3-di(i):GOTO 10
rot2:di(i)=1-di(i):GOTO 10
rot3:di(i)=(di(i)+1) MOD 3
10 pipmap(INT(yi(i)/8)+1,INT(xi(
i)/8)+1)=di(i)+1
PUT(xi(i),yi(i)),arrow(0,di(i)),
PSET
NEXT:RETURN

MoveBaLLs:
ct=0:ms=0:cx=156:cy=162:OBJECT.X
1,156:OBJECT.Y 1,162:OBJECT.ON 1
:i=STRIG(2)
FOR i=0 TO npip-1:CALL MoveSprit
e&(0,sprite&+48*i,ballx(i)*8-5,b
aLLy(i)*8-13)
NEXT:WHILE STRIG(2)=0:cx=cx+STIC
K(2)*8:OBJECT.X 1,cx:FOR i=0 TO
300:NEXT:WEND
bd=0
WHILE bd=0:bd=1
FOR j=0 TO npip-1
IF baLLy(j)<21 THEN
bd=0
IF pipmap(baLLy(j),ballx(j))>0 T
HEN balldir(j)=pipmap(baLLy(j),b
allx(j))-1
ballx(j)=ballx(j)+dx(balldir(j))
:baLLy(j)=baLLy(j)+dy(balldir(j)
)
CALL MoveSprite&(0,sprite&+48*j,
ballx(j)*8-5,baLLy(j)*8-5)
ELSEIF baLLy(j)=21 THEN
IF j=2 OR j=3 THEN
IF ballx(j)-INT(cx/8)=4 THEN GOS
UB Catch ELSE GOSUB Miss:PUT((ba
llx(j)-1)*8,170),fbr,PSET
ELSE
IF ballx(j)-INT(cx/8)=2 THEN GOS
UB Catch ELSE GOSUB Miss:PUT((ba
llx(j)-1)*8,170),fbb,PSET
END IF
baLLy(j)=22
END IF
NEXT
IF STRIG(2) THEN GOSUB RotateArr
ows
cx=cx+STICK(2)*8:OBJECT.X 1,cx
WEND
OBJECT.OFF 1:LINE(8,170)-(304,17
1),5,bf
RETURN

Catch:
SOUND 2000,.5:CALL MoveSprite&(0
,sprite&+48*j,-16,0):ct=ct+1:RET
URN

Miss:
SOUND 20,.5,255,3:ms=ms+1:CALL M
oveSprite&(0,sprite&+48*j,-16,0)
:RETURN

LeftArrow:
DATA 7,7,3,1800,2800,4E00,8200,4
E00
DATA 2800,1800,FE00,EE00,CF00,83
00,CF00,EE00
DATA FE00,0,1000,3100,7D00,3100,
1000,0
DATA 0,0

DownArrow:
DATA 7,7,3,3800,2800,EE00,8200,4
400
DATA 2800,1000,FE00,EE00,EF00,83
00,C700,EE00
DATA FE00,0,1000,1100,7D00,3900,
1000,0
DATA 0,0

RightArrow:
DATA 7,7,3,3000,2800,E400,8200,E
400
DATA 2800,3000,FE00,EE00,E700,83
00,E700,EE00
DATA FE00,0,1000,1900,7D00,1900,
1000,0
DATA 0,0

CupData:
DATA 0,0,0,0,0,0,0,0
DATA 0,0,0,4,0,0,0,32
DATA 0,0,0,11,0,24,0,15
DATA 0,0
DATA 15,128,15,128,50,64,50,64
DATA 127,252,127,252,255,253,255
,253
DATA 55,240,55,240,136,13,104,13

DATA 0,248,64,248,64,114,96,114
DATA 99,228,99,228,15,8,15,8
DATA 0,96,0,96
DATA 15,240,15,240,48,76,48,76
DATA 112,6,112,6,96,2,96,2
DATA 240,12,48,12,255,252,127,25
2
DATA 127,248,127,248,127,240,127
,240
DATA 63,224,63,224,15,0,15,0
DATA 0,0,0,0
DATA 15,240,15,240,48,76,63,188
DATA 112,6,79,250,102,178,223,25
5
DATA 242,172,255,255,255,252,143
,243
DATA 127,248,128,6,127,240,0,14
DATA 63,224,64,28,15,0,16,248
DATA 0,0,7,224
DATA 15,240,15,240,63,188,63,252

DATA 79,250,127,254,217,79,249,7
9
DATA 61,95,253,95,15,243,255,255

```

COMPUTE! Books introduces a new line of easy-to-use computer books.

COMPUTE!'s Quick and Easy Guides are effective, dependable reference guides to using applications on your personal computer. From command summaries and quick-reference charts to tips for power users, you'll have the information you need right at your fingertips. Each book is specially designed to be convenient and easy to understand.





COMPUTE!'s Quick and Easy Guide to Using MS-DOS
Bonnie Derman and Strawberry Software
ISBN 0-87455-105-6 $12.95
Whether you're an experienced MS-DOS user or just starting out, this book is *your* easy-to-use reference to this powerful and flexible operating system for IBM and compatible computers. No longer will you need to wade through hundreds of pages to find just the right DOS command. *COMPUTE!'s Quick and Easy Guide to Using MS-DOS* puts the information right at your fingertips. With two indices and an expanded table of contents, simply look up the task you want to accomplish or the DOS command you need—and you'll find an explanation of exactly what to do.

COMPUTE!'s Quick and Easy Guide to Learning *Lotus 1-2-3*
Doug Wolf
ISBN 0-87455-106-4 $12.95
The perfect guide for the novice user of the popular and powerful *Lotus 1-2-3*, this book assumes no previous computer experience whatsoever. *COMPUTE!'s Quick and Easy Guide to Learning* Lotus 1-2-3 covers everything from building a spreadsheet model, setting up a database, and using *1-2-3* for word processing, to creating graphs and powerful macros. Also included are discussions on how to start *1-2-3* and available add-on software that can be used with *Lotus 1-2-3*.

COMPUTE!'s Quick and Easy Guide to *AppleWorks*
Tom and Ellen Dougher
ISBN 0-87455-109-9 $10.95
AppleWorks, the unchallenged leader in personal productivity software on the Apple II series, is an integrated package combining a word processor, database, and spreadsheet. It's a complete program that lets you do everything from writing letters and reports to keeping extensive files and projecting financial earnings. With *COMPUTE!'s Quick and Easy Guide to* Appleworks, you'll see how to get started with *AppleWorks*, and how to use each of the three major applications. A Command Summary, Keyboard Template, and Quick-Reference Chart provide information in one-stop locations. Numerous ideas offer practical examples on new ways to use *AppleWorks*. And there's even all the latest information on *AppleWorks*' add-ons and enhancements. Clearly defined sections apply to novices and power users alike.

COMPUTE!'s Quick and Easy Guide to *dBase III Plus*
Chuck Doherty
ISBN 0-87455-107-2 $10.95
For the beginning or experienced user of this popular database management program. Ashton-Tate has sold more than a million copies of *dBase*, and *COMPUTE!'s Quick and Easy Guide to* dBase III Plus is *the* guide and reference to help you tap all the power and flexibility that *dBase III Plus* has to offer. This is not a manual for *dBase*; rather, it's a guide that makes getting started with *dBase* easy and finding specific information quick. For all users of *dBase III Plus* on the IBM PC, PC XT, PC AT, or compatible computer.

COMPUTE!'s Quick and Easy Guide to *WordPerfect*
Patricia Farrell
ISBN 0-87455-011-4 $10.95
Write, edit, format, and print with *WordPerfect* your first day, using this simple-to-use guide to the most popular word processor for the IBM and IBM compatibles. Organized so that you can immediately begin to use this impressive software, this book covers all the basics and many of the program's advanced features. Read what you need and then start to work with *WordPerfect*. Learn how to enter, edit, shift, and print text. Use the more advanced techniques when you're ready. See how to prepare a complete report, set text columns, merge information to create form letters, and create special forms.

COMPUTE!'s Quick and Easy Guide to Desktop Publishing
Dan McNeill
ISBN 0-87455-112-9 $10.95
With a computer, a printer, and the right software, you can turn out documents, flyers, brochures, and more—all of the highest quality available outside a professional print shop. This easy-to-use reference guide explains the ins and outs of desktop publishing, shows what you can do with the variety of available software, and even helps you select your system. If you're tired of wading through long and technical manuals, this guide and tutorial will get you started today with desktop publishing.

COMPUTE!'s Quick and Easy Guide to Dow Jones News/Retrieval
Donald B. Trivette
ISBN 0-87455-113-7 $10.95
Learn how to more efficiently use the Dow Jones Information Service, a powerful business and stock information database. Access up-to-date stock quotations, financial information, company background data, and more—all with Dow Jones. Using a personal computer, a modem, the phone, and the right software, you can search through back issues of publications like the *Wall Street Journal* or directly place orders. This simplified guide to the preeminent business information service puts everything within easy reach, and gets you started faster and more easily than any manual.

Look for COMPUTE!'s Quick and Easy Guides at your local book or computer store.
To order any COMPUTE! book, call toll free 1-800-346-6767 (in NY call 212-887-8525)
or write COMPUTE! Books, Customer Service, P.O. Box 5038, F.D.R. Station, New York, NY 10150.
Customer Service hours are 10:00 a.m.–12:30 p.m. and 1:30 p.m.–3:00 p.m. EST Mon. through Fri.

COMPUTE! Publications, Inc.
Part of ABC Consumer Magazines, Inc.
One of the ABC Publishing Companies

COMPUTE! books are available outside the United States from subsidiaries of McGraw-Hill International Book Company.

Lyco Computer

Marketing & Consultants

Since 1981

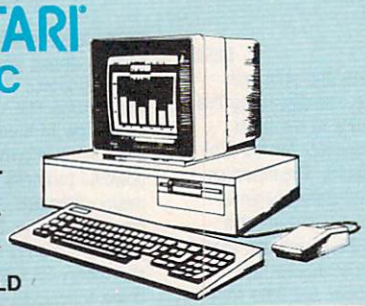
BLUE CHIP PERSONAL COMPUTER

- 100% IBM PC/XT compatible!

$499.00

(Green, amber & color monitors available)

ATARI PC



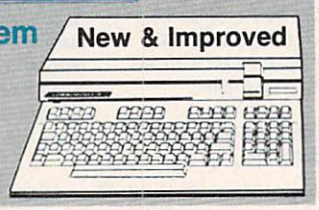
~~$499~~

WE WILL NOT BE UNDERSOLD

COMMODORE 128 D System

New & Improved

$459.95



Commodore 128 Obsolete Discontinued

ATARI 1040 ST Color System

$799.95

PC COMPATIBLE HARDWARE

PC 5¼ Drive	$119
MS DOS + Basic Soft	$59
BCM 12G Gr. Monitor	$89
BCM 12A Am. Monitor	$89
BCM 14C Color RGB Monitor	$269
BCC CG Color Card	$94.99
BCPC PC/XT Comp	$559
Avatex Graphics Card	$169
Zuckerboard	$CALL
Iomega Bernoulli	$CALL
Hercules	$CALL
AST	$CALL

Seagate HARD DRIVES

20 meg	**$275.95**
30 meg	**$315.95**
40 meg	**$489.95**

Drives include controller

COMMODORE HARDWARE

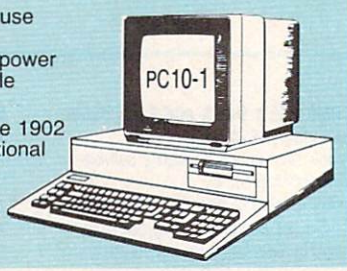
1541 C Disk Drive	$168.95
1571 Disk Drive	$219.95
1581 Disk Drive	$199.95
Indus GT C-64 Drive	$169.95
Blue Chip 64C Drive	$CALL
Blue Chip C 128 Drive	$CALL
1802C Monitor	$194.95
1902A Monitor	$239
2002 Monitor	$249.95
1901 Monitor	$CALL
64 C Computer	$149.95
128D Computer/Drive	$459.95
1670 Modem	$CALL
C-1351 Mouse	$32.95
1700 RAM	$109.95
C-1750 RAM	$169.95
1764 RAM C64	$124.95
Geos	$35.95

ATARI HARDWARE

520 ST FM Mono	$475.95
520 ST FM Color	$599.95
1040 ST Mono	$665.95
1040 ST Color	$799.95
130XE Computer	$125
1050 Drive	$134.95
SX551 Drive	$179.95
SF 314 Disk Drive	$199.95
SF 354 Disk Drive	$119.95
1050 Drive (XE,XL)	$135
Indus GT Atari Drive	$175.95
SHD 204 20 MEG Drive	$569.95
XM301 Modem	$42.95
SX212 Modem	$89.95

COMMODORE PC 10-1

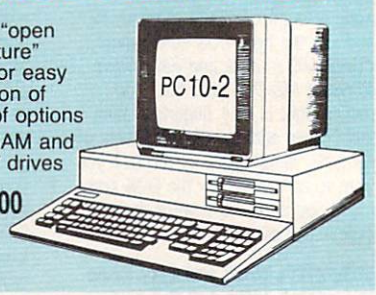
- A ready-to-use package of computing power and versatile graphics!
- Commodore 1902 monitor optional



$489.00

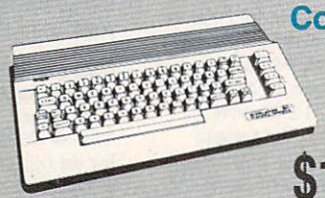
COMMODORE PC 10-2

- Flexible "open architecture" design for easy installation of variety of options
- 640 K RAM and two 5¼" drives



$589.00

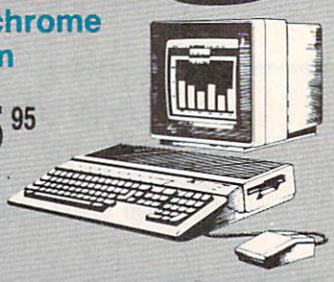
COMMODORE 64C Computer



$149.95

ATARI 520 ST-FM Monochrome System

NEW



$475.95

Internal drive included

1-800-233-8760

Price Guarantee

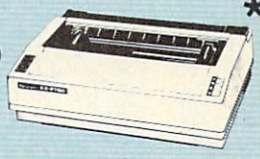
Since 1981, we have led the industry by continuing to offer the lowest national prices while providing quality service. Many companies have come and gone trying to imitate our quality and service. If by some oversight we do not have the lowest prices advertised on the products you desire, then we would appreciate the opportunity to rectify this oversight.

* The Reliable, Affordable Choice

- 120 cps Draft Mode
- 24 cps NLQ
- Word Process
- Friction Feed

(2-year warranty)

Panasonic 1080i



***$139.95**

Monitors

Thompson:

230 Amber TT1/12"	$85
450 Amber TTL/15"	$149
980 White TT1/20"	$695
4120 CGA	$229
4120 RGB/COM	$235
4160/36382	$279
4160 CGA	$289
4460 EGA	$359
4375 UltraScan	$525

Panasonic:

1320	$369
1361	$459

Zenith:

ZVM 1220	$89
ZVM 1230	$89

NEC:

Multisync	$559

THOMSON 4120 Monitor

- 14" RGBI/video composite/analog
- Compatible with IBM and Commodore
- RGB data cable included

$229.95

Modems

Avatex:

1200	$89
1200i	$99
1200hc Modem	$99.95
2400	$219
2400i	$CALL

Hayes:

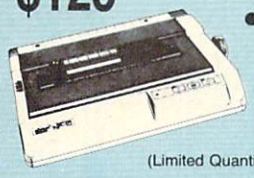
Smartmodem 300	$125
Smartmodem 1200	$369
Smartmodem 2400	$559

* Special Purchase

star MICRONICS, INC

NP 10

***$125.00**



- 100 cps draft
- 25 NLQ
- EZ Front Panel Selection
- Auto Single Sheet Feed
- Adjustable Tractor Feed

EXCELLENT VALUE

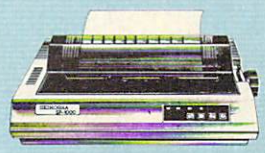
(Limited Quantity) w/purchase of 2 ribbons

* Great Performance, Great Price

SEIKOSHA 180Ai

- 100 cps draft
- 20 cps NLQ
- standard parallel and IBM

***$124.95**



PRINTERS

star MICRONICS, INC

NP-10	*SPECIAL
NX-10	**$144.95**
NX-10C w/interface	$165.95
NL-10	$195.95
NX-15	$279.95
ND-10	$249.95
ND-15	$375.95
NR-15	$409.95
NB-15	$699.95
NB24-10	$409.95
NB24-15	$549.95
SD-10	$229
SD-15	$345
SR-10	$369
SR-15	$429

DIABLO

D25	$499
635	$779
D-80 IF	$1299

(special shipping — call for details)

Printer ribbons available. Priced from $4.99

SEIKOSHA

SP 180Ai	$124.95
SP 180VC	$124.95
SP 1000A	$159
SP 1000I	$159
SP 1000VC	$135.95
SP 1200VC	$144.95
SP 1200Ai	$155.95
SP 1200AS RS232	$155.95
SL 80Ai	$295.95
MP1300Ai	$299.95
MP5300Ai	$375.95
MP5420Ai	**$879.95**
SP Series Ribbon	$7.95
SK3000 Ai	$349
SK3005 Ai	$429

BROTHER

M1109	$195
M1409	$339
M1509	$365
M1709	$475
Twinwriter Dot & Daisy	$849
2024 Letter Quality	$665
HR40	$579
HR60	$CALL

Panasonic

1080i	Special
1091i	**$165.95**
1092i	$295
1592	$379
1595	$419
3131	$259
3151	$399
KXP 4450 Laser	$CALL

OKIDATA

an OKI AMERICA company

Okimate 20	$119
Plug N Play Cart	$CALL
120	$189.95
180	$219.95
182	$229.95
192+	$309.95
193+	$449.95
292 w/interface	$449.95
293 w/interface	$589.95
294 w/interface	$839.95
393	$989.95

NEW

Panasonic 1080i Mod. 2	$154.95
Panasonic 1091i Mod. 2	174.95

EPSON

LX800	$169.95
FX86E	**$289.95**
FX286E	$415.95
EX800	$349.95
EX1000	$469.95
LQ800	$419.95
LQ1000	$589.95
LQ2500	$834.95
GQ3500	$LOW

SILVER REED

EXP 420P	$209
EXP 600P	$539
EXP 800P	$649

CITIZEN

120 D	$164.95
MSP-10	$259.95
MSP-20	$289.95
MSP-15	$324.95
MSP-25	$389.95
MSP-50	$CALL
MSP-55	$CALL
Premiere 35	$464.95
Tribute 224	$624.95

We stock interfacing for Atari, Commodore, Apple and IBM.

SOFTWARE

ATARI ST

ulator ... $24.95
... $24.95
... $24.95
... $24.95
... $26.95
... $26.95
... $26.95
s ... $27.95
... $27.95
aseball .. $35.95
ger ... $19.95
... $24.95
gle ... $24.95
tems:
al ... $48.95
g ... $51.95
ulations:
... $24.95
... $24.95
) ... $24.95
iest ... $24.95
r 2 ... $18.95
... $24.95
rs ... $22.95

ODORE

... $24.95
ses #1 . $13.95
ses #2 . $13.95
... $23.95
... $29.95
... $24.95
ent ... $13.95
... $13.95
... $13.95
r Brd. ... $24.95
... $18.95
ier ... $NEW
... $22.95
ketball . $22.95
... $9.50
... $22.95
... $19.95
... $22.95
... $18.95
... $14.95
sses ... $24.95
... $24.95
... $24.95
ıator ... $19.95
works:
... $35.95
... $23.95
... $19.95
... $24.95
... $29.95
op ... $29.95
128K .. $27.95
iego ... $22.95
II, III ... $15.95
... $12.95
... ... $25.95
npan. ... $22.95
er ... $12.95
ıdar ... $17.95
... $22.95

Fastload ... $22.95
Football ... $13.95
Movie Monster ... $13.95
Multiplan ... $24.95
Sub Battle ... $24.95
Winter Games ... $24.95
Super Cycle ... $13.95
Graphics Scrapbook ... $14.95
California Games ... $24.95
Str. Sports Basketball . $24.95
Wld.'s Great Baseball . $13.95
Summer Games II ... $22.95
Vorpol Utility Kit ... $13.95
World Games ... $22.95

Firebird:
Colossus Chess IV ... $19.95
Elite ... $19.95
Golden Path ... $22.95
Guild of Thieves ... $24.95
Pawn ... $22.95
Talking Teacher ... $22.95
Tracker ... $22.95
Starglider ... $19.95
Sentry ... $22.95

Hi Tech Expressions:
Heart Ware ... $6.50
Holiday Paper ... $8.95
Card Ware ... $6.50
Party Ware ... $9.50

Learning Company:
Reader Rabbit ... $24.95
Magic Spells ... $22.95
Gertrude's Secrets ... $19.95
Rocky's Boots ... $22.95

Microleague:
Microleag. Baseball ... $24.95
General Manager ... $18.95
Stat Disk ... $15.95
'86 Team Disk ... $13.95
Microleag. Wrestling ... $New

Microprose:
F-15 Strike Eagle ... $22.95
Gunship ... $22.95
Kennedy Approach ... $15.95
Silent Service ... $22.95
Solo Flight ... $18.95
Top Gunner ... $15.95

Springboard:
Newsroom ... $31.95
Certificate Maker ... $31.95
Clip Art Vol. #1 ... $19.95
Clip Art Vol. #2 ... $25.95
Clip Art Vol. #3 ... $19.95
Graphics Expander ... $32.95

Strategic Simulations:
Gemstone Healer ... $18.95
Gettysburg ... $35.95
Kampfgruppe ... $35.95
Phantasie II ... $24.95
Phantasie III ... $24.95
Ring of Zelfin ... $24.95
Road War 2000 ... $24.95
Shard of Spring ... $24.95
Wizards Crown ... $24.95
War in the S. Pacific ... $35.95
Wargame Constr. ... $18.95
Battlecruiser ... $35.95
Gemstone Warrior ... $14.95
Battle of Antetiem ... $32.95
Colonial Conquest ... $24.95
Computer Ambush ... $35.95
B-24 ... $22.95

Sublogic:
Flight Simulator II ... $31.95
Jet Simulator ... $25.95
Night Mission Pinball ... $21.95
Scenery Disk ... $15.95

Timeworks:
Partner C64 ... $29.95
Partner 128 ... $34.95

COMMODORE

Swift Calc 128 ... $39.95
Wordwriter 128 ... $39.95

Unison World:
Art Gallery 1 or 2 ... $15.95
Print Master ... $22.95

Apple

Access:
Triple Pack ... $13.95

Activision:
Zork Trilogy ... $29.95
Game Maker ... $29.95
Leather Goddesses ... $24.95
Term Paper ... $29.95
Champ. Baseball ... $24.95
Champ. Basketball ... $24.95
Championship Golf ... $New
GFL Football ... $24.95
Top Fuel Eliminator ... $19.95

Broderbund:
Airheart ... $22.95
Print Shop ... $26.95
Print Shop Comp. ... $22.95
Carmen S. Diego (USA)..$26.95
On Balance ... $62.95
Bank St. Writer + ... $44.95

Epyx:
Create A Calendar ... $17.95
Destroyer ... $24.95
Karate Champ ... $19.95
Movie Monster ... $14.95
St. Sports Basketball ... $24.95
Sub Battle Simulator ... $24.95
Winter Games ... $22.95
World Games ... $24.95
Wrestling ... $24.95

Firebird:
Colossus IV Chess ... $24.95
Elite ... $22.95
The Pawn ... $26.95
Starglider ... $26.95

Learning Company:
Think Quick ... $29.95
Reader Rabbit ... $24.95
Math Rabbit ... $24.95
Writer Rabbit ... $24.95
Magic Spells ... $22.95

Microleague:
Microleag. Baseball ... $23.95
General Manager ... $18.95
Stat Disk ... $13.95
'86 Team Disk ... $11.95

Microprose:
Crusade in Europe ... $24.95
Decision in Desert ... $24.95
F-15 Strike Eagle ... $20.95
Silent Service ... $20.95

Strategic Simulations:
Battlecruiser ... $35.95
Colonial Conquest ... $24.95
Gettysburg ... $34.95
Phantasie III ... $22.95
Realms of Darkness ... $22.95
Shard of Spring ... $22.95
Wizards Crown ... $22.95
Eternal Dagger ... $24.95
Shiloh ... $NEW

Sublogic:
Flight Simulator II ... $29.95
Jet Simulator ... $25.95
Night Mission Pinball ... $22.95
Scenery #1-#6 ... $69.95

Unison World:
Art Gallery 2 ... $19.95
Print Master ... $25.95

AMIGA

Access:
Leader Board ... $24.95
10th Frame ... $24.95
Tournament #1 ... $14.95

Activision:
Champ. Basketball ... $27.95
Championship Golf ... $24.95
GFL Football ... $27.95
Hacker ... $16.95
Hacker 2 ... $24.95
Music Studio ... $29.95
Tass Times ... $24.95

Epyx:
Apshai Trilogy ... $13.95
Rogue ... $22.95
Winter Games ... $22.95
World Games ... $22.95

Firebird:
Guild of Thieves ... $26.95
Pawn ... $26.95
Starglider ... $26.95

Microprose:
Silent Service ... $24.95

Sublogic:
Flight Simulator II ... $32.95
Jet Simulator ... $32.95
Scenery Disk ... $17.95

Unison World:
Print Master ... $24.95
Art Gallery 1 or 2 ... $18.95
Fonts & Borders ... $22.95

IBM

Access:
Wld. Cl. Leader Board . $CALL

Activision:
Pebble Beach Golf ... $24.95
Champ. Baseball ... $24.95
Champ. Basketball ... $24.95
Zork Trilogy ... $39.95
Leather Goddesses ... $24.95
Moonmist ... $24.95

Broderbund:
Ancient Art of War ... $28.95
Print Shop ... $32.95
Print Shop Compan. ... $31.95
Graphic Lib. I or II ... $21.95
Karateka ... $21.95
Toy Shop ... $29.95
Bank St. Writer + ... $54.95

Epyx:
Apshai Trilogy ... $14.95
Create A Calendar ... $17.95
Destroyer ... $24.95
Movie Monster ... $14.95
Rogue ... $19.95
St. Sports Basketball ... $24.95
Sub Battle Simulator ... $23.95
Winter Games ... $24.95
World Games ... $23.95

Firebird:
Starglider ... $26.95
Guild of Thieves ... $26.95

Learning Company:
Think Quick ... $29.95
Reader Rabbit ... $24.95
Math Rabbit ... $24.95
Writer Rabbit ... $24.95
Magic Spells ... $22.95

Microleague:
Microleag. Baseball ... $23.95
General Manager ... $18.95
Stat Disk ... $15.95
'86 Team Disk ... $11.95

Microprose:
Conflict in Vietnam ... $24.95
Crusade in Europe ... $24.95
Decision in Desert ... $24.95
F-15 Strike Eagle ... $20.95
Silent Service ... $20.95
Gunship ... $28.95

Strategic Simulations:
Road War 2000 ... $24.95
Kampfgruppe ... $35.95
Shard of Spring ... $24.95
Battle of Antetiem ... $35.95
Computer Baseball ... $14.95
Gettysburg ... $35.95
Wizards Crown ... $24.95
Chickamauga ... $35.95

Sublogic:
Jet Simulator ... $31.95
Scenery Japan ... $17.95
Scenery San Fran. ... $17.95
Scenery #1-#6 ... $69.95
Flight Simulator ... $36.95

Timeworks:
Swiftcalc ... $39.95
Wordwriter ... $39.95

Unison World:
Art Gallery 2 ... $18.95
News Master ... $55.95
Print Master ... $36.95
Fonts & Borders ... $22.95

Diskettes

5-1/4

Maxell:
SSDD ... $8.99
DSDD ... $9.99

Bonus:
SSDD ... $6.95
DSDD ... $7.95

SKC:
DSDD ... $9.95
DSHD ... $16.95
Generic DSDD ... $6.99

Verbatim:
SSDD ... $9.95
DSDD ... $12.95

3.5

Maxell:
SSDD ... $12.50
DSDD ... $17.99

Verbatim:
SSDD ... $16.95
DSDD ... $24.95

SKC:
SSDD ... $13.95
DSDD ... $16.95
Generic SSDD ... $13.95
Generic DSDD ... $15.95

Joysticks

Tac 3 ... $9.95
Tac 2 ... $10.95
Tac 5 ... $12.95
Tac 1 + IBM/AP ... $26.95
Economy ... $5.95
Slik Stick ... $6.95
Black Max ... $10.95
Boss ... $11.99
3-Way ... $19.99
Bathandle ... $16.75
500XJ ... $14.75
500XJ-Apple/PC ... $28.95
Winner 909 ... $24.95
Wico IBM/AP ... $29.95
PC Joystick Board ... $34.95

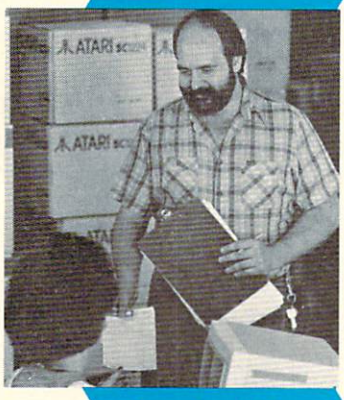
Lyco Means Total Service.

ATARI

Mark "Mac" Bowser, Sales Manager

I would personally like to thank all of our past customers for helping to make Lyco Computer one of the largest mail order companies and a leader in the industry. Also, I would like to extend my personal invitation to all computer enthusiasts who have not experienced the services that we provide. Please call our trained sales staff at our toll free number to inquire about our diverse product line and weekly specials.

Call Lyco

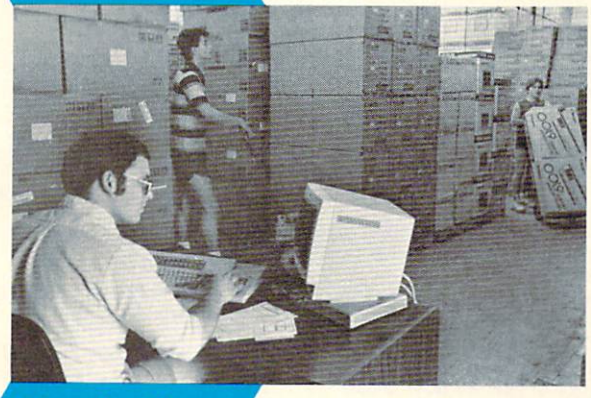


First and foremost our philosophy is to keep abreast of the changing market so that we can provide you with not only factory-fresh merchandise but also the newest models offered by the manufacturers at the absolute best possible prices. And we offer the widest selection of computer hardware, software and accessories.

Feel free to call Lyco if you want to know more about a particular item. I can't stress enough that our toll-free number is not just for orders. Many companies have a toll-free number for ordering, but if you just want to ask a question about a product, you have to make a toll call. Not at Lyco. Our trained sales staff is knowledgeable about all the products we stock and is happy to answer any questions you may have. We will do our best to make sure that the product you select will fit your application. We also have Saturday hours — one more reason to call us for all your computer needs.

Order Now

Once you've placed your order with Lyco, we don't forget about you. Our friendly, professional customer service representatives will find answers to your questions about the status of an order, warranties, product availability, or prices.



Lyco Computer stocks a multimillion dollar inventory of factory-fresh merchandise. Chances are we have exactly what you want right in our warehouse. And that means you'll get it fast. In fact, orders are normally shipped within 24 hours. Shipping on prepaid cash orders is free in the U.S., and there is no deposit required on C.O.D. orders. Air freight or UPS Blue/Red Label shipping is available, too. And all products carry the full manufacturers' warranties.

Here's How

I can't see why anyone would shop anywhere else. Selection from our huge in-stock inventory, best price, service that can't be beat — we've got it all here at Lyco Computer.

TO ORDER, CALL TOLL-FREE: 1-800-233-8760
In PA: 1-717-494-1030

Hours: 9AM to 8PM, Mon. - Thurs.
9AM to 6PM, Friday — 10AM to 6PM, Saturday

For Customer Service, call 1-717-494-1670,
9AM to 5PM, Mon. - Fri.
Or write: Lyco Computer, Inc.
P.O. Box 5088, Jersey Shore, PA 17740

MasterCard VISA

Risk-Free Policy: • full manufacturers' warranties • no sales tax outside PA • prices show 4% cash discount; add 4% for credit cards • APO, FPO, international: add $5 plus 3% for priority • 4-week clearance on personal checks • we check for credit card theft • compatability not guaranteed • return authorization required • price/availability subject to change

Attention Educational Institutions:
If you are not currently using our educational service program, please call our representatives for details.

Access:

Leader Board	$24.95
Tournament #1	$14.95
Triple Pack	$14.95

Activision:

Hacker	$14.95
Hitch Hikers	$24.95
Leather Goddesses	$24.95
Moonmist	$24.95
Music Studio	$24.95
Stationfall	$22.95
Lurking Horror	$22.95

Broderbund:

Print Shop	$25.95
Print Shop Compan.	$22.95
Graphic Lib. I, II, III	$15.95
Karateka	$12.95
Bank St. Writer	$29.95

Firebird:

The Pawn	$27.95

Microleague:

Microleag. Baseball	$24.95
General Manager	$24.95
Stat Disk	$17.95
'86 Team Disk	$14.95
Mach II	$17.95

Microprose:

Conflict in Vietnam	$24.95
F-15 Strike Eagle	$22.95
Kennedy Approach	$18.95
Silent Service	$22.95
Top Gunner	$18.95

Optimized Systems:

Action	$46.95
Action Tool Kit	$18.95
Basic XE	$46.95
Basic XL	$36.95
Basic XL Tool Kit	$18.95
Mac 65	$46.95
Mac 65 Tool Kit	$18.95

Strategic Simulations:

Battle of Antetiem	$32.95
Battlecruiser	$35.95
Nam	$24.95
Phantasie	$24.95
Wargame Construc.	$18.95
Warship	$35.95
Wizards Crown	$24.95

Sublogic:

Flight Simulator II	$31.95
Night Mission Pinball	$21.95
Scenery #1-#6 ea.	$14.95
Scenery #7	$17.95

ATARI ST

Access:

Leader Board	$24.95
Tournament #1	$13.95
10th Frame	$24.95

Activision:

Champion. Baseball	$24.95
Champion. Basketball	$24.95
Championship Golf	$New
GFL Football	$24.95
Hacker 2	$24.95
Leather Goddesses	$24.95
Moonmist	$24.95
Music Studio	$29.95
Paint Works	$24.95
Shanghai	$24.95
Bureaucracy	$24.95
Stationfall	$24.95
Lurking Horror	$24.95
Top Fuel	$19.95

Broderbund:

Karateka	$CALL

Epyx:
Sub Battle Sim
World Games
Wrestling
Winter Games

Firebird:
Pawn
Starglider
Golden Path
Guild of Thiev
Tracker

Microleague:
Microleague B
General Mana

Microprose:
Silent Service
F-15 Strike Ea

Optimized Sy
Personal Pasc
Personal Prol

Strategic Sim
Phantasie
Phantasie II
Road War 200
Colonial Conq

Unison World
Art Gallery 1
Print Master
Fonts & Borde

COMA

Access:
Leader Board
Famous Cour
Famous Cour
Mach 5
Mach – 128
10th Frame
Exec. Tournan
Tournament #
Triple Pack
Wld. Cl. Leade

Action Soft:
Up Periscope
Thunder Chop

Activision:
Aliens
Champion. Bas
Hacker
Hacker 2
Labyrinth
Music Studio
Tass Times
Titanic
Leather Godde
Stationfall
Lurking Horror
Top Fuel Elimi

Berkeley Soft
Geos
Deskpak I
Fontpak I
Geodex
Geofile
Writers Worksh

Broderbund:
Bank St. Write
Carmen San D
Graphic Lib. I,
Karateka
Print Shop
Print Shop Cor
Print Shop Pap

Epyx:
Create A Caler
Destroyer

```
DATA 128,6,255,254,0,14,127,254
DATA 64,28,127,252,16,248,31,248
DATA 7,224,7,224
InitSprites:
sprite&=ALLocRaster&(16,144):IF
sprite&=0 THEN CLoseSprites
CALL BLtCLear&(sprite&,288,0)
FOR i=2 TO 7:offset=48*(i-2)
IF GetSprite&(sprite&+offset,i)<
>i THEN CLoseSprites1
POKEW sprite&+offset+4,7:POKEW s
prite&+offset+6,-17
RESTORE BaLLData:FOR j=16 TO 42
STEP 2:READ k:POKE sprite&+offse
t+j,k:NEXT
CALL ChangeSprite&(0,sprite&+off
set,sprite&+offset+12)
NEXT:RETURN
BaLLData:
DATA 0,60,0,126,3,255,1
DATA 255,15,255,6,126,28,60
CLoseSprites:
FOR i=2 TO 7:CALL FreeSprite&(i)
:NEXT
CLoseSprites1:
CALL FreeRaster&(sprite&,16,144)
WINDOW CLOSE 3:SCREEN CLOSE 1:ON
ERROR GOTO 0:END
```

Program 5: IBM PC/PCjr Marbles

Version by Tim Midkiff, Editorial Programmer

```
LB 10 'COPYRIGHT 1987 COMPUTE! P
      UBLICATIONS, INC.     ALL
      RIGHTS RESERVED.
BN 20 KEY OFF:DEF SEG=0:DEFINT A
      -Z:POKE 1047,PEEK(1047) OR
       64:RANDOMIZE TIMER
CE 30 DIM SC(21,40),PNO(21,40),P
      IPMAP(21,40),XI(30),YI(30)
      ,DI(30),TI(30)
DJ 40 WIDTH 40:SCREEN 0,1:RESTOR
      E 40:FOR I=0 TO 2:READ A:A
      RROW$(I)=CHR$(A):NEXT:DATA
       27,25,26
BN 50 RESTORE 50:FOR I=0 TO 2:RE
      AD DX(I),DY(I):NEXT:DATA -
      1,0,0,1,1,0
HN 60 RESTORE 60:FOR I=0 TO 2:BC
      (I*2)=14:BC(I*2+1)=12:NEXT
DI 70 U$=CHR$(72):L$=CHR$(74):R$
      =CHR$(78):E$=" ":C$=CHR$(1
      78):B$=CHR$(153)
MD 80 COLOR 15,2,0:CLS:PRINT TAB
      (12) "Copyright 1987"
CM 90 PRINT TAB(7) "COMPUTE! Pub
      lications, Inc."
QK 100 PRINT TAB(10) "All Rights
       Reserved."
AC 110 LOCATE 10,5:PRINT"Enter N
      umber of Players (1-5) ";
DN 120 K$="":WHILE K$="":K$=INKE
      Y$:WEND:IF K$<"1" OR K$>"
      5" THEN 120
KK 130 PRINT K$:FOR X=1 TO 500:N
      EXT:CLS:NUMPL=VAL(K$)-1:P
      TS=0:PN=NUMPL:LN$=SPACE$(
      40)
DK 140 IF NUMPL=0 THEN LOCATE 23
      ,6:PRINT"Marbles:":LOCATE
       23,25:PRINT"Rating:":NPI
      P=3:GOTO 210
DJ 150 LOCATE 10,1:PRINT"Enter t
      he initials of player"
NG 160 FOR I=0 TO NUMPL:LOCATE 1
      0,29:PRINT LEFT$(STR$(I+1
      ),2);:INPUT NAM$(I)
DH 170 NAM$(I)=LEFT$(NAM$(I)+SPA
      CE$(3),3):LOCATE 10,29:PR
      INT SPACE$(11):NEXT
IJ 180 CLS:OFST=INT(34/NUMPL):NP
      IP=7-NUMPL:COLOR 7
OB 190 FOR I=0 TO NUMPL:CP(I)=I:
      LOCATE 23,I*OFST+2:PRINT
      NAM$(I):NEXT
CC 200 FOR I=0 TO NUMPL:J=INT(RN
      D*(NUMPL+1)):SWAP CP(I),C
      P(J):NEXT
IF 210 LOCATE ,,0:FOR I=0 TO 2:C
      K(I)=0:NEXT:FOR I=2 TO 20
      :FOR J=2 TO 38:SC(I,J)=0:
      PNO(I,J)=0:PIPMAP(I,J)=0:
      NEXT J,I
OI 220 IF NUMPL>0 THEN LOCATE 23
      ,CP(PN)*OFST+2:COLOR 15,2
      :PRINT NAM$(CP(PN)):LOCAT
      E 21,1:PRINT LN$:IF PN=NU
      MPL THEN LOCATE 24,1:PRIN
      T LEFT$(LN$,38);
QK 230 COLOR ,1:PIPE=0:LOCATE 2,
      1:FOR I=2 TO 20:PRINT LN$
      ;:NEXT
QO 240 ROW=2:PIPE=PIPE+1:COLOR ,
      5
EG 250 COL=INT(RND*37)+2:IF (SC(
      2,COL-1) OR SC(2,COL) OR
      SC(2,COL+1))>0 THEN 250
OA 260 GOSUB 850:IF LN<3 THEN 25
      0
GC 270 BALLX(PIPE-1)=COL:BALLY(P
      IPE-1)=ROW:PIPMAP(2,COL)=
      2
IO 280 LOCATE 2,COL:PRINT E$;:SC
      (2,COL)=PIPE:LN=2:PDIR=1:
      DIR=1:GOTO 410
KE 290 DIR=INT(RND*3):CK(DIR)=CK
      (DIR)+1
PE 300 IF NOT(CK(0)>1 AND CK(1)>
      1 AND CK(2)>1) THEN 350
GC 310 IF PNO(ROW,COL)=PIPE THEN
       LOCATE ROW,COL:COLOR ,1:
      PRINT E$;:COLOR ,5:SC(ROW
      ,COL)=0:PNO(ROW,COL)=0:PI
      PMAP(ROW,COL)=0:ROW=ROW-D
      Y(PDIR)
KI 320 IM=IM+1:IF IM>10 THEN 210
HH 330 ROW=ROW-DY(PDIR):COL=COL-
      DX(PDIR)
PH 340 CK(PDIR)=2:CK((PDIR+1) MO
      D 2)=0:CK((PDIR+2) MOD 2)
      =0
FE 350 IF SC(ROW+DY(DIR),COL+DX(
      DIR))=PIPE OR CK(DIR)>1 T
      HEN 290
LL 360 IF DIR=0 THEN GOSUB 730
AG 370 IF DIR=1 THEN GOSUB 850
GD 380 IF DIR=2 THEN GOSUB 790
OM 390 IF LN>1 THEN CK(0)=0:CK(1
      )=0:CK(2)=0:IM=0
JB 400 LN=INT(RND*(LN+1))
GB 410 IF PDIR<>DIR AND LN>1 THE
      N PIPMAP(ROW,COL)=DIR+1:P
      DIR=DIR
NH 420 WHILE LN>1
GH 430 ROW=ROW+DY(DIR):COL=COL+D
      X(DIR):LOCATE ROW,COL:PRI
      NT E$;
GJ 440 SC(ROW,COL)=PIPE:PNO(ROW,
      COL)=PNO(ROW,COL)+PIPE
GP 450 LN=LN-1:WEND
BM 460 IF ROW<20 THEN 290
DA 470 IF PIPE<NPIP THEN 240
HB 480 GOSUB 910:IF NI<1 OR NI>3
      0 THEN 210
IA 490 GOSUB 1130:COLOR 7,2
JB 500 IF NUMPL<>0 THEN 550
CK 510 PTS=PTS+CT-MS:IF PTS<0 TH
      EN PTS=0
QO 520 LOCATE 23,14:PRINT PTS"
      ";:TOT=TOT+CT+MS:CTOT=CTO
      T+CT
GC 530 LOCATE 23,32:PRINT INT(CT
      OT/TOT*100)"  ";
PB 540 NPIP=INT(PTS/10)+3:IF NPI
      P>6 THEN 680 ELSE 670
PJ 550 LOCATE 23,CP(PN)*OFST+2:P
      RINT NAM$(CP(PN))
DF 560 PPTS(CP(PN))=PPTS(CP(PN))
      +CT:LOCATE 24,CP(PN)*OFST
      +2:PRINT PPTS(CP(PN));
CA 570 IF PN<>0 THEN PN=PN-1:GOT
      O 670
JJ 580 FOR I=NUMPL TO 1 STEP -1:
      FOR J=0 TO I-1
PK 590 IF PPTS(CP(J))<PPTS(CP(J+
      1)) THEN SWAP CP(J),CP(J+
      1)
CG 600 NEXT J,I
HD 610 IF NOT(PPTS(CP(NUMPL))<PP
      TS(CP(NUMPL-1))) THEN 660
MG 620 NPIP=NPIP+1:COLOR 14,4:LO
      CATE 21,1
AB 630 PRINT LN$:LOCATE 21,13:PR
      INT NAM$(CP(NUMPL))" Elim
      inated"
GN 640 LOCATE 23,CP(NUMPL)*OFST+
      2:COLOR 8,2:PRINT NAM$(CP
      (NUMPL));
JH 650 FOR I=0 TO NUMPL:PPTS(CP(
      I))=0:NEXT:NUMPL=NUMPL-1:
      IF NUMPL=0 THEN 680
DA 660 PN=NUMPL
CE 670 GOTO 210
DN 680 FOR I=0 TO 5000:NEXT:COLO
      R 14,4:LOCATE 21,1:PRINT
      LN$
DF 690 LOCATE 21,5:PRINT"End of
      Game.  Play again (Y/N)?"
AE 700 K$="":WHILE K$="":K$=INKE
      Y$:WEND
OB 710 IF K$="Y" THEN 80
IB 720 IF K$="N" THEN CLS:END EL
      SE 700
HK 730 I=COL:LN=0
JB 740 LN=LN+1
MK 750 IF SC(ROW+1,I)>0 THEN IF
      SC(ROW+1,I-1)>0 THEN RETU
      RN
CA 760 IF SC(ROW-1,I)>0 THEN IF
      SC(ROW-1,I-1)>0 THEN RETU
      RN
GF 770 I=I-1:IF I<2 THEN RETURN
IK 780 GOTO 740
HG 790 I=COL:LN=0
JK 800 LN=LN+1
GD 810 IF SC(ROW+1,I)>0 THEN IF
      SC(ROW+1,I+1)>0 THEN RETU
      RN
NJ 820 IF SC(ROW-1,I)>0 THEN IF
      SC(ROW-1,I+1)>0 THEN RETU
      RN
ID 830 I=I+1:IF I>38 THEN RETURN
FB 840 GOTO 800
EL 850 I=ROW:LN=0
JG 860 LN=LN+1
JL 870 IF SC(I,COL+1)>0 THEN IF
      SC(I+1,COL+1)>0 THEN RETU
      RN
QN 880 IF SC(I,COL-1)>0 THEN IF
      SC(I+1,COL-1)>0 THEN RETU
      RN
PH 890 I=I+1:IF I>20 THEN RETURN
JM 900 GOTO 860
GP 910 NI=0:COLOR 11
GH 920 FOR ROW=3 TO 19:FOR COL=2
       TO 38
EI 930 IF NOT(SC(ROW,COL)>0 AND
      SC(ROW-1,COL)>0) THEN 102
      0
LH 940 SDN=ABS(SC(ROW+1,COL)>0):
      SLT=ABS(SC(ROW,COL-1)>0):
      SRT=ABS(SC(ROW,COL+1)>0)
EP 950 IF NOT(SDN+SLT+SRT>1) THE
      N 1020
```

```
MB 960 XI(NI)=COL:YI(NI)=ROW
LA 970 IF SDN=0 THEN TI(NI)=0:DI
       (NI)=INT(RND*2)*2:GOTO 10
       10
FP 980 IF SLT=0 THEN TI(NI)=1:DI
       (NI)=INT(RND*2)+1:GOTO 10
       10
KK 990 IF SRT=0 THEN TI(NI)=2:DI
       (NI)=INT(RND*2):GOTO 1010
MD 1000 TI(NI)=3:DI(NI)=INT(RND*
        3)
CI 1010 PIPMAP(ROW,COL)=DI(NI)+1
        :PNO(ROW,COL)=91:LOCATE
        ROW,COL:PRINT ARROW$(DI(
        NI)):NI=NI+1
JA 1020 NEXT COL,ROW
HM 1030 NI=NI-1:J=0:FOR I=2 TO 3
        8:IF SC(20,I)>0 THEN PEX
        IT(J)=I:J=J+1
IP 1040 NEXT:PEXT=J:RETURN
FJ 1050 COLOR 11:FOR I=0 TO NI
MK 1060 ON TI(I) GOTO 1080,1090,
        1100
OG 1070 DI(I)=2-DI(I):GOTO 1110
PF 1080 DI(I)=3-DI(I):GOTO 1110
OA 1090 DI(I)=1-DI(I):GOTO 1110
QJ 1100 DI(I)=(DI(I)+1) MOD 3
GJ 1110 PIPMAP(YI(I),XI(I))=DI(I
        )+1:LOCATE YI(I),XI(I):P
        RINT ARROW$(DI(I))
EK 1120 NEXT:COLOR 14:RETURN
LA 1130 FOR I=0 TO NPIP-1:LOCATE
         BALLY(I),BALLX(I):COLOR
         BC(I):PRINT B$;:NEXT
CN 1140 CT=0:MS=0:CC=0:CX=PEXIT(
        0):CY=21:WHILE INKEY$<>"
        ":WEND:LOCATE CY,CX:COLO
        R BC(0),2:PRINT C$:PEX=0
BN 1150 BD=0:COLOR ,5
HE 1160 WHILE BD=0:BD=1
BB 1170 FOR J=0 TO NPIP-1
BN 1180 IF NOT(BALLY(J)<21) THEN
         1230
HD 1190 IF PIPMAP(BALLY(J),BALLX
        (J))>0 THEN BALLDIR(J)=P
        IPMAP(BALLY(J),BALLX(J))
        -1
OJ 1200 IF PNO(BALLY(J),BALLX(J)
        )>90 THEN LOCATE BALLY(J
        ),BALLX(J):COLOR 11:PRIN
        T ARROW$(BALLDIR(J)) ELS
        E LOCATE BALLY(J),BALLX(
        J):PRINT E$;
JL 1210 BALLX(J)=BALLX(J)+DX(BAL
        LDIR(J)):BALLY(J)=BALLY(
        J)+DY(BALLDIR(J))
HP 1220 LOCATE BALLY(J),BALLX(J)
        :COLOR BC(J):PRINT B$;:B
        D=0:GOTO 1260
II 1230 IF NOT(BALLY(J)=21) THEN
         1260
DH 1240 LOCATE BALLY(J),BALLX(J)
        :BALLY(J)=22
FN 1250 IF BALLX(J)-CX=0 AND BC(
        J)=BC(CC) THEN COLOR BC(
        CC),2:PRINT C$:COLOR ,5:
        SOUND 1000,2:CT=CT+1 ELS
        E COLOR ,2:PRINT"_";:COL
        OR ,5:SOUND 37,4:MS=MS+1
AC 1260 NEXT
NN 1270 K$=RIGHT$(INKEY$,1):IF K
        $="" THEN 1300
FD 1280 IF K$=E$ THEN GOSUB 1050
         ELSE IF K$=U$ THEN CC=1
        -CC:K$=CHR$(76)
ND 1290 IF K$>L$ AND K$<R$ THEN
        COLOR BC(CC),2:LOCATE CY
        ,CX:PRINT E$;:PEX=(PEX+P
        EXT+ASC(K$)-76) MOD PEXT
        :CX=PEXIT(PEX):LOCATE CY
        ,CX:PRINT C$;:COLOR ,5
EM 1300 WEND:LOCATE 21,1:COLOR ,
        2:PRINT LN$:RETURN
```

©

Computers and Society

David D. Thornburg, Associate Editor

Son Of A Nation Of Thieves

I am writing what I hope will be my last column on this topic for a while. Some time back, I wrote a piece highly criticizing software theft by computer users, who were depriving developers of income for their hard work. The responses to this column were so defensive of software theft that I published another article to respond to some misconceptions I felt needed to be addressed. It seemed that many people took the position that anything one could get away with was fair game, and if some software developers went out of business, they should have chosen better fields in which to make money to start with.

There were a few letters that went on to justify software theft, by pointing out that manufacturers weren't very supportive of their customers anyway, but I tended to treat this response as a weak justification for unjustifiable behavior.

As soon as this second article hit the streets, my mailbox once again filled with responses from readers. This time I heard from those who defended my position in support of copyright law enforcement. However, the subject of poor after-sale support reared its ugly head, and since these people were not using this phenomenon to justify software copying, I thought it was worth a little investigation. After making a few calls and exploring some problems with software I had purchased, I found that the readers' horror stories were all too true.

I think it's time for a heart-to-heart talk with software developers and manufacturers. If my position on copying irritated a lot of software users, this article may irritate a few developers as well.

We shall see.

The Nature Of Software

It seems to me that all companies in the software-development business should acknowledge that a computer program is not a static document, sold into a marketplace with no need for after-sale support. Even shoe salesmen acknowledge that a customer may have to wear a pair of shoes for a few hours to make sure they fit properly.

There are two important reasons for after-sale software support. First, the customer may have purchased a program to perform a task that's not quite what the program was designed to do. This happens a lot. Software names and advertisements are not always good indicators of the program's ability to function, and software reviews are not always helpful. Many times the program can be used to do what the customer wanted, but the manual isn't clear enough. In this case, it is essential that a software manufacturer provide sufficient support to help customers. New applications for the product should be flagged so they can be addressed in later releases of the product, thus amplifying the software's value.

The second reason for providing good customer support is that all software is buggy to one extent or another. For example, some programs I have used since 1984 suddenly stopped working because Apple "fixed" its operating system. As a software developer, this is an annoyance, but I must upgrade my software if I'm to have any more sales, since all the new computers (and most of the old ones) will be using the new operating system within a few months. This is expensive, of course, but it isn't the customer's fault.

Aside from this type of bug, there are those bugs, indigenous to programs, that can lie fallow for a long time before anyone notices them. These too must be fixed, and the customer has a right to benefit from these fixes, since most bugs are found by the customers anyway. My own company, Innovision, is in its second year of production of *Calliope*, and we are still finding gremlins in the code. I don't know any other developer of a sophisticated program who has done any better in releasing bug-free code.

The point is, as the designer and manufacturer of these products, bugs are our problems, not the customers'. Many readers complained that software companies were rude to them when they told them about the bugs in their programs. Why? The customer didn't put them there—the developer did.

The Nature Of Support

Let's say you are a responsible software developer and you have a schedule for bug fixes. How does the customer benefit from these repairs? First, if the repair is for a feature that seriously cripples the functional ability of the product—or worse, destroys the user's data—new disks should be sent to the entire installed base of users at absolutely no cost. Customers should be encouraged to send registration cards to software developers, in order for those companies to be able to be reached. If car companies have to conduct recalls for oversights that can damage their users, we need to support recalls for software that can damage data.

As far as I'm concerned, it's fine to ask the customer to return the original disk first, so you don't have to pay for new media, but it's not OK to charge for "upgrades" that are actually bug fixes. If you're concerned about the cost of upgrading customers for life, build this cost into your product price in the first place, and let the customer know why the product costs so much. The customer who balks at a $100 software package because

disks only cost $2 needs to be told where that money is going. On the other hand, it is annoying for a customer to be charged an extra $10–$50 for a disk that repairs certain bugs not present on new releases of the product, while the new releases are sold to new customers at the old price—new customers don't have to pay extra for the "debugged" version. Remember, your customers are your friends, and you should treat them as such.

Oversold Programs

While not in the category of bugs, there is an aspect of software marketing that can cause as much user frustration as a buggy program. I'm talking about programs with "features" that never made it from the ad copy to the product.

As a member of the press, I'm deluged with glitz on a daily basis. Most of the programs I get are pretty good, but lately I've been getting some heavily-hyped software that's a waste of electricity to boot up. In fact, it seems the fancier the promotion, the worse the software. Tragically, much of the Apple IIGS software falls into this category. This same software is backed by megabucks ad campaigns designed to get it into the lap of every Apple IIGS owner.

Unfortunately, because many users don't know who to blame, I've been hearing from readers who are complaining about the "slow speed" of the IIGS graphics, when in fact, the fault lies in the overpromoted software, not in the computer. By the way, I'm also an Apple IIGS software developer and, yes, development for this machine is hard, but that's my problem, not the customer's.

Some software companies appear to have allocated too much money for advertising and not enough for software development. This approach leads to immense initial sales, and based on letters from readers, it leads to a lot of dissatisfied customers. It's time we reexamined our priorities.

Why Are You In Business?

The whole foundation of business is to find a need and fill it. Businesses exist to serve the needs of their customers. In our quest for financial wealth, many people place the focus on money as though that were a goal in itself.

While money is an important part of business, and profitability is essential, the way we think of money needs to be examined. I've seen a lot of companies that operated from the spreadsheet. The entire focus was on bringing in the maximum amount of cash with the least amount of effort. Most of these companies have disappeared, but for a while, it seemed their stars were rising. An alternative, and far healthier view, is taken by companies based on the notion of service to customers. In such organizations, money is viewed as "applause." It is the reward that comes from providing service—from truly meeting the needs of customers. This message lies at the heart of the numerous "in pursuit of the search of one-minute excellence" books on the market today. And yet, in their quests for the easy buck, many companies still fail to understand the functions of their business.

Companies that don't take responsibility for their products not only hurt themselves, they also hurt our entire industry. My readers have showered me with horror stories (no more please!), and I have, in a few cases, been so shocked that I called the companies in question to try to get things straightened out. In most cases the problem was one of miscommunication, so the problem was solved. However, in a disturbing number of cases, the attitude was one of *caveat emptor*—an attitude that has no place in our society today.

Let's put some pride back in our industry. Let's realize that our customers are our most treasured asset, and let's start treating them as we should. If we do, we will all be rewarded beyond our dreams. Developers will be able to make a living from their efforts, and computer users will get the quality and service they deserve.

Dr. Thornburg welcomes letters from readers and can be reached at P.O. Box 1317, Los Altos, CA 94023.

Fred D'Ignazio, Associate Editor

Computer Learning Month

You will be getting this issue just as you gear up for another school year as an educator, a parent, or a student. In case you haven't heard, October 1987 has been designated by Congress as Computer Learning Month. It's an opportunity to think about how computers can enrich learning in the classroom and at home, and to remember how far we've come in a few short years.

I began writing this monthly column for COMPUTE! back in April 1982. At that time only about 20 percent of U.S. schools used computers in the classroom. Now, five years later, the figure is above 96 percent (99 percent of all public middle and secondary schools).

In 1982, there were only a few thousand computers in public schools. Now there are more than one million. Over 15 million students and 500,000 teachers use computers.

These are momentous changes. But even more important is how computer use has evolved. Back in the early 1980s, almost all computer instruction centered on learning programming and the internal operation of computers. Computers were taught in special laboratories, separate from regular classrooms and regular teachers. Boys swarmed around computers, both during and after school, but girls were nowhere to be seen.

Today, programming is still taught in many schools (for good reason—as a problem-solving and thinking skill and to convey a working understanding of how computers process information). However, most schools are moving away from programming instruction and are trying to integrate computers into the day-to-day curriculum and to encourage teachers to use computers to enrich lessons in a variety of subjects.

I recently attended the National Educational Computing Conference (NECC) in Philadelphia. I got a moment to sneak away from the hubbub on the floor, climb the stairs, and look out over the area from the relative quiet of the pressroom. What a view! Educational computing—which I once equated with my high school math teacher sneaking several of us boys into his closet in the back of the room to do calculus problems on his homemade analog computer—now was big time. Dozens of colorful booths filled the giant arena. The floor was alive with hundreds of excited educators rushing from booth to booth to catch up on the latest products and ideas. Gone were the booths focused on computer science, algorithms, and data structures. In their place were signs advertising new programs to fit into the K–6 science curriculum, how to use computers in reading and writing, computers in American History, computer programs for special education. And the people had changed. Instead of the young male hackers of yesteryear, the booths—and attendees—were made up of teachers and ex-teachers, most of them female.

Women are becoming more prominent in educational computing at all levels—even back in the classroom. According to one study, boys and girls are enrolled almost equally in elective computer classes in middle and high schools.

Another difference: Five years ago, few teachers knew anything about computers, and many teachers simply avoided them. Today, "teacher training" is a top-priority goal of all major state and district-level educational computing projects. And the results are heartening. Almost a third of all U.S. teachers and more than half of all computer-using teachers have had ten or more hours of computer training.

So it's time to celebrate. And Computer Learning Month is a good time for a celebration.

But we should also use this month to set new goals. There is still much work which remains to be done. For example:

- On the average, there are still more than 37 students per computer—less than the equivalent of one computer per classroom. If we are to truly integrate computers into a student's daily learning, we need more computers.
- Students in lower-income schools have significantly less access to computers than students in upper-income schools.

Also, work needs to continue. Primary tasks are:

- To get girls more involved in computers (especially after hours, when boys tend to take over).
- To encourage teachers to incorporate computer-assisted learning into their daily lesson plans.
- To make it clear where the best computer programs fit into the curriculum; then to distribute the results so teachers can easily make use of computers when they get to the right point in their lessons.

Plan A Celebration

Schools, teacher organizations, and parent groups across America are holding events to celebrate Computer Learning Month, including contests, computer fairs, essay activities, and computer open houses for parents. Why not join them? It's likely that students and teachers at your school have been doing some exciting things with computers.

For more information about special offers, posters, and national contests, write: Computer Learning Month, Software Publishers Association, P.O. Box 19763, Washington, DC 20036-0763. ©

ST Outlook

Philip I Nelson

ComputerEyes Video Digitizer

This month we'll look at ComputerEyes, an affordable color video digitizer from Digital Visions in Needham, Massachussets. Video digitizers aren't exactly new; various eight-bit versions have been around for years. But the ST version has been available for only a short time.

From A To D

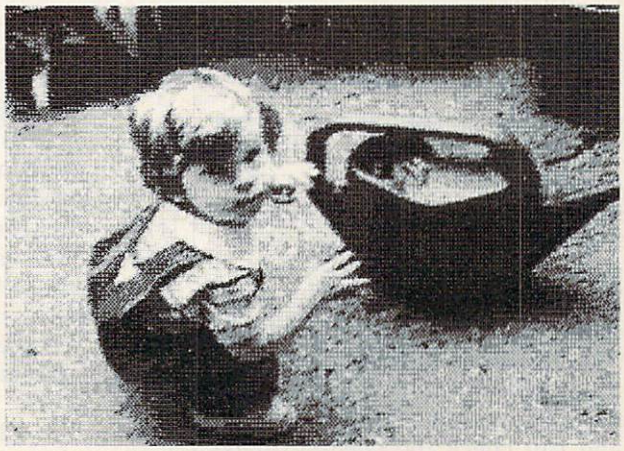
It's the job of a video digitizer to convert analog data from the world around us (light waves or sound vibrations, for instance) into a series of ones and zeros, or digital values, which a computer can handle conveniently. A device that performs such a conversion is known, not surprisingly, as an analog-to-digital converter, and that's what is contained in the fairly large cartridge that houses the hardware part of the ComputerEyes system.

Apart from a cord to the outboard 9-volt power supply, there's nothing on the outside of the cartridge but a phono jack for video input. Into that jack you can send the signal from any standard compositive video source: a VCR, video camera, videodisc player, a TV set with video output, or even another computer.

A comprehensive program, also included in the package, allows you to capture, fine-tune, and store an image as a *NEOchrome*- or *DEGAS*-format disk file. At that point, you can treat the image like any other picture file; you can enhance it with a paint program, display it in a slideshow, make a hardcopy printout, and so on.

Hold That Pose

Video images consist of a huge amount of data. Even at top processing speed, it takes the ST a minimum of several seconds to capture and store the raw image data in memory; during that time the scanned image must be absolutely motionless. In other words, ComputerEyes is not a "frame grabber"—you can't feed it changing input like a live TV broadcast and expect it to freeze the action unaided.



One solution is to record the image on videotape and then freeze it on replay for digitization. I've had excellent success using ComputerEyes with an eight-millimeter camcorder. The camcorder has onboard VCR capabilities, including a very stable pause mode. Once you've shot the tape, digitizing is simply a matter of replaying to the desired frame, pausing, and clicking Capture on the ComputerEyes program's dialog box.

Not Just A Brownie

The first temptation when you get a digitizer is to use it like a Brownie camera, digitizing everything in sight to see how faithfully the computer can reproduce reality. The figure accompanying this column was shot on eight-millimeter videotape and then was digitized using the program's 16 gray-scale mode (which actually gives you eight shades of gray plus eight colored tints of gray, since the ST's color palette has only eight pure grays). After saving the image as a lo-res *DEGAS* file, I switched to the monochrome monitor and converted the lo-res picture to hi-res so it could be typeset in black and white.

You lose a bit of detail in converting from color to monochrome, since the lo-res colors are replaced by various fill patterns. But the figure gives you a reasonably accurate idea of what is possible. The major obstacle to finer resolution comes not from the digitizer, but from the ST itself, which is limited to a 16-color palette and three different screen resolutions.

Snapshots are amusing, but the system allows you to do much more. When you capture an image, ComputerEyes initially stores it in memory as a huge table of raw numbers. Depending on what options you've chosen, the program then calculates the best color value for every pixel on the screen, displaying the result as soon as the calculations are finished. But the raw data remains in memory until you take another picture or save the current one to disk. As long as the raw data remains, you can adjust the contrast, brightness, and RGB values of the picture and then recalculate it for a new display. Sometimes a small adustment makes a dramatic change in the final image, and this scheme offers you a considerable degree of creative control.

Video digitizers aren't for everyone, but the reasonably priced ComputerEyes system is simple to use, and it offers just about every feature a home user could want.

Donald B. Trivette

Bookshelf On A Disc

Knowledge is of two kinds. We know a subject ourselves, or we know where we can find information upon it.

That quote, and scores of others, are attributed to Samuel Johnson, the eighteenth-century English writer of dictionaries. Using Microsoft's new product, *Bookshelf*, I know where I can retrieve quotations—*By necessity, by proclivity, and by delight* (Emerson)—on almost any subject. I can also look in its thesauras for alternate choices, and can check the spelling of *retrieve* in its dictionary.

Bookshelf is a reference library consisting of *Bartlett's Familiar Quotations, Roget's Thesaurus, The American Heritage Dictionary, The Chicago Manual of Style, The World Almanac and Book of Facts*, a U.S. zip-code directory, and a word-usage manual; the contents of all these books, and a few other goodies, are recorded on a disc that's identical in appearance to the audio compact disc that holds an hour of Billy Joel's music. In printed form, these books take more than a foot of shelf space; in electronic form they occupy less than 40 percent of the disc's capacity. There's plenty of room for Microsoft to add new reference works to the disc, and that is indeed the intention. Expect an encyclopedia in some future version and perhaps a directory of toll-free telephone numbers.

In order to run *Bookshelf*, you'll need a CD-ROM player, which attaches to your computer like an extra disk drive. In fact, that's just how it works. My Sony CD 100 drive is configured as drive D: on my IBM PC. The *Bookshelf* installation software has an option allowing you to install a Sony, Philips, Hitachi, or Amdek CD-ROM player. Currently, *Bookshelf* is in computer stores in two versions: about $300 for the product itself, and packaged along with an Amdek CD ROM drive for $1,300.

Using *Bookshelf* is easy, and as much fun as using any of the other pop-up resident products that are so popular for IBM PCs. Once installed and activated by the AUTOEXEC.BAT, *Bookshelf* is called by pressing the left shift and ALT keys. This causes a menu bar to appear with 12 choices at the top of the screen: Thesaurus, Dictionary, Spell, Usage, Manual, Almanac, Quote, BIS, ZIP, Forms, Options, Help. Using the cursor keys to highlight a choice and then pressing Enter selects a reference work.

Zipping Around

Let's look at the zip-code directory. First, a window appears on the screen over the top of the letter or document you're editing. The window contains boxes for you to enter a street or post office box number, a city, and a state. Once the state is entered, the *Bookshelf* search software, called a retrieval engine, goes to work on the files located on the CD-ROM disc. In the case of zip codes, there's a separate file for each state; the file contains the names and zip codes for each city and town, and in towns where there are multiple zip codes, there's a street directory complete with house numbers, where appropriate. Entering 1600 Pennsylvania Avenue, Washington, DC, causes the zip code, 20006, to display on the screen faster than President Reagan vetoes tax hikes. The zip-code directory software has a special address parser that will automatically extract a street, city, and state when the cursor is left one space beyond the state name. When used in this mode, the zip code is automatically inserted at the proper place.

For this and the other features to work correctly, the program must have an intimate knowledge of your word-processing software and how it works. *Bookshelf* was designed and tested to be compatible with *Bank Street Writer, Displaywrite, Easy Writer, Microsoft Word, MultiMate, New Word, PC-Write, Perfect Writer, pfs:Write, Super Writer, Volkswriter, WordPerfect, WordStar*, and *XYWrite III*. Support for other popular word processors will likely be added in the future, but even if your favorite isn't listed, you can still use *Bookshelf* either from within your program or from DOS—you just may not be able to take advantage of cut-and-paste and automatic parsing features.

Vincy, Vinchy

Selecting Quote (from *Bartlett's Quotations*) causes *Bookshelf* to display a pull-down menu offering three choices: Search, Table of Contents, and General Index. The search screen, which appears over the document you're editing, has three boxes similar to the zip-code locator. In these you type not an address, but rather the term or terms defining the quotation. Want to know who said what about spelling? Type the word *spell* in the first box and press enter (the other boxes may be used to narrow the scope of a search). The first lines of five quotations appear. By highlighting with the cursor, you can zoom in on an entire quote. In this case, the third quote is one from Mark Twain's *Innocents Abroad*, Chapter 19: "They spell it *Vinci* and pronounce it *Vinchy*; foreigners always spell better than they pronounce."

Bookshelf is a product that will significantly affect the way we use computers. It has dozens of features I'd like to describe, but I get paid to write only 1000 words. I keep telling my editor what Samuel Johnson said: *No man but a blockhead ever wrote except for money.* ©

Microscope

Sheldon Leemon

Not too long ago, the closest thing to desktop publishing on your home computer was using *Print Shop* to produce greeting cards. Turning out professional-quality documents on a PC required a desktop publishing system that cost close to $10,000. While some businesses might find this inexpensive when compared to the cost of typesetters, graphic artists, and offset printing, it's clearly beyond the reach of most individuals and small businesses. Lately, however, there have been a number of developments which indicate that the ante required to get into desktop publishing may soon drop to the $3,500 price range, or even lower.

The first two requirements for desktop publishing are a powerful microcomputer and some good software. The two large home computer manufacturers, Atari and Commodore, each have machines that fit the bill at very low prices. The Atari ST and Commodore Amiga systems offer a fast 68000 processor, a megabyte or more of memory, 640 x 400–resolution display capability, and high-capacity disk storage—all for about $1,000. The second requirement, the software, has been slower in coming, but there are promising signs. Soft Logik's *Publishing Partner* has been released for the ST, and Atari has announced that *Ready-Set-Go*, one of the first desktop publishing programs available for the Macintosh, is being ported over to the ST. On the Amiga side, three desktop publishing programs—*PageSetter*, *Publisher 1000*, and *City Desk*—are now available. The ST and Amiga programs clearly aren't as strong as industry leaders like *PageMaker* and *Ventura Publisher*, but they're quite good for first attempts, and no doubt will get even better.

****

With the hardware and software in place, the only remaining problem is how to get the output onto paper. No matter how good the hardware and software are, a document is still going to look rather primitive if it's printed with a nine-pin dot-matrix printer. The printer of choice for desktop publishing is a laser printer. Although the output from these machines looks almost as good as offset printing, they usually cost more than the computer system that drives them. Although less than half the price of the Apple LaserWriter, a typical representative of the current lot of these printers still costs about $3,000. The next generation, however, may well cut that price tag in half again.

Atari, for example, has announced an under-$1,500 laser printer for the Mega ST line of computers. Atari's approach to cost cutting has been to take most of the electronic "brains" out of the printer and to have the computer control it directly. This means, however, that the printer will work only when attached to an ST with a few megabytes of memory. Though Atari's price sounded shockingly low when the printer was announced in January, some competition is on the way. Okidata recently introduced the LaserLine 6, a 6-page-per-minute laser printer, and Panasonic has announced the Laser Partner, an 11-page-per-minute machine. Though the retail price for these machines is $2,000; the most probable street price appears to be about $1,500. In fact, deep-cut discounters have already been advertising the Okidata printer for as low as $1,300. While that may not seem inexpensive to you, it's a far cry from the LaserWriter, and in fact, it is not much more than office-quality impact printers cost.

****

As laser technology becomes more commonplace, prices for laser printers will undoubtedly fall still further. In the meantime, however, impact-printer technology is also moving ahead. Dot-matrix printers with 24-pin print heads have finally begun to come into their own. By using a larger number of very fine wires to print each character, these printers achieve a much higher resolution than the older nine-pin printers. This means that they can produce typewriter-quality output like a daisywheel printer, at higher speeds, and yet still be used to print out charts and other business graphics, like other dot-matrix printers. For this reason, 24-pin printers have begun to make serious inroads into the daisywheel printer market, and appear to be poised to take the home and small business markets by storm as well. At the June COMDEX show, almost every printer manufacturer had at least one 24-pin model to show.

At prices starting in the $650–$750 range, most of these printers are about $200–$300 more than their nine-pin counterparts. But NEC has recently announced a new 24-pin model, the NEC 5500, that lists for only $499. Like many of the new crop of 14-pin printers, the 5500 boasts a graphics density of 360 dots per inch, as compared to the 300-dots-per-inch density of most laser printers. So while laser output may be faster and cleaner, the quality from 24-pin printers should still compare very favorably, at a price little more than the least expensive of any current printer. When software gets around to taking advantage of these powerful new impact printers, desktop publishing may become an application that adds only $300–$400 to the price of a home computer system, with complete systems available for well under $2,000. ©

Sheldon Leemon

Behind The Workbench

Often, you can tell the level of a user's experience by the kinds of questions he asks. Someone who's worked with the Amiga for a while might say that he's having trouble loading a certain program, while an absolute beginner will complain, "I can't get into this disk." It's usually not too hard to figure out his problem, but the way he has stated it shows that he really doesn't have any concept of how files are stored on a disk or how you find out information about these files.

Part of the problem is that the Workbench makes it so easy to operate the computer using the mouse and icons that most of the time the user doesn't have to worry about what's really happening when he moves those pictures. So the first time he comes across a program disk with no icons on it at all, his natural reaction is, "What do I do now?" At that point, harsh reality can no longer be hidden behind pretty pictures. It's time to sit down and have that difficult talk about the bits and the bytes.

Storage In Files

A 3½-inch Amiga disk holds over 900,000 characters of information, but letters and numbers aren't stored on the disk randomly. Instead, information is stored in named files, which are made up of related alphabetic or numeric characters, or of computer instructions. For example, all of the computer instructions that make up a word-processing program are stored in a single program file. If you use the word processor to create and save a document to disk, all of the characters that make up the document are stored in a single data file.

A file has a number of attributes, such as its name, size, and so on. To find out this information, you need a special type of program. The Workbench program, for example, displays some files as icons. The name of the file appears below the icon, and you can find out additional information by highlighting the icon and using the Show Info menu item. The Workbench program only displays icons for certain types of files, however (as we saw in the August column). Therefore, you must use another type of program to find information about the rest of the files.

One such program is called the CLI (Command Line Interface). You can find the icon for this program in the System drawer on the Workbench disk (but only if you remember to turn the CLI switch on the Preference screen to ON). When you start the CLI program, all you'll see is an empty window with this prompt: *1>*. This means the program is waiting for you to type a command. If you type the DIR command, you'll get a list of all of the files in the current directory. Type DIR OPT A, and you'll get a list of all of the files in each subdirectory as well. To list all of the files on an external disk drive, you could type DIR DF1: OPT A. There are a lot of other CLI commands, and they are discussed in depth in books like *COMPUTE!'s AmigaDOS Reference Guide*. But the DIR command is enough to let you start looking at disk filenames.

Programs That Help

Although the CLI program is included with every Amiga, it's harder to use than the Workbench, since it requires you to remember a lot of commands. Several intermediate programs are available that combine the power of the CLI with the ease of use of the Workbench. For example, there are public domain programs like DirUtil, which let you display the list of files on a disk in a scrolling window. These programs may be found on bulletin board services, commercial information services like CompuServe and GEnie, and in collections of public domain software like the one put together by Fred Fish. There are also some commercial programs that provide even more elaborate functions, such as *CLI Mate* and *Zing*.

Once you've used the CLI DIR command, or one of the other programs described above, you'll soon find out that there can be hundreds of different files on a disk. Some are program files, but many are data files. A data file is just a collection of information, like a word processing document, or a picture from a paint program. You can't run these files, since they don't contain computer instructions like a program does. Data files are helpful only when used in conjunction with a program that knows what to do with them. How can you find out what a particular file does? One way is to try and run that file. If you type in the filename at the CLI prompt, the CLI will try to run the program. If you get back a message that says that it was *not an object module*, you'll know that it's a data file, and not a program.

Another way to tell is to pay close attention to the name. For example, sometimes, data file names will end in a string of characters that indicate which program will run them, such as .pic for a picture file, or .bas for a BASIC program file. And, files named Readme, Manual, or that end in the characters .doc or .txt will usually contain text, often instructions on how to run the program. To read these files, use the CLI TYPE command (TYPE Readme, for example), or load them into your favorite word processor. If you look at the instructions first, you'll often find an explanation of what some of the other files are for, as well. ©

Bill Wilkinson

Machine Language Graphics: The Final Installment

This month I will finally show you the machine language equivalents of the most important BASIC screen I/O operations. We'll begin with an example. Suppose we wanted to implement a GRAPHICS 7 statement. From two months ago, we know that the equivalent low-level statements are

```
CLOSE #6
OPEN #6,12+16,7,"S:"
```

A direct translation into machine language follows.

```
;GRAPHICS 7
;CLOSE IOCB 6
  LDX  #$60            ; IOCB number
  LDA  #12             ; the CLOSE command
  STA  ICCOM,X         ; put in place
  JSR  $E456           ; call CIO
;OPEN  IOCB 6
  LDX  #$60            ; IOCB number
  LDA  #3              ; the OPEN command
  STA  ICCOM,X         ; put in place
  LDA  #12+16          ; give it the same
                       ; value
  STA  ICAX1,X         ; as you would in
                       ; BASIC
  LDA  #7
  STA  ICAX2,X
  LDA  #DEVICE&$FF     ; don't worry why
  STA  ICBAL,X         ; this works
  LDA  #DEVICE/$100    ; it just does.
  STA  ICBAL+1,X
  JSR  $E456           ; do the real work
```

Don't bother assembling this code yet—it won't work without some of the help given later in this article.

Now, if you all you ever wanted to do was emulate GRAPHICS 7, that would be an adequate method. But in BASIC, the general form of the command is GRAPHICS *mode*, where mode is any numeric variable or expression or your choice. It would be better if we could emulate *that* in machine language. And, to some degree, we can.

In BASIC's GRAPHICS statement, the mode value is called a *parameter* to the operation. In machine language, we also use parameters. With the 6502 microprocessor that Atari machines use, we usually try to pass the parameters in one or more of the three registers that the chip possesses: the A register (also called the accumulator), the X register, and the Y register. Suppose you need to pass an IOCB number. Since it needs to be in the X register for the call to CIO anyway, why not pass it there?

The listing that follows is *not* a program in and of itself. Rather it is a set of subroutines that your program may call (via JSR) to implement the given operation. At the very end of the article you will find a sample program that calls these subroutines.

When you use the subroutines in your own programs, you must note carefully the description of the parameters that I have given. Be sure that the appropriate registers contain the proper values before you jump to a subroutine.

The listing is given without line numbers. Some assemblers use line numbers, but, when they do, it's for editing purposes only—the numbers have no effect on the program. Comments are preceded by a semicolon. You may omit any of them that you like. When you have typed all this in (and have checked it *carefully* for errors—one mistake can cause a lockup), you should save it (or LIST it, depending upon your assembler) to disk or tape. You can then use it as the nucleus of your own graphics programs.

```
;
; Equates
;
; Without these, the program won't assemble properly
;
ICCOM   = $342  ; the COMMAND byte in the IOCB
ICBAL   = $344  ; the low byte of the buffer address (filename)
ICBLL   = $348  ; the low byte of the buffer length
ICAX1   = $34A  ; auxiliary byte 1: type
ICAX2   = $34B  ; auxiliary byte 2: mode
;
CIO     = $E456 ; Central Input/Output routine
ROWCRS  = 84    ; ROW CuRSor—y position
COLCRS  = 85    ; COLumn CuRSor—x position
ATACHR  = 763   ; where line color goes for DRAWTO
;
; Now the working routines
;
; REMEMBER: these are only subroutines
; You must call them via JSR from your own code
;
;
; CLOSE channel
;
; Parameter:  X register holds IOCB number
; On exit:    Y register holds error code
;
CLOSE
  LDA  #12              ; close command
  STA  ICCOM,X          ; in place
  JMP  CIO              ; do the real work
;
;
; OPEN channel,type,mode,file
;
; Parameters: X register holds IOCB number
;             A register holds type
;             Y register holds mode
;             the address of the file/device
;             name must already be set up
;             in the IOCB—
; On exit:    Y register holds error code
;
OPEN
  STA  ICAX1,X          ; the type value
  TYA
```

```
 STA  ICAX2,X        ; and the mode, if appropriate
 LDA  #3             ; OPEN command
 STA  ICCOM,X        ; in place
 JMP  CIO            ; the real work
;
;
; GRAPHICS mode
;
; Parameter:  A register holds desired mode
; On exit:    Y register holds error code
;
GRAPHICS
 PHA                 ; save the mode for a moment
 LDX  #$60           ; always use IOCB #6
 JSR  CLOSE          ; be sure it is closed
 LDX  #$60           ; the same IOCB again
 LDA  #SNAME&$FF     ; the "S:" device name
 STA  ICBAL,X        ; must be put in place
 LDA  #SNAME/$100    ; before we go further
 STA  ICBAL+1,X      ; (take this part on faith)
 PLA                 ; recover the GRAPHICS mode
 TAY                 ; put it where OPEN wants it
 AND #16+32          ; isolate the text window and no-clear bits
 EOR  #16            ; flip state of the text window bit
 ORA  #12            ; allow both input and output
 JMP  OPEN           ; do this part of the work
;
;
; PUT channel,byte
;
; Parameters: A register holds byte to output
;             X register holds channel number
; On exit:    Y register holds error code
;
PUT
 TAY                 ; save the byte here for a moment
 LDA  #0
 STA  ICBLL,X        ; $0000 to length
 STA  ICBLL+1,X      ; as noted last month
 LDA  #11            ; the command value
 STA  ICCOM,X
 TYA                 ; data byte back where CIO wants it
 JMP  CIO
;
;
; byte = GET( channel )
;
; Parameter:  X register holds IOCB number
; On exit:    A register holds byte from GET call
;
GET
 LDA  #0
 STA  ICBLL,X        ; $0000 to length...
 STA  ICBLL+1,X      ; as noted last month
 LDA  #7             ; the command value
 STA  ICCOM,X        ; where CIO wants it
 JMP  CIO            ; believe it or else, that's all
;
;
; PLOT x,y,color
;
; Parameters: A register holds color
;             X register holds x location
;             Y register holds y location
; NOTE: not for use with GR.8 or GR.24
;
PLOT
 STX  COLCRS         ; see my August column
 STY  ROWCRS         ; these are just POKEs
 LDX  #$60           ; the S: graphics channel
 JMP  PUT            ; color is already in A
;
;
; byte = LOCATE( x,y )
;
; Parameters: X register holds x location
;             Y register holds y location
; On exit:    A register holds color of point at (x,y)
;
LOCATE
 STX  COLCRS         ; again, see column
 STY  ROWCRS         ; from two months ago
 LDX  #$60           ; the S: graphics channel
 JMP  GET            ; color returned in A
;
;
; DRAWTO x,y,color
;
; Parameters: A register holds color
;             X register holds x location
;             Y register holds y location
; NOTE: not for use with GR.8 or GR.24
;
DRAWTO
 STX  COLCRS         ; once more: see the article
 STY  ROWCRS         ; from two months ago
 STA  ATACHR         ; location 763, also in that article
 LDX  #$60           ; again, we use IOCB #6
 LDA  #17            ; the XIO number for DRAWTO
 STA  ICCOM,X        ; is actually the command number
 JMP  CIO            ; and that's all we really need to do
```

Now that we have these routines, how do we use them? A full explanation would need at least the beginnings of a tutorial book. But here's a short example. First, a small program in BASIC:

```
100 GRAPHICS 3+16
110 COLOR 2 : PLOT 10,10
120 COLOR 3 : PLOT 20,20
130 COLOR 1 : PLOT 0,15 : DRAWTO 30,15
140 GOTO 140 : REM (just wait for RESET)
```

Now the same thing in machine language, using the routines of the program above. The only decision you will have to make is where in memory to place the assembled code. My first line reflects what should be a safe choice for most assemblers used in most 48K byte (or 64K byte) machines. If your assembler has a SIZE or MEMORY command, use it to get an idea of what is safe. In any case, LIST or SAVE your code to disk or tape before assembling, just in case.

```
              *= $6000      ; my "usually safe" location
;
START LDA  #3+16            ; first
      JSR  GRAPHICS         ; emulate GRAPHICS 19
;
      LDX  #10              ; now do PLOT 10,10
      LDY  #10              ; (x and y locations)
      LDA  #2               ; with COLOR 2, a slight
      JSR  PLOT             ; change from BASIC, but close
;
      LDX  #20              ; similarly:
      LDY  #20              ; we want PLOT 20,20
      LDA  #3               ; with COLOR 3
      JSR  PLOT             ; one call does it all
;
      LDX  #0               ; last PLOT:
      LDY  #15              ; PLOT 0,15
      LDA  #1               ; with COLOR 1
      JSR  PLOT             ;
;
      LDX  #30              ; and now the DRAWTO:
      LDY  #15              ; DRAWTO 30,15
      LDA  #1               ; still with COLOR 1
      JSR  DRAWTO           ; the routine does the work
;
LOOP1 JMP  LOOP1            ; (loop here until RESET is pressed)
;
; Append all of the code for all of ; the subroutines here
;
; Your assembler may need .END or END as
;    the very last line
;
```

The Beginner's Page

C. Regena

Detecting Keypresses

Last month, this column discussed INPUT for interactive programming. INPUT receives whatever the user types before he or she presses the RETURN or ENTER key, whether it is one character, several lines of characters, function keys, cursor keys, or no other keys. The user can enter something that the computer is not expecting and cause the program to crash, act strangely, cause later errors, or even end unexpectedly. You can make your programs more user-friendly and limit the input by scanning the keyboard instead.

This method of interaction receives input by scanning the keyboard. The key pressed is not printed on the screen. This method is useful if you want your user to respond with just a one-key answer, such as in a yes or no situation; a multiple-choice or menu selection; or selecting a certain letter or number.

The command to retrieve a single keypress depends on your computer. The Commodore and Applesoft versions of BASIC both use GET. The versions of BASIC for the Amiga and IBM PC and compatibles use INKEY$. (Amiga and IBM BASICs also have a GET statement, but it has a different function in these versions and cannot be used to retrieve keypresses.) The versions of BASIC for the Atari ST and Atari eight-bit models have no built-in command explictly for reading a single keypress, but alternative techniques for these machines are discussed at the end of this article.

A sample program segment would be

```
200 PRINT "PRESS Y FOR YES OR N
    FOR NO"
210 K$=INKEY$
220 IF K$="" THEN 210
230 IF K$="Y" THEN 500
240 IF K$<>"N" THEN 210
250 REM PROGRAM CONTINUES FOR
    N RESPONSE
500 REM PROGRAM CONTINUES FOR
    Y RESPONSE
```

Line 200 prints a message so the user will know he or she needs to do something. Line 210 assigns the value of whatever key is pressed to the variable K$. The line works as shown for the IBM and Amiga. If you have a Commodore or Apple, replace line 210 with

```
210 GET K$
```

Line 220 is used in case no key is pressed. It is necessary because INKEY$ and the Commodore version of GET do not wait for a keypress. If no key is being pressed, these commands return a *null string*, "". Line 220 detects this and keeps the program in a loop until a key is pressed. Such loops are often used in conjunction with INKEY$ or GET. Note, however, that the Applesoft version of GET does wait for a keypress. Thus, Apple users can omit line 220. The GETKEY statement in BASIC 7.0 for the Commodore 128 has the same effect—it waits until a key is pressed, eliminating the need to test for a null string.

When a key is pressed, line 230 shows that if the key pressed is a Y, the program should pass control (branch) to line 500 and continue. Line 240 checks for the N key. If the key pressed is not N, then the computer branches back to line 210 to wait for another key to be pressed. If an N is pressed, the program continues. Notice that this prevents user error by accepting only Y or N and ignoring any other keypress.

You don't have to limit your input choices to Y and N. With appropriate IF-THEN tests you could look for any key, or set up an entire menu of single-letter choices.

This method is also often used to create a pause in a program until a certain key is pressed. For example, to allow a user to read a screen of instructions at his or her own speed, you can have the program wait until the space bar is pressed:

```
200 PRINT "PRESS THE SPACE BAR
    TO CONTINUE."
210 S$=INKEY$
220 IF S$<>" " THEN 210
```

(This and the following examples all use INKEY$. If you're using a Commodore or Apple, remember to replace that statement with an appropriate GET.)

Line 220 contains a space between quotation marks. If the value of S$ is not a space, then the program branches back to line 210.

Instead of specifying a particular key, you can allow the user to press any key to continue:

```
200 PRINT "PRESS ANY KEY TO
    CONTINUE"
210 C$=INKEY$:IF C$="" THEN 210
```

The IF-THEN statement in line 210 keeps the computer in a loop until a key is pressed. Note that there is nothing between the quotation marks in that line. Since the Applesoft version of GET and the Commodore 128's GETKEY wait for a key to be pressed, the IF test can be omitted if those statements are used.

Some keys that you may want to detect cannot be printed between quotation marks in a program. Examples include the RETURN key and the function keys. You can use CHR$ to check for any valid character. For example, CHR$(13) is the same as the RETURN or ENTER key. Look in your computer manual to find out the ASCII codes for the cursor keys or function keys. Here is an example using CHR$:

```
200 PRINT "PRESS RETURN TO
    CONTINUE."
210 R$=INKEY$:IF R$="" THEN 210
220 IF R$<>CHR$(13) THEN 210
```

Another method is to check the ASCII code of the key pressed:

```
300 PRINT "PRESS RETURN TO
    CONTINUE."
```

```
310 R$=INKEY$:IF R$="" THEN 310
320 IF ASC(R$)<>13 THEN 310
```

Note that the IF-THEN statement in line 310 is necessary (except on the Apple) because the ASC function in line 320 causes an error when asked to return the ASCII value of a null string.

INKEY$ always returns a string character. You may choose any string variable name. GET can retrieve either a string or numeric value; you select which one by the variable type you use following the GET statement. For example, a statement like GET A$ returns a single-character string, while one like GET A returns a single-digit number (0–9). If you were to type a 1, GET A$ would return the character 1, equivalent to CHR$(49), while GET A would return the numeric value 1. The string form is much preferred because the numeric form will stop the program with an error message if any key other than 0–9 is pressed.

If you really want a number rather than a string, you can use the VAL( ) function to convert the string to a number. Here is an example: Suppose you have printed a menu screen with four choices, numbered 1–4. The user needs to press 1, 2, 3, or 4. All other keys are to be ignored.

```
400 N$=INKEY$:IF N$="" THEN 400
410 IF N$<"1" OR N$>"4" THEN 400
420 ON VAL(N$) GOTO 1000,2000,
    3000,4000
```

When testing characters with IF-THEN statements, you should note that the case of the character is significant. The computer does not recognize *Y* and *y* as being equivalent, nor *N* and *n*. Your IF-THEN statements must be more complex to check for both uppercase and lowercase letters. For example:

```
400 PRINT "PRESS Y OR YES OR N OR
    NO."
410 A$=INKEY$:IF A$="" THEN 410
420 IF A$="Y" OR A$="y" THEN 1000
430 IF A$<>"N" AND A$<>"n" THEN
    410
440 REM PROGRAM CONTINUES FOR
    N RESPONSE
```

If you have an Amiga, Amiga Basic has a handy solution—the UCASE$ function. This converts any specified string to all uppercase characters. For example, the following lines retrieve a character C$ and insure that it will be uppercase. Any following IF-THEN statements need only check for uppercase characters:

```
PICK:
C$=INKEY$:IF C$="" THEN PICK
C$=UCASE$(C$)
```

Scanning The Keyboard Buffer

Another fact that you should be aware of in scanning the keyboard is that some computers have a *keyboard buffer*. When the computer detects a keypress, it stores the key value in memory until it is called for (up to a maximum of 10–15 keypresses, depending on the computer). Thus, keys pressed when a program is not looking for input may be retrieved by subsequent INKEY$ or GET commands.

It is wise to clear the buffer before you use any INKEY$ statement. Some computers have a POKE command to clear the keyboard buffer. (For example, POKE 198,0 does the job on the Commodore 64). Another method—if you know how many keypresses the buffer can hold—is to use a FOR-NEXT loop. For example, the IBM PC keyboard buffer holds 15 keypresses:

```
50 FOR K=1 TO 15:A$=INKEY$:NEXT
   K
60 PRINT"PRESS ANY KEY TO
   CONTINUE"
70 A$=INKEY$:IF A$="" THEN 70
```

Alternative Tactics

As mentioned earlier, the version of BASIC for Atari's eight-bit computers has no built-in statement designed specifically to retrieve a single keypress. However, you can achieve the same effect by first opening a channel to the keyboard device with the statement OPEN #1,4,0,"K:", then using GET#1 to retrieve character values. (Any unused file number in the range 1–7 can be substituted for the 1 in these statements). The main difference between the Atari's GET#, and the INPUT$ and GET commands described above, is that GET# returns character code values rather than string characters, so GET# must always be followed by a numeric variable. If characters are desired, the CHR$ function can be used. Also, note that GET# waits for a key to be pressed.

To modify the first example program segment in this article for eight-bit Ataris, you must first add lines to open the channel for input and set up the testing variable:

```
100 OPEN #1,4,0,"K:"
110 DIM K$(1)
```

This needs to be done only once in the program, but it must be done before the first GET# command. Then replace line 210 with:

```
210 GET#1,K:K$=CHR$(K)
```

You may find it easier to simply check for character code values rather than characters. Refer to your BASIC manual for a complete list of Atari ASCII codes. For example, you could test for a RETURN keypress with

```
300 PRINT "PRESS RETURN TO
    CONTINUE."
310 GET#1,R:IF R<>155 THEN 310
```

ST BASIC, for the Atari STs, likewise has no practical command for retrieving a single keypress. (There is an INPUT$ function, but it is rather unwieldly.) Fortunately, you can achieve the same effect using the INP function. This function retrieves a value from one of the computer's internal input/output (I/O) ports. The ST's keyboard is actually an intelligent peripheral with its own microprocessor. It returns keypress information to the computer through port 2. Thus, INP(2) returns the ASCII value of the next key pressed. (The function will wait until a key is pressed.) A statement like K=INP(2) will assign the ASCII code of the next key pressed to the variable K. If you want characters instead of ASCII values, you can duplicate the INPUT$ function with a statement like K$=CHR$(INP(2)).

The following program segment illustrates one way to read the Atari's function keys:

```
400 print "Press any function key:"
410 k=inp(2)
420 if k<187 or k>196 then print "That's
    not a function key!":goto 410
430 print "You pressed the F";
    K-186;"key."
```

Arlan R. Levitan

Top Modem

Group Commander Arlan Smythe-Fawlty grimly surveyed the horizon for any sign of the enemy as he nervously checked his Spitfire's fuel gauge. He glanced to his right and located the familiar silhouette of his wingman's P51-Mustang. As if on cue, the Mustang's pilot, "Maverick" Duffy, flipped his radio on.

"I've got Major Von Kreuger and Baron Schultz on radar Arlan. They're flying Messerschmitts today. Let's come to heading 230, climb to 8000 feet and come in from behind."

"Roger, Maverick. Beginning roll on the count of three. Don't fire until we're within 1000 yards."

This is *Air Warrior*, a fascinating realtime air combat game that is gaining a dedicated following on the GEnie information service. The game uses sophisticated software on your micro and the GEnie mainframe, working in tandem to produce a quick, responsive multiplayer online flight simulator.

The world of *Air Warrior* is populated by three countries, apolitically named A, B, and C. Their macrocosm is an eight by eight map grid that represents about 1600 square miles of terrain.

About a dozen different aircraft of World War I and II vintage may be flown. Each plane has different performance characteristics, but a little "seat of the pants" flying can go a long way toward evening the odds. I once managed to knock a Zero out of the sky with an old Fokker Triplane by putting myself into a spin that convinced my opponent that I was a goner, and then pulling up at the last moment with guns blazing. Some of the planes have data files that give you a realistic view of your own plane as you scan the skies. You can even ride with other players on planes that can carry a crew, such as the B-17 bomber.

First Fly—Then Shoot

To be successful in *Air Warrior*, you must master the rudiments of flying, just like the popular single-player flight simulators that are available for many personal computers. The in-flight display of Air Warrior will make experienced microcomputer flyers feel right at home. It is very similar to other flight programs, complete with an instrument panel and animated scenery that depicts mountains, roads, and buildings.

The flight controls operate in both novice and expert modes. In novice mode, the aircraft are very forgiving of maneuvers that would normally place the craft into stalls or tailspins. Alternatively, expert mode offers maximum maneuverability and performance and won't stop a foolish pilot from putting a craft through a high-stress maneuver that will rip its wings off.

As in real flight, other aircraft appear to be small specks until you get within 500 yards or so, at which time their silhouettes are easily discernible. You must be on the lookout for other planes by checking your radar. Players can send suitable taunts to each other during play as they take potshots at each other.

Air Warrior software, developed by Kesmai Corporation, may be downloaded from GENIE. It is currently available for the Macintosh and Amiga, but versions for the IBM PC and Atari ST should be ready in late fall. Players are notified upon entering the game when new releases of the *Air Warrior* software and/or terrain maps will be available for downloading.

Prospective aces should be forewarned that playing *Air Warrior* can be like eating potato chips: Near the end of your mission, you'll want to play "just one more" minute, then another minute, and another. Toward the intended end of my last mission, I decided to play for just a couple more minutes; the next time I glanced at the clock, three hours had passed.

Even so, *Air Warrior* is a delight to play and a marvel of programming prowess. If you decide to take up the challenge, download the software and spend a lot of time reading the manual and flying in the program's offline practice mode. This lets you master basic flight maneuvers and commands before taking to the air while the meter is running. You can enhance your in-flight survivability significantly by studying some of the *Air Warrior* download files that discuss dogfight tactics.

Half The Battle

Knocking your opponents out of the sky is only half the battle—you must also safely return and land at a friendly air strip to get full credit for your victories in the air. *Air Warrior* keeps a running total of your performance and won/lost records. Updated pilot standings are posted every week or so.

If you don't have a GEnie ID, you can apply for one online via computer by selecting half-duplex (local echo mode) in your terminal program and dialing 1-800-638-8369 at 300 or 1200 bps. When connected, type HHH and press the return key. At the *U#=* prompt, enter *XJM11868,AIR* and press return. You can also get info on GEnie via voice at 1-800-638-9636.

Remember, in *Air Warrior* there are no delayed flights or lost baggage. See you in the not-so-friendly skies. ©

Chrome

Double Hi-Res Graphics Commands For Applesoft

Zachary T. Smith

If you are frustrated by the lack of firmware support for the latest Apple graphics modes, you'll be more than pleased with this Applesoft BASIC extension that gives you six new graphics commands for the double hi-res mode on the Apple IIe (Revision B), IIc, and IIGS computers. DOS 3.3 is required.

Here's a utility that extends Applesoft with six new commands and a revised HPLOT to provide full control over the double hi-res graphics mode. It supports both monochrome and 16-color plotting and leaves 21.5K of RAM for the Applesoft programmer.

The whole idea of double high-resolution video (including 80-column text) on the Apple seems to be more of an afterthought than an innovation. For example, the original ROM routine for scrolling the 80-column screen behaved more like two successive 40-column scrolls on the same area of RAM (with first the main RAM and then the auxiliary RAM switched in). While some ROM routines have been revised to improve support for double high-resolution graphics, Applesoft BASIC has not been upgraded. As a result, there is no easy way for BASIC programmers to take advantage of the double hi-res mode. "Chrome" changes all that.

Chrome is a utility which extends Applesoft. It adds several new commands and revises some old ones to allow full exploitation of the double hi-res graphics mode of the 128K IIe, IIc, and IIGS. It can be used with either color or monochrome monitors to support screen displays of 140 × 192 pixels in 16 colors or 560 × 192 pixels in monochrome.

Chrome does not support the new medium-resolution graphics mode (80 × 48 pixels in 16 colors). However, that mode can be emulated in double hi-res.

Creating The Startup Disk

The simplest way to use Chrome is to dedicate a disk to it. Format a fresh disk under DOS 3.3. Next, type in the following short program:

```
10 PRINT CHR$ (4)"PR#3"
15 PRINT "DOUBLE HI-RES EXTENDE
   D APPLESOFT BEING LOADED."
20 PRINT CHR$ (4)"BRUN CHROME"
30 NEW
```

To have this setup program run automatically when the disk is booted, save the program to the disk with the name HELLO.

Since Chrome is written entirely in machine language, it should be entered with the Apple version of the "MLX" machine language entry program, found elsewhere in this issue. Be sure you read and understand the instructions for using MLX before you begin entering data. When you run MLX, you'll be asked for a starting address and an ending address. For Chrome (Program 1), give the following values:

```
STARTING ADDRESS?    1800
ENDING ADDRESS?      1FA7
```

When you finish typing in all the data from Program 1, be sure to save a copy before leaving MLX. When saving, use the filename CHROME, since that is the name the setup program above looks for. Be sure to save the final copy of the program on the same disk as the setup program.

To use Chrome, turn off your computer, place in the drive the disk containing the setup HELLO program and the CHROME file, and turn on the system. With this disk booted, you'll be provided with a new version of Applesoft with three distinguishing characteristics: First, only 21.5K of RAM is available for your programs. Second, the 80-column text screen is now the default display. And third, several new commands and enhancements are available. Only a Control-Reset will disengage the new BASIC and return you to standard Applesoft.

If at any time something goes wrong with BASIC, the recover procedure is to press Control-Reset as usual, followed by CALL 6144 to reinitialize Chrome's BASIC extension and its new tokens. Neither the reset nor the call performs a NEW, so the residing BASIC program is still there afterwards.

If you feel adventuresome, skip ahead to the description of the commands and start trying some of them. If you would rather start with a demo program, type in, save, and run Program 2. If you have a color

monitor, switch it to monochrome if such a switch is available. The first part of the demo uses monochrome graphics. It draws ten circles, a line, and a string of box-shaped characters. Switch your monitor back to color and press Return. You'll see a demonstration of the color mode. It draws 16 lines in different colors and then a blue oval. Press Return again to exit the demo.

Using the New Commands

If you've examined Program 2, you have already seen some of the new commands. Let's take a close look at each of them.

HGR 3 sets up a double hi-res screen just as HGR sets up normal hi-res, including the four-line text window at the bottom of the screen. To switch back to 40 columns from within a running program you must give DOS a PR#0 command, such as:

```
PRINT CHR$(4) "PR#0" : HOME : HGR2
```

If DOS is not informed of the change, the system will do strange things. Actually, this makes perfect sense—when you type PR# from the direct mode, it is not only executed as an Applesoft command but also as a DOS 3.3 command.

HGR 4 is the same as HGR3, except that it sets up a full graphics screen with no text window at the bottom. Unlike HGR2 (its normal hi-res equivalent), HGR 4 uses memory from hi-res page 1.

DOUBLE puts the HPLOT command in 560-point monochrome mode. In this mode, each plot affects only one bit of screen memory. The particular effect of each plot is determined by the most recent HCOLOR= statement (see the discussion below of HCOLOR=). This mode is most useful in conjunction with a monochrome monitor, where each bit of screen memory corresponds to one screen pixel. On a color monitor, four bits are required to show one colored dot. In DOUBLE mode with a color monitor, changing one bit changes a screen dot's color.

COLOR puts the HPLOT command in 140-point color mode. In this mode, each plot affects four bits of screen memory. The value specified in the most recent COLOR= statement (see below) will be stored in the four bits of screen memory at the position specified by the HPLOT command. This mode is most useful in conjunction with a color monitor, where each four-bit group in screen memory specifies the color of one screen pixel. (Since 16 different colors can be displayed, four bits are required to select the color. Four bits can hold values in the range 0–15.) With a monochrome monitor, each bit of screen memory corresponds to one screen pixel, so in COLOR mode with a monochrome monitor each plot affects four screen dots.

HCOLOR= specifies the way that the HPLOT and ELLIPSE commands affect screen memory bits (and hence screen pixels) in the DOUBLE (560-point monochrome) mode. The syntax for the statement is

HCOLOR=*n*

where *n* can have one of the following values:

Value	Effect of plot
0 or 4	Set bit
1 or 5	Clear bit
2 or 6	Invert (toggle) bit
3 or 7	Read bit and store its value (0 or 1) in location 6

For a monochrome screen, setting a bit will make the corresponding pixel appear white, while clearing a bit will make the corresponding pixel black. After a bit is read, a PEEK(6) can be used to determine the current state of the corresponding pixel. HCOLOR=0 is the default setting after an HGR 3 or HGR 4 statement.

COLOR= selects the color to be used when HPLOT or ELLIPSE draws points in COLOR mode. The syntax for the statement is

COLOR=*n*

where *n* is a value in the range 0–15. The correspondence between numbers and colors is as follows:

0	black	8	dark blue
1	magenta	9	purple
2	brown	10	gray 2
3	orange	11	pink
4	dark green	12	medium blue
5	gray 1	13	light blue
6	green	14	aqua
7	yellow	15	white

Chrome provides no facility to directly read the color of a pixel in COLOR mode. However, color values can be determined by switching to DOUBLE mode, specifying HCOLOR=3, and performing four successive reads with HPLOT to collect the four bits of the color value.

HPLOT uses the same syntax as the standard Applesoft version:

HPLOT *x1,y1* [TO *x2,y2* [TO *x3,y3* ...]]

However, the range of allowable values for the horizontal *(x)* coordinates is extended to 0–559.

ELLIPSE can be used to draw ellipses, circles, and similar shapes. It can also be used to draw arcs and other curved lines. The syntax for the statement is

ELLIPSE *x radius, y radius, quadrant* [AT *x center, y center*]

The AT part is optional. The *x center* and *y center* values specify the center of the figure. If omitted, the ellipse will be centered at the current pixel cursor position. This allows faster recursive ellipse drawing around a central point.

The *x radius* and *y radius* values are not checked for validity, but an ellipse can go off the screen without problems if the values are not unreasonably large.

Some other drawing utilities allow you to specify the starting angle and ending angle when drawing portions of ellipses or circles. For simplicity, ELLIPSE only allows you to choose whether or not to draw each of the four quadrants of the figure.

You specify which quadrants to draw using the *quadrant* value in the ELLIPSE statement. The following table shows which *quadrant* values draw corresponding quadrants of the figure:

upper left	8
upper right	4
lower left	2
lower right	1

The values are cumulative. For example, a *quadrant* value of 15 (8 + 4 + 2 + 1) is required to draw a complete ellipse. A value of 9 (8 + 1) would result in only the upper left and lower right quadrants being drawn.

Bitmapped Images

Chrome also provides a way to place bitmapped images onto the double hi-res screen. This method of shape generation is much different than the shape capabilities built into the computer. Bitmapped shape tables are easier to design and can be drawn more rapidly by the system. Shape rotations are

performed by switching between bitmaps rather than by extra computation when the shape is drawn. You may have to experiment a while before you feel comfortable with the next commands. Beginners may wish to ignore the SHAPE command.

The new command has the following syntax:

SHAPE *width, height, location* [AT *x, y*]

The AT portion of the statement is optional. If it is included, the *x* and *y* values specify the coordinates at which the upper left corner of the shape will be drawn.

The *width* value is added to the *x* coordinate after each SHAPE execution. You can use this value to specify the horizontal spacing of shapes on the screen.

The *height* value specifies the height of the bitmapped image in screen lines. All bitmaps are two bytes (14 screen pixels) wide, so the height is half the length of the bitmapped shape table.

The *location* value specifies the starting memory location for the bitmap shape table in RAM. More information on Chrome's memory usage is provided later in this article.

If the *x* coordinate for the shape does not begin on an exact byte boundary (pixel position 0, 7, 14, 21, and so on), the data from the bitmap must be preshifted before drawing. Preshifting does not retain the shifted bits. For instance if a full 14-bit-wide shape is to be plotted at coordinates 55,10, then six pixels will be lost during shifting because the preshifter must move the shape over 6 bits (55 MOD 7 = 6). Only shapes less than 8 bits wide can be drawn anywhere horizontally without risk of being partly lost. All bitmap data is in the same format as screen RAM, with bit 7 unused.

The drawing routine simply takes the data from a buffer and places it on the screen in exclusive-OR mode. Shapes are drawn the same whether DOUBLE or COLOR plotting is enabled.

Memory Usage

Machine language programmers may appreciate the new commands as much as BASIC programmers will. Following is a description of how memory is allotted when Chrome is being used.

The ellipse routine keeps most of its data in the lower 16 bytes of the auxiliary zero page. Main-memory zero-page use is limited to those locations also used by the standard hi-res routines.

The program itself begins at location $1800 hex (6144 decimal) and ends at $1FFF. Screen RAM is at $2000–$3FFF.

When the program is BRUN (loaded and run) it initializes itself in five steps. First, it copies the high ROM ($D000 to $FFFF), including Applesoft, into the main memory RAM and alters Applesoft to accommodate the new commands.

Once the copy-and-change phase is done, four tables are created. Two reside in main-memory high RAM and are switched in when needed. They hold video addresses to speed up the pixel plotter. The third table, located at $1600, is the new tokenizer table; it holds the ASCII keywords of all of the old commands as well as those of the new commands. The fourth table, located at $1400, is used by the ELLIPSE routine only and holds the 16-bit squares of 0–255. These values are also used to speed up pixel position calculations.

The area from $0800 to $13FF is free for machine language routines and bitmaps of character sets.

Program 1: Chrome

For instructions on entering this program, please refer to the "Apple MLX" article elsewhere in this issue.

```
1800: A0 00 84 60 A9 D0 85 61 14
1808: 8D 89 C0 2C 89 C0 2C 89 6D
1810: C0 B1 60 91 60 C8 D0 F9 F3
1818: E6 61 D0 F5 A0 00 B9 2A 30
1820: 19 99 8F 17 F0 03 C8 D0 9C
1828: F5 A0 00 B9 D0 D0 99 00 14
1830: 16 C8 D0 F7 A0 8F B9 CF BD
1838: D1 99 FF 16 88 D0 F7 A9 3A
1840: 00 8D 93 D5 8D 3D D7 A9 5E
1848: 15 8D 97 D5 8D 41 D7 A9 81
1850: 4C 8D 2E D8 A9 42 8D 2F FD
1858: D8 A9 19 8D 30 D8 A9 5C EF
1860: 8D 26 D0 A9 19 8D 27 D0 B3
1868: A9 67 8D 22 D0 A9 19 8D 08
1870: 23 D0 A9 7E 8D 20 D0 A9 BB
1878: 19 8D 21 D0 A0 00 84 67 3F
1880: AD 97 19 18 69 20 85 68 51
1888: 98 91 67 E6 67 8D 80 C0 F7
1890: AD 81 C0 AD 81 C0 A9 00 4D
1898: 85 E6 AA A8 8D B0 18 8D 12
18A0: B2 18 8D B6 18 A9 D0 8D E3
18A8: B3 18 A9 D4 8D B7 18 A9 60
18B0: 6D 99 00 D3 8A 99 00 D7 CD
18B8: E8 E0 07 D0 05 EE B0 18 E0
18C0: A2 00 C8 D0 EA EE B3 18 FA
18C8: EE B7 18 AD B3 18 C9 D3 A1
18D0: D0 DD A0 00 98 48 A2 00 20
18D8: A0 00 20 11 F4 68 A8 A5 AE
18E0: 26 99 00 D8 A5 27 99 00 15
18E8: D9 C8 C0 C0 90 E6 A9 08 D7
18F0: 8D 7F 1D A9 00 8D 09 C0 0F
18F8: 48 85 00 85 03 A9 00 85 4B
1900: 01 85 04 A2 06 A0 03 20 97
1908: 62 1D 68 AA A5 06 A4 07 FF
1910: 8D 08 C0 9D 00 15 98 9D 20
1918: 00 14 E8 8A D0 D7 8D 88 9E
1920: C0 60 C7 19 CD 19 0F 1D 63
1928: 3E 1F 43 4F 4C 4F D2 44 28
1930: 4F 55 42 4C C5 53 48 41 B9
1938: 50 C5 45 4C 4C 49 50 53 EC
1940: C5 00 C9 40 B0 03 4C 32 EE
1948: D8 E9 6B B0 03 4C C9 DE 95
1950: 0A A8 B9 23 19 48 B9 22 9A
1958: 19 48 4C B1 00 2C 92 19 BC
1960: 30 03 4C FE F6 4C D4 19 90
1968: C9 34 D0 03 4C 04 1A C9 47
1970: 33 D0 03 4C 0A 1A 4E 92 7D
1978: 19 20 88 19 4C E2 F3 4E 06
1980: 92 19 20 88 19 4C D8 F3 6E
1988: AD 54 C0 AD 5F C0 8D 0C BE
1990: C0 60 80 00 00 00 00 20 6B
1998: 00 00 01 02 04 08 10 20 8B
19A0: 40 20 67 DD 20 52 E7 A5 85
19A8: 51 A6 50 C9 02 90 06 D0 03
19B0: 50 E0 30 B0 4C 48 8A 48 35
19B8: A9 2C 20 C0 DE 20 F8 E6 2B
19C0: E0 C0 B0 3D 68 A8 68 60 94
19C8: A9 80 8D 98 19 60 A9 00 C8
19D0: 8D 98 19 60 C9 C1 F0 13 63
19D8: 20 A1 19 86 E2 84 E0 85 7F
19E0: E1 20 4E 1A 20 B7 00 C9 21
19E8: C1 D0 12 20 C0 DE 20 A1 D7
19F0: 19 86 E3 84 E6 85 E7 20 53
19F8: F3 1A 4C E4 19 8D 54 C0 EB
1A00: 60 4C 06 F2 AD 52 C0 4C EB
1A08: 0D 1A AD 53 C0 AD 50 C0 52
1A10: AD 57 C0 AD 5E C0 8D 01 F5
1A18: C0 8D 0D C0 A9 00 85 E0 F6
1A20: 85 E1 85 E2 85 E4 A8 AE 2E
1A28: 97 19 86 27 84 26 A2 1F D2
1A30: 8D 54 C0 91 26 C8 D0 FB 63
1A38: 8D 55 C0 91 26 C8 D0 FB AB
1A40: E6 27 CA 10 EB 8E 92 19 E3
1A48: 8D 54 C0 4C B1 00 2C 98 B3
1A50: 19 10 19 A5 E0 48 A5 E1 E7
1A58: 48 46 E1 66 E0 46 E1 66 2F
1A60: E0 20 00 1C 68 85 E1 68 54
1A68: 85 E0 60 60 A5 E4 4A 29 28
1A70: 03 8D 99 19 2C 98 19 10 54
1A78: 03 4C 00 1C A6 E2 E0 C0 46
1A80: F0 E9 A5 E1 C9 02 90 0A FB
1A88: F0 02 B0 DF A5 E0 C9 30 3E
1A90: B0 D9 8D 80 C0 BD 00 D8 23
1A98: 85 26 BD 00 D9 0D 97 19 1C
1AA0: 85 27 A4 E0 A5 E1 09 D0 9B
1AA8: 8D B2 1A 09 D4 8D B5 1A 86
1AB0: B9 00 D0 BE 00 D4 8D 88 BE
1AB8: C0 2C 93 19 10 01 60 4A EB
1AC0: A0 01 90 01 88 99 54 C0 BB
1AC8: A8 BD 9A 19 AE 99 19 F0 A4
1AD0: 10 E0 02 90 11 F0 16 A2 A9
1AD8: 01 31 26 D0 01 CA 86 06 F1
1AE0: 60 11 26 91 26 60 49 7F 2C
1AE8: 31 26 91 26 60 51 26 91 F9
1AF0: 26 60 00 2C 98 19 10 16 72
1AF8: 46 E1 66 E0 46 E1 66 E0 0B
1B00: 46 E7 66 E6 46 E7 66 E6 14
1B08: A9 1B 48 A9 F6 48 A2 FF 9B
1B10: 8E F2 1A E8 A0 02 A5 E6 5B
1B18: 38 E5 E0 85 D0 A5 E7 E5 2B
1B20: E1 B0 12 49 FF AA A5 D0 11
1B28: 49 FF 69 01 90 01 E8 85 20
1B30: D0 8A A2 01 38 85 D1 A5 F6
1B38: E3 E5 E2 B0 06 49 FF 69 FF
1B40: 01 A0 03 85 D2 A5 D0 C5 6C
1B48: D2 A5 D1 E9 00 B0 14 86 9B
1B50: D3 84 1A A6 1A A5 D0 A4 EC
1B58: D2 85 D2 84 D0 A9 00 85 AE
1B60: D1 A4 D3 A5 D1 85 1C 4A A4
1B68: 85 D5 A5 D0 85 1B 6A 85 8B
1B70: D4 BD EF 1B 8D 96 1B B9 E6
1B78: F3 1B 8D D4 1B 2C F2 1A F7
1B80: 30 03 20 6C 1A 4E F2 1A 64
1B88: A5 1B 38 E9 01 85 1B A5 F7
```

Increase your knowledge about all aspects of computers

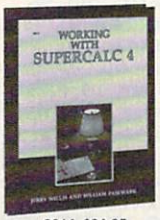
An absolutely no-risk guarantee.

Select 5 Books for only $3.95

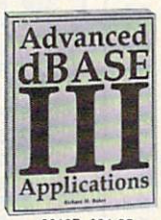
Values to $127.75

More programs, projects, and ways to use your micro. Keep well-informed about the latest books available—and get them at discounts of up to 50% off the publishers' prices!

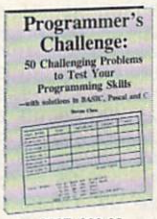
2771 $25.95



2814 $24.95



2618P $21.95



2837 $29.95
Counts as 2

1085P $10.25

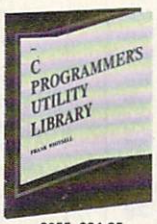
1160P $10.95

2754 $28.95
Counts as 2

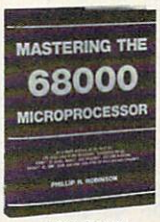
2831 $22.95

2892P $16.95

1295P $10.25

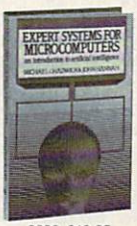


2855 $24.95



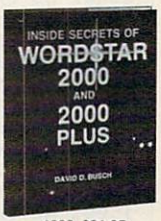
1886 $22.95

2688 $26.95
Counts as 2



2838 $19.95

1251P $10.25



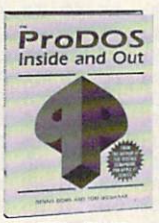
1993 $21.95

2852 $25.95

1970 $22.95

2828P $18.95

1718P $15.50

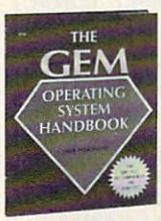


2745 $24.95

2627P $17.95

2862 $19.95

2748 $21.95

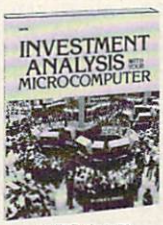


2742 $23.95

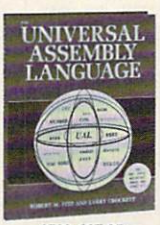
1873P $17.95

1997 $21.95

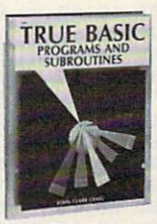
2798 $26.95
Counts as 2



1479P $13.50



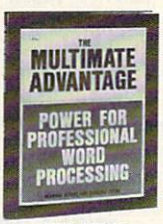
2730 $27.95
Counts as 2



1990 $24.95

2791 $21.95

2692 $27.95
Counts as 2

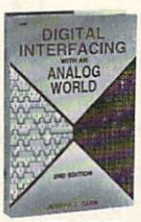


2793 $22.95

2654 $18.95

2840 $24.95

2872 $19.95



2850 $25.95

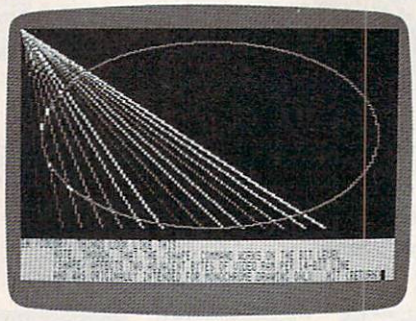
All books are hardcover editions unless numbers are followed by a P for paperback.

(Publishers' Prices Shown)

If card is missing, use this address to join: THE COMPUTER BOOK CLUB®
P.O. Box 80, Dept. CMPT-1087, Blue Ridge Summit, PA 17214

The Computer Book Club®

Membership Benefits • **Big Savings.** In addition to this introductory offer, you keep saving substantially with members' prices of up to 50% off the publisher's list price. • **Bonus Books.** Starting immediately, you will be eligible for our Bonus Book Plan, with savings of up to 80% off publishers' prices. • **Club News Bulletins.** 14 times per year you will receive the Book Club News, describing all the current selections—mains, alternates, extras—plus bonus offers and special sales, with hundreds of titles to choose from. • **Automatic Order.** If you want the Main Selection, do nothing and it will be sent to you automatically. If you prefer another selection, or no book at all, simply indicate your choice on the reply form provided. As a member, you agree to purchase at least 3 books within the next 12 months and may resign at any time thereafter. • **Ironclad No-Risk Guarantee.** If not satisfied with your books, return them within 10 days without obligation! • **Exceptional Quality.** All books are especially selected by our Editorial Board.



Add a wealth of double hi-res graphics commands to Applesoft BASIC with this superb utility.

```
1B90: 1C E9 00 85 1C B0 FE 60 A9
1B98: E6 E0 D0 14 E6 E1 B0 10 06
1BA0: A5 E0 D0 02 C6 E1 C6 E0 48
1BA8: B0 06 C6 E2 B0 02 E6 E2 FD
1BB0: A5 D4 18 65 D2 85 D4 90 2F
1BB8: 02 E6 D5 A5 D0 C5 D4 A5 AB
1BC0: D1 E5 D5 B0 B8 A5 D4 38 5D
1BC8: E5 D0 85 D4 A5 D5 E5 D1 46
1BD0: 85 D5 38 B0 FE E6 E0 D0 77
1BD8: A4 E6 E1 B0 A0 A5 E0 D0 90
1BE0: 02 C6 E1 C6 E0 B0 96 C6 30
1BE8: E2 B0 92 E6 E2 B0 8E 01 75
1BF0: 09 17 13 00 08 16 12 06 96
1BF8: E0 26 E1 06 E0 26 E1 60 89
1C00: A5 E0 48 A5 E1 48 06 E0 C3
1C08: 26 E1 06 E0 26 E1 A5 30 CE
1C10: 85 E8 A9 04 85 E9 06 E8 83
1C18: A2 01 90 01 CA 8E 99 19 E0
1C20: 20 7C 1A E6 E0 D0 02 E6 6E
1C28: E1 C6 E9 D0 E9 68 85 E1 2B
1C30: 68 85 E0 60 00 00 00 00 20
1C38: 00 00 00 00 00 00 00 00 70
1C40: 00 00 00 00 00 00 00 00 78
1C48: 00 00 00 00 00 00 00 00 80
1C50: 00 00 00 00 00 00 00 00 88
1C58: 00 00 00 00 00 00 00 00 90
1C60: 00 00 00 00 00 00 00 00 98
1C68: 00 00 00 00 00 00 00 00 A0
1C70: 00 00 00 00 8E 02 1D 0A 69
1C78: 8D FB 1C A9 80 8D 93 19 0F
1C80: 20 7C 1A 8D D7 1C 8E C5 16
1C88: 1C 4E 93 19 A5 E2 48 8D 3D
1C90: AC 1C A0 55 A2 54 4E D7 6A
1C98: 1C 90 02 88 E8 8E ED 1C 45
1CA0: 8C D9 1C A2 00 8E E4 1C 63
1CA8: 6E E4 1C A0 00 C0 C0 F0 54
1CB0: 4D 8D 80 C0 B9 00 D8 85 14
1CB8: 26 B9 00 D9 0D 97 19 85 8E
1CC0: 27 8D 88 C0 A0 00 F0 0E 02
1CC8: 1E 34 1C 10 02 38 90 18 CB
1CD0: 3E 35 1C 88 D0 F2 A0 00 15
1CD8: 8D 54 C0 B1 26 5D 34 1C 4B
1CE0: 91 26 E8 A9 00 10 05 C8 36
1CE8: C0 28 F0 0A 8D 54 C0 B1 3B
1CF0: 26 5D 34 1C 91 26 E8 EE C1
1CF8: AC 1C E0 10 90 AD A5 E0 13
1D00: 18 69 0E 85 E0 90 02 E6 EE
1D08: E1 68 85 E2 8D 54 C0 60 CB
1D10: 20 F8 E6 86 60 A9 2C 20 FF
1D18: C0 DE 20 F8 E6 86 61 A9 BB
1D20: 2C 20 C0 DE 20 67 DD 20 F8
1D28: 52 E7 A5 51 85 63 A5 50 A4
1D30: 85 62 20 B7 00 C9 C5 D0 C8
1D38: 0C 20 C0 DE 20 A1 19 86 C6
1D40: E2 84 E0 85 E1 A5 61 0A F3
1D48: A8 88 30 08 B1 62 99 34 FD
1D50: 1C 4C 49 1D A5 61 A6 60 07
1D58: 4C 74 1C 00 00 00 00 00 59
1D60: 00 00 A5 00 85 18 A5 01 28
1D68: 85 19 B9 00 00 85 15 B9 DC
1D70: 01 00 85 16 A9 00 85 17 AC
1D78: 95 00 95 01 95 02 A0 08 3E
1D80: 46 19 66 18 90 13 18 A5 19
1D88: 15 75 00 95 00 A5 16 75 3C
1D90: 01 95 01 A5 17 75 02 95 53
1D98: 02 06 15 26 16 26 17 88 5A
1DA0: D0 DE 60 8D 61 1D 78 8D DD
1DA8: 09 C0 AD 60 1D 85 1B A9 32
1DB0: 00 8D 5D 1D 85 08 85 05 28
1DB8: 85 02 85 0F 85 10 85 11 60
1DC0: AE 5E 1D 86 00 8E 5B 1D 04
1DC8: AD 5F 1D 85 01 8D 5C 1D C1
1DD0: 09 08 8D 7F 1D C9 08 D0 2C
1DD8: 0D BD 00 15 85 03 BD 00 0E
1DE0: 14 85 04 4C ED 1D A0 00 F0
1DE8: A2 03 20 62 1D A6 1B BD D6
1DF0: 00 15 85 06 BD 00 14 85 1D
1DF8: 07 A0 06 A2 09 20 62 1D 74
1E00: 18 A5 03 65 06 85 0C A5 6C
1E08: 04 65 07 85 0D A5 05 69 4B
1E10: 00 4A 85 0E 66 0D 66 0C B0
1E18: 46 0E 66 0D 66 0C 38 A5 12
1E20: 0C E5 09 85 0C A5 0D E5 4C
1E28: 0A 85 0D A5 0E E5 0B 85 6A
1E30: 0E 06 03 26 04 26 05 06 80
1E38: 06 26 07 26 08 06 09 26 D4
1E40: 0A 26 0B A5 00 18 65 1B 0D
1E48: 85 12 A5 01 69 00 85 13 F9
1E50: 8D 08 C0 20 D0 1E 8D 09 92
1E58: C0 A5 0E 10 2C EE 5D 1D 16
1E60: 18 A5 0F 65 03 85 0F A5 3C
1E68: 10 65 04 85 10 A5 11 65 7D
1E70: 05 85 11 18 A5 0C 65 0F 6B
1E78: 85 0C A5 0D 65 10 85 0D 83
1E80: A5 0E 65 11 85 0E 4C BA 88
1E88: 1E AD 5B 1D D0 03 CE 5C 09
1E90: 1D CE 5B 1D 38 A5 09 E5 9C
1E98: 06 85 09 A5 0A E5 07 85 30
1EA0: 0A A5 0B E5 08 85 0B 38 AF
1EA8: A5 0C E5 09 85 0C A5 0D BC
1EB0: E5 0A 85 0D A5 0E E5 0B 20
1EB8: 85 0E A5 12 38 E9 01 85 02
1EC0: 12 A5 13 E9 00 85 13 90 3D
1EC8: 03 4C 50 1E 8D 08 C0 60 F3
1ED0: A5 E0 85 E6 18 6D 5B 1D 81
1ED8: 85 E0 A5 E1 85 E7 6D 5C E5
1EE0: 1D 85 E1 A5 E2 85 E3 18 B0
1EE8: 6D 5D 1D 85 E2 48 A5 E3 96
1EF0: 38 ED 5D 1D 48 AD 61 1D 1B
1EF8: 29 01 F0 03 20 6C 1A 68 A7
1F00: 85 E2 AD 61 1D 29 04 F0 0C
1F08: 03 20 6C 1A A5 E6 38 ED 26
1F10: 5B 1D 85 E0 A5 E7 ED 5C 07
1F18: 1D 85 E1 AD 61 1D 29 08 37
1F20: F0 03 20 6C 1A 68 85 E2 C2
1F28: AD 61 1D 29 02 F0 03 20 C5
1F30: 6C 1A A5 E3 85 E2 A5 E7 09
1F38: 85 E1 A5 E6 85 E0 60 20 65
1F40: 67 DD 20 52 E7 A5 51 8D D8
1F48: 5F 1D A5 50 8D 5E 1D A9 01
1F50: 2C 20 C0 DE 20 F8 E6 8E F3
1F58: 60 1D A9 2C 20 C0 DE 20 E7
1F60: F8 E6 8A 48 20 B7 00 C9 54
1F68: C5 D0 0C 20 C0 DE 20 A1 A4
1F70: 19 86 E2 84 E0 85 E1 2C 8E
1F78: 98 19 10 0F A0 02 4E 5F 45
1F80: 1D 6E 5E 1D 46 E1 66 E0 ED
1F88: 88 D0 F3 68 20 A3 1D 2C 3A
1F90: 98 19 10 08 06 E0 26 E1 C5
1F98: 06 E0 26 E1 8D 54 C0 60 94
1FA0: A0 D3 D4 00 00 00 00 00 BE
```

Program 2: Chrome Demonstration

For instructions on entering this program, please refer to "COMPUTE!'s Guide to Typing In Programs" elsewhere in this issue.

```
51 5 HOME
94 6 REM COPYRIGHT 1987 COMPUTE!
     PUBLICATIONS, INC.
62 7 REM ALL RIGHTS RESERVED.
D6 8 PRINT "COPYRIGHT 1987": PRI
     NT "COMPUTE! PUBLICATIONS,
     INC.": PRINT "ALL RIGHTS RE
     SERVED."
14 9 FOR X = 0 TO 1200: NEXT : F
     OR X = 0 TO 15: READ A: POK
     E X + 768,A: NEXT
18 10 HGR 3
71 20 DOUBLE
CD 30 HPLOT 0,0 TO 559,159
9A 40 HCOLOR= 3: HPLOT 0,0: HCOL
     OR= 0
8C 50 A = 768
66 60 FOR S = 1 TO 20
91 70 SHAPE 8,8,A
AB 80 NEXT
93 100 DATA 127,0,65,0,65,0,65,0
     ,65,0,65,0,127,0,0
8F 105 DATA 0
7C 200 REM  DRAW CIRCLES
E2 210 HPLOT 280,96
0D 220 HCOLOR= 3: HPLOT 280,96:
     HCOLOR= 0: REM SET MIDDLE
BE 225 REM /ELLIPSE/ WITHOUT "AT
      X,Y" IS FASTER
C7 230 FOR X = 1 TO 90 STEP 8
D0 240 ELLIPSE X * 2,X,15
06 260 NEXT
A2 300 HOME : VTAB 21
1C 310 PRINT "THIS IS A SIMPLE D
     EMO OF THE NEW GRAPHICS C
     OMMANDS
B2 320 PRINT TAB( 10)"AT THE UPP
     ER LEFTHAND CORNER OF THE
      SCREEN IS THE OUTPUT OF
     /SHAPE/
14 330 PRINT TAB( 10)"THE CIRCLE
     S ARE ACTUALLY ELLIPSES,
     PRODUCED BY /ELLIPSE/
F8 340 PRINT TAB( 10)"AND THE LI
     NE IS DRAW BY HPLOT... HI
     T <RETURN>";
D9 350 GET A$
AE 360 HOME : VTAB 21
EF 370 HGR 3
6B 380 PRINT "IN /COLOR/, THINGS
      LOOK LIKE THIS:
FB 390 COLOR
61 400 FOR X = 0 TO 15: COLOR= X
40 410 HPLOT 0,0 TO 30 * X,150
FF 420 NEXT
FC 430 COLOR= 12: REM BLUE
0F 440 ELLIPSE 250,70,15 AT 280,
     80
59 450 PRINT TAB( 8)"NOTE, THOUG
     H, THAT THE /SHAPE/ COMMA
     ND WORKS ON THE BIT LEVEL
     .
9A 460 PRINT TAB( 8)"/SHAPE/ AFF
     ECTS TWO ADJACENT BYTES O
     F VIDEO RAM PER 14-BIT LI
     NE"
E4 465 PRINT TAB( 8)"AND WAS ORI
     GINALLY INTENDED FOR MONO
     CHROME DRAWING ONLY";
72 470 PRINT "    HIT <RETURN>";:
      GET A$
67 480 TEXT : HOME
```

A BUTTON Command
For Amiga Basic

Robert Katz

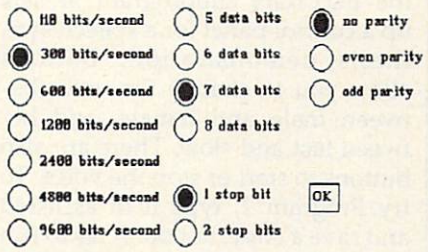
Take advantage of the mouse and icon interface with this subprogram that you can use in your own Amiga Basic programs. A demo that lets you change the speed and gender of the Amiga's voice is included.

Perhaps more than any other BASIC, Amiga Basic is extensible. New commands are easily added to the language. "BUTTON" takes advantage of this flexibility to add a BUTTON command similar to the one in Microsft BASIC for the Macintosh.

Buttons allow a menu-like choice without forcing the user to hold down the right mouse button and perform a menu action. In general, you use a button when you want the choice to be intrusive; you use menus when you want the choice to be available only when the user looks for it.

Buttons are useful when you want several choices to be displayed and alterable at once. As an example, if you were writing a telecommunications terminal program, you would probably want all the options to be displayed on the screen simultaneously. (See the accompanying figure for an example.) All at once, the user can see that the speed of communication varies between 110 and 9600; parity may be even, odd, or not selected; and the data and stop bits may be selected in a given number of ways. All of the current settings are clearly shown.

The BUTTON command is implemented as a subprogram. Subprograms differ from subroutines in that subprograms allow local variables that cannot be accessed by the main program or other subprograms. When you use a subprogram, you don't have to worry about using a variable that is already being used by the main program—no side effects are possible (unless you specify that certain variables are to be shared among program and subprograms). Although the variables are independent, it is possible to pass values (called *parameters*) back and forth between the subprogram and its calling program.



Example: Telecommunications Parameters

A subprogram must begin with a SUB statement containing the name of the subprogram and a list of parameters. The END SUB statement marks the end of a subprogram. All statements between SUB and END SUB are considered part of the subprogram rather than part of the main program.

Program 1 is the subprogram to implement the BUTTON command. Type it in and save a copy to disk, but don't try to run it yet. This isn't a complete program, only a subprogram. Since you will want to merge this with your own programs, be sure to save it in ASCII format. To do this, you must use a command of the form

SAVE *"filename"*,**A**

Calling All Subprograms

There are two ways for the main program to access, or *call*, a subprogram. The formal method is with a statement of the form

CALL ***subprogram name (parameter list)***

However, Amiga Basic also permits you to omit the CALL and the parentheses surrounding the parameter list. This format makes the subprogram call look just like any other BASIC statement, so the subprogram essentially becomes a new BASIC command.

Here is the syntax for the BUTTON command:

BUTTON ***ID, state, title, x position, y position, type***

The *ID* and *state* parameters are two-way values. Setting them before you call the subprogram provides instructions to the subprogram, and reading their values after

the subprogram is called, returns information on the buttons. The other parameters are one-way—they provide information to the subprogram, but nothing is reported back through them. One difference between subprograms and actual commands is that you must supply some value for all the parameters in a subprogram call. None of the parameters are optional, even in cases where the parameter values aren't meaningful.

The *ID* value is a unique number by which you can identify each button. You may have any number of buttons on the screen at once. Allowable button numbers range from 1 up to the maximum number you specify in Program 1.

An *ID* value of 0 has a special meaning. When the subprogram is called with this value in the first parameter position, it checks whether the mouse is currently positioned over any previously defined button. (In this case, the values of the other parameters are not important.) If the mouse is over a button, then the ID value of that button is returned in the *ID* parameter, the state of the button is inverted (from 1 to 2, or vice versa), and the new state of the button is returned in the *state* parameter. If the mouse isn't over any of the buttons, the *ID* parameter will still be 0 upon return.

Note that this function compares the current mouse pointer position against the coordinates for all defined buttons, without regard for whether the buttons are actually being displayed on the screen at the time of the test. To prevent spurious readings, it is important to clear button definitions when the corresponding buttons are erased from the screen.

The *state* parameter determines the state of the button specified by the *ID* parameter. If this parameter is set to 0 when the subprogram is called, then the current definition for the specified button is erased. This makes the button inactive, but does not remove it from the screen. If this parameter is set to 1, then the specified button is deselected (turned off). Setting this parameter to 2 causes the corresponding button to be selected (turned on).

The remaining parameters are only meaningful the first time the subprogram is called for a particular *ID* value. *Title* is the name you assign to a button, *x position* and *y position* define the screen coordinates of the button, and *type* identifies the shape of the button as described below. Once a button is defined, these features cannot be changed.

The subprogram provides three different types of buttons. Type 1 is a rectangle with the name of the button inside it. When a type 1 button is selected, the button name appears in reverse video. Type 2 buttons are squares with their names beside them. When a type 2 button is selected, it is marked with an X. Type 3 buttons are circles with their names beside them. When a type 3 button is selected, a smaller filled circle is placed inside the circle. The figure shows all three types of buttons.

Two preparatory steps are required when using BUTTON in your own programs. First, your main program should begin with the statement DEFINT a–z to force all variables to default to short integer type. Secondly, because the subprogram makes calls to operating system graphics routines, you must place a copy of the graphics.bmap file into the disk directory that holds your program. The graphics.bmap file can be found in the BasicDemo folder on the Extras disk that came with your Amiga.

Buttoned Speech

The best way to learn how to use BUTTON is by looking at an example. Program 2 illustrates all three types of buttons provided by the BUTTON subprogram. It sets up a control panel for a speech synthesis demonstration. Buttons allow you to change the voice between male and female, and between fast and slow. There are also buttons to start or stop the voice. To try Program 2, type it in as listed and save a copy, but don't try to run it yet. First you must merge in the button subprogram. Use a command of the form

MERGE *"filename"*

where *filename* is the name you used for the ASCII file containing the subprogram. After the subprogram is merged in, the combined program is ready to run. If you want, you can save a copy of the completed program to avoid having to merge the subprogram again.

The first BUTTON command in Program 2, in the FOR-NEXT loop, provides the initial definitions of the six buttons. The second BUTTON command, following the WEND, has the *ID* parameter set to 0 to check whether the mouse pointer was over a button when the mouse button was clicked. If so, the variable ID will have a nonzero value upon return, and array element State(ID) will hold the new state of that variable. (Although Program 2 uses a variable named ID for the *ID* parameter, that's not mandatory. Any other variable name would work just as well.) The final BUTTON command is used to set the other button in the panel pair to the complementary state. Note that dummy values are supplied for the last three parameters in the latter two BUTTON commands. These values are necessary to satisfy the syntax of the subprogram call.

Since Program 2 defines only six buttons, the default value for the maxbut variable in the subprogram is sufficient. Should your application require more than ten buttons, simply increase maxbut to whatever value you need.

After examining Program 2, try using the BUTTON subprogram in your own programs. Remember that you must place a copy of the graphics.bmap file into the disk folder that holds your BASIC programs for the program to function properly.

Technical Details

Because there is no easy way for Amiga Basic to print text at an arbitrary pixel location, the BUTTON subprogram uses an Amiga operating system graphic routine called Move. Now, characters can be placed outside of normal character boundaries. The Move routine is called like an Amiga Basic subprogram. The routine has three parameters. The first is the address of the current window's RastPort, a structure used by the operating system. We can find this address with the WINDOW(8) function. The other two parameters are the X and Y locations (in pixels) that specify

where the text should begin. Here's an example of a call to Move:

CALL Move& (WINDOW(8), X%, Y%)

Note that X and Y are long integers (as the suffix % shows).

One other system routine is used in BUTTON. This routine, SetDrMd, changes the default drawing mode. This is used to invert the colors when a type 1 button is selected. From BASIC, it's called like this:

CALL SetDrMd&(WINDOW(8),*value*)

Value represents one of the four available drawing modes. The following list shows the modes and their corresponding values:

Mode	Value
Jam1	0
Jam2	1
Complement	2
Inversvid	3

The default mode is Jam2. It "jams" the foreground and background colors (as an 8 x 8 rectangle) onto the screen memory. Jam1 jams only the foreground color onto the screen, allowing you to overlay characters. Complement reverses the colors, changing all ones to zeros and all zeros to ones. For instance, if you are using a four-color screen, color 0 would switch places with color 3, and color 1 would switch places with color 2 (it makes more sense in binary). Inversvid exchanges the foreground color with the background color to make inverse characters.

For instructions on entering these programs, please refer to "COMPUTE!'s Guide to Typing In Programs" elsewhere in this issue.

Program 1. BUTTON

```
'BUTTON  subprogram
' Copyright 1987, COMPUTE! Publi
cations, Inc.
' ALL rights reserved.
 
SUB BUTTON(ID,State,TitLe$,X,Y,T
ype) STATIC
IF ID=0 THEN GOTO CheckButton
IF numbut>0 THEN GOTO Skipped
LIBRARY "graphics.library"
rastport&=WINDOW(8)
CALL SetDrMd&(rastport&,0)
maxbut=10
DIM ID(maxbut),State(maxbut),Tit
Le$(maxbut),Type(maxbut)
DIM X1(maxbut),Y1(maxbut),X2(max
but),Y2(maxbut)
AREA(10,10):AREAFILL
 
Skipped:
IF (ID<0 OR ID>maxbut OR State<0
OR State>2 OR Type>3) THEN EXIT
SUB
IF State=0 THEN GOTO EraseButton
 
IF ID(ID)<>ID THEN
ID(ID)=ID
IF ID>numbut THEN numbut=ID
TitLe$(ID)=" "+TitLe$+" "
X1(ID)=X:Y1(ID)=Y
Type(ID)=Type
END IF
IF Type(ID)<1 THEN GOTO EraseBut
ton
State(ID)=State
ON Type(ID) GOTO One, Two, Three
 
 
One:
X2(ID)=X1(ID)+LEN(TitLe$(ID))*10
+1:Y2(ID)=Y1(ID)+11
LINE(X1(ID),Y1(ID))-(X2(ID),Y2(I
D)),0,bf
LINE(X1(ID),Y1(ID))-(X2(ID),Y2(I
D)),1,b
IF State=2 THEN
LINE(X1(ID),Y1(ID))-(X2(ID),Y2(I
D)),1,bf
CALL SetDrMd&(rastport&,5)
END IF
CALL Move&(rastport&,X1(ID)+1,Y1
(ID)+8)
PRINT TitLe$(ID):CALL SetDrMd&(r
astport&,1)
EXIT SUB
 
Two:
X2(ID)=X1(ID)+20:Y2(ID)=Y1(ID)+1
0
LINE (X1(ID),Y1(ID))-(X2(ID),Y2(
ID)),1+(State=1)
LINE (X1(ID),Y2(ID))-(X2(ID),Y1(
ID)),1+(State=1)
LINE (X1(ID),Y1(ID))-(X2(ID),Y2(
ID)),1,b
CALL Move&(rastport&,X1(ID)+23,Y
1(ID)+8)
PRINT TitLe$(ID)
EXIT SUB
 
Three:
X2(ID)=X1(ID)+12:Y2(ID)=Y1(ID)+1
2
Xc=X1(ID)+5:Yc=Y1(ID)+5
CIRCLE (Xc,Yc),12
IF State=2 THEN CIRCLE (Xc,Yc),6
 
PAINT (Xc,Yc),1+(State=1)
CALL Move&(rastport&,X1(ID)+19,Y
1(ID)+8)
PRINT TitLe$(ID)
EXIT SUB
 
CheckButton:
Xm=MOUSE(1):Ym=MOUSE(2)
Checkloop:
ID=ID+1
IF ID>numbut THEN
ID=0
EXIT SUB
END IF
IF ID(ID)=0 THEN GOTO Checkloop
IF (Xm>X1(ID) AND Xm<X2(ID) AND
Ym>Y1(ID) AND Ym<Y2(ID)) THEN
State=1-(State(ID)=1)
GOTO Skipped
END IF
GOTO Checkloop
 
EraseButton:
ID(ID)=0:State(ID)=0
TitLe$(ID)=""
X1(ID)=0:Y1(ID)=0
Type(ID)=0
EXIT SUB
 
END SUB
 
 
 
```

Program 2. A BUTTON Demonstration

```
'BUTTON Demo
' Copyright 1987, COMPUTE! Publi
cations, Inc.
' ALL rights reserved.
 
DEFINT a-z
DIM voice(8),word$(9),State(6),T
itLe$(6),X(6),Y(6),Type(6)
FOR i=0 TO 8:READ voice(i):NEXT
DATA 110,0,150,1,22200,64,10,0,0
 
FOR i=1 TO 9:READ word$(i):NEXT
DATA one,two,three,four,five,six
,seven,eight,nine
LOCATE 1,10:PRINT "Copyright 198
7, COMPUTE! Publications, Inc."
LOCATE 2,21:PRINT "All rights re
served."
'Initialize buttons
FOR ID=1 TO 6
READ State(ID),TitLe$(ID),X(ID),
Y(ID),Type(ID)
BUTTON ID,State(ID),TitLe$(ID),X
(ID),Y(ID),Type(ID)
NEXT
DATA 1,Male,150,25,1
DATA 2,Female,300,25,1
DATA 1,Normal,150,50,2
DATA 2,Fast,300,50,2
DATA 1,Start,155,75,3
DATA 2,Stop,305,75,3
'Initialize voice parameters
GOSUB Gender
GOSUB Speed
GOSUB Status
count=1
 
here:
WHILE MOUSE(0)<1
IF speak THEN
SAY TRANSLATE$(word$(count)),voi
ce
count=count+1
IF count>9 THEN count=1
END IF
WEND:ID=0
BUTTON ID,State,"",0,0,0
IF ID=0 THEN GOTO here 'mouse pr
essed when not on a button
State(ID)=State
'Reverse state of complementary
button
IF (ID MOD 2) THEN ID2=ID+1 ELSE
ID2=ID-1
State(ID2)=1-(State=1)
BUTTON ID2,State(ID2),"",0,0,0
'Change operating conditions acc
ording to button selection
ON INT((ID/2)+.5) GOSUB Gender,S
peed,Status
GOTO here
END
 
Gender:
IF State(1)=2 THEN voice(0)=110:
voice(3)=0 ELSE voice(0)=200:voi
ce(3)=1
RETURN
Speed:
IF State(3)=2 THEN voice(2)=150
ELSE voice(2)=250
RETURN
Status:
speak=(State(5)=2)
RETURN
 
'Merge BUTTON subprogram at this
point
 
 
 
```

Super Turtle PILOT

Alan Poole

Here's an exciting new language for the Commodore 64. Graphics, sound, and disk commands make it the most powerful and flexible version of PILOT that we've ever seen. And, for even greater flexibility, it allows BASIC commands to be included in your PILOT programs.

PILOT—an acronym for Programmed Inquiry, Learning, Or Teaching—is a programming language developed in 1968 by John Starkweather. Originally designed as a simple language to help educators write computerized lessons, PILOT is also an excellent language for teaching beginners how to program. In addition to including the standard PILOT commands, "Super Turtle PILOT" enhances the language, adding simple commands for turtle graphics, sprites, sounds, and disk file management. If you prefer BASIC for certain tasks, you can even include BASIC commands in your Super Turtle PILOT (STP) programs.

Getting Started

Since STP is a machine language program, you will need to type it in using "MLX," the machine language entry program found elsewhere in this issue. Be sure you are familiar with MLX before you begin typing in STP. When MLX asks for starting and ending addresses, respond with the following values:

Starting address: 0801
Ending address: 2068

Type in the data and be sure to save a copy before leaving MLX.

Although Super Turtle PILOT is written in machine language, it loads and runs just like a BASIC program.

Program 2 is an example graphics program. Type it in *after* running STP. Save the program, and then load it and type RUN.

Working with STP is similar to working with BASIC. After you run Super Turtle PILOT, you can use the standard full-screen editor to type in your PILOT programs just as you would BASIC programs. The LIST, NEW, RUN, LOAD, and SAVE commands all work as they normally do in BASIC. Note, however, that multiple-statement lines are *not* allowed in PILOT.

You can include BASIC commands anywhere in your program, but the one-statement-per-line rule still applies. When assigning values to variables, you must use BASIC's LET command or PILOT's C: command. Normally acceptable BASIC statements such as A=12 or B$="TURTLE" are not allowed; you'd need to use statements like LET A=12 or C:B$="TURTLE". If you have a program made up of both BASIC and STP commands, the two can freely share variables.

The following commands have been added to make programming as easy as possible:

PLIST Sends a program listing to the printer.

DIR Displays the disk directory on the screen.

LOCK Disables the RUN/STOP and RESTORE keys and the LIST command.

PILOT Syntax

This article cannot teach you how to program in the PILOT language. If you are already familiar with BASIC, you should find PILOT relatively easy to learn. Before writing your own programs, study and modify the examples given here. If you are new to programming, your library or bookstore may have books that will teach you how to program in PILOT.

Here's a short quiz written in PILOT. Refer to this program as you read the description below of how PILOT instructions are put together. (Many of PILOT's instructions are similar to BASIC commands.)

```
10      R:EXAMPLE USING JM:
20      T:WHAT IS THE NAME OF T
   HE LARGEST OCEAN?
30      T:
40      T:ATLANTIC OCEAN
50      T:PACIFIC OCEAN
60      T:INDIAN OCEAN
70     *ANSWER
80      T:
90      A:
100     M:ATLANTIC,PACIFIC,INDI
   AN
110     JM:*ATLANTIC,*PACIFIC,*
   INDIAN
120     T:PLEASE TYPE ONE OF TH
   E ANSWERS.
130     J:*ANSWER
140    *ATLANTIC
150     T:NO, THE ATLANTIC OCEA
   N IS THE SECOND LARGEST. TR
   Y AGAIN.
160     J:*ANSWER
170    *PACIFIC
180     T:THAT'S RIGHT!
190     E:
```

```
200    *INDIAN
210      T:NO, THE INDIAN OCEAN
   {SPACE}IS THE THIRD LARGEST
   . TRY AGAIN.
220      J:*ANSWER
```

A PILOT instruction is divided into five parts:

1. *Instruction name.* One or two letters.
2. *Conditioner.* A *Y* or an *N*. Instructions with a *Y* conditioner are executed if the last Match instruction was successful. An *N* conditioner causes the instruction to be executed if the last Match failed. See below for a description of Match. Conditioners allow the program to make decisions based on user input. The conditioner is optional.
3. *Expression.* A comparison of variables, numbers, or strings. Expressions are placed inside parentheses. The instruction is executed only if the expression is true. The expression is optional.
4. *Colon (:).*
5. *Object.* Everything that follows the colon is part of the object. The object usually contains data for the command. Some instructions have subcommands that are placed in the object.

A special PILOT instruction is the *label*. A label is a word preceded by an asterisk, and it must be on a line of its own. It is used as a destination for jumps to other parts of the code.

STP Instructions

T Type. Displays the text specified in the object. Quotation marks are not used. Variables can be displayed by preceding the variable name with a pound sign (#) for numeric variables, or with a dollar sign for string variables. The variable name must be followed by a space or any other character that is not a letter or number. If you place a back arrow (←) at the end of the object, the computer will continue printing the next line at the same position on the screen (similar to the effect of using a semicolon at the end of a PRINT statement in BASIC).

R Remark. Used to put comments in the program. R: is similar to the BASIC command REM.

A Accept. Similar to BASIC's INPUT statement. Makes the computer stop and wait for the user to type a line. Any character (including commas) may be typed. An object is not required, but the input can be assigned to a particular variable by placing the variable name in the object.

M Match. The object contains a series of items (words or numbers) separated by commas. The computer will search the last line typed from an Accept for a match with one of the items in the list. If a match is found, the Y conditioner is set. Otherwise, the N conditioner is set. The position of the matching item in the list is also recorded.

J Jump. Similar to BASIC's GOTO command. Makes the computer continue executing the program at the label that matches the one in the object.

JM Jump on Match. Similar to BASIC's ON-GOTO command. The object contains a series of labels separated by commas. The computer will jump to the label at the position in the list corresponding to the position of the matched item in the most recently executed Match instruction. For example, if the last Match instruction found a match for the third item in its object list, the JM: instruction will cause a jump to the third label in the series.

U Use. Similar to BASIC's GOSUB command. The computer jumps to the label in the object. It will return to the line following the Use when an E (End) instruction is executed.

UM Use on Match. Similar to BASIC's ON-GOSUB command. Like JM:, the object contains a series of labels separated by commas, and the computer will jump to the label at the position of the most recent match.

E End. Has two functions. If a Use instruction has been executed, the computer will return to the line following the Use. If a Use has not been executed, the program ends.

C Compute. Similar to BASIC's LET command. Assigns values to a numeric or string variable. The object should consist of the variable name, an equal sign (=), and the value to be assigned to the variable. The rules for naming variables are the same as in BASIC: Variable names must begin with an alphabetic character and can include any number of alphanumeric characters, but only the first two are significant (for example, DOWN and DOG are both equivalent to DO). String variable names must end with a $. The value to be assigned to the variable can be either a literal value or an expression to be evaluated. The expression can include other variables, even ones defined in BASIC lines, and any of the BASIC arithmetic operators and functions. For example, C:X=4 and C:X=(Y*2)+COS(Z) are both valid.

PA Pause. Creates a delay. The object specifies the length of time to pause (in 1/60-second units).

PC Position Cursor. Sets the screen position of the cursor. The object contains two numbers separated by a comma. The first number is the vertical row (0–24). The second number is the horizontal column (0–39).

H Home. Clears the screen and places the cursor in the top left corner.

S Sound. This must precede all sound commands. See the discussion of sound commands below.

G Graphics. This must precede all graphics commands. See the discussion of graphics commands below. Multiple graphics commands can follow a G: instruction if the commands are separated by semicolons (;).

F File. This must precede all disk commands. See the discussion of file commands below.

DS Define Sprite. Use this command to define the shapes of a sprite. The first instruction in each pattern definition should consist of DS: followed by a # and the sprite number (0–7). Subsequent DS: instructions are used to define the pattern for each line of the sprite. The object for each DS: pattern instruction is a series of up to 24 periods and *X*'s. An *X* tells the computer to draw a dot in the corresponding position. A period indicates that nothing is to be drawn. Each sprite can be up to 21 lines in height, so up to 21 DS: pattern instructions can be used for each sprite. See the SPRITE command below for additional information about using sprites in STP.

PILOT Sound

To use the sound commands most effectively, you should be familiar with the operation of the 64's SID chip. Check your *Programmer's Reference Guide* or COMPUTE!'s *Mapping the 64* for details.

To create a sound, the S: instruction is followed by one, two, or three pairs of numbers. The first number in each pair is the pitch (1–65535). The second number is the duration (1–255).

To change the character of the sound produced, the S: instruction can also be followed by one of the following commands:

VOLUME Followed by a number from 0 to 15.

WAVE Followed by a value which specifies the waveform: 1 for triangle, 2 for sawtooth, 3 for pulse, or 4 for random noise.

PULSE Followed by a number from 0 to 4095 to set the pulse width.

ADSR Followed by two numbers separated by a comma. The first number is the attack/decay setting (0–255). The second number is the sustain/release value (0–255).

Graphics Commands

All of the following graphics commands must be preceded by PILOT's G: instruction. More than one command can follow the G:, but multiple commands must be separated by semicolons.

Some commands manipulate a *turtle*, a kind of graphics cursor that shows your current screen position. The turtle walks across the screen, dragging a pen. You can tell the turtle what direction to go, how far to go, and what color of pen to drag. Turtles make geometric drawing simple. For example, here's a sample program that draws a blue square:

```
10      G:CLEAR
20      G:PEN BLUE
30      G:DRAW 25;TURN 90;DRAW
   {SPACE}25;TURN 90;DRAW 25;T
   URN 90;DRAW 25
```

CLEAR Sets up and erases a high-resolution graphics screen with a four-line text window at the bottom.

FULL Makes the text window disappear to display full-screen graphics.

MIX Displays a four-line text window at the bottom of the hi-res graphics screen.

QUIT Returns the screen to normal text mode.

POINT Draws a dot on the graphics screen. The command is followed by two values separated by a comma. The values specify the horizontal *(x)* and vertical *(y)* coordinates of the point. The Super Turtle PILOT screen coordinate system is quite different from the commonly used one which places the 0,0 point in the upper left corner of the screen. In STP, the 0,0 point is at the center of the screen. As a result, the *x* coordinate can range from −79 to 80, and the *y* coordinate can range from −99 to 100. Positive values specify points above or to the right of the center, while negative coordinates are below or to the left.

LINE Draws a line. LINE is followed by coordinates of the first point, the word TO, and the coordinates of the second point. The coordinates used are the same as those used for the POINT command, with the *x* coordinate in the range −79 to 80 and the *y* coordinate in the range −99 to 100.

DRAW Tells the turtle to draw a line in the direction of its current heading. The length of the line in pixels is specified by the number following the DRAW command. The turtle can move freely on or off the screen.

DRAWTO Makes the turtle draw a line from its current position to the point specified by the coordinates following the DRAWTO command. The coordinate system is the same as that used in the POINT command.

GO Moves the turtle a specified number of pixels in the direction of its current heading without drawing a line on the screen.

GOTO Makes the turtle jump to the specified coordinates without drawing a line.

TURN Tells the turtle to turn the number of degrees indicated by the number following the TURN command. A positive number turns the turtle counterclockwise, and a negative number turns it clockwise. For example, if the turtle's current heading is 90 degrees, then a TURN 45 command will change the heading to 135 degrees.

TURNTO Sets the angle of the turtle's heading. The angle is specified in degrees, where 0 is east (right), 90 is north (straight up), 180 is west (left), and 270 is south (straight down). Since there are 360 degrees in a circle, the heading is periodic with respect to 360 degrees. For example, TURNTO 360 is equivalent to TURNTO 0, and TURNTO 450 is equivalent to TURNTO 90. The next time you move the turtle, it will move in the specified direction.

PEN Sets the turtle's pen color. The command should be followed by a color number from 0 to 15 or by one of these color names: BLACK, WHITE, RED, CYAN, PURPLE, GREEN, BLUE, YELLOW, ORANGE, BROWN, LT RED, DARK GRAY, MED GRAY, LT GREEN, LT BLUE, or LT GRAY. The PEN command may also be followed by UP, to make the turtle stop drawing lines, or by DOWN, to return it to drawing. PEN ERASE tells the turtle to erase rather than draw as it moves.

FILL Moves the turtle like the DRAW command, but the turtle also fills in to the right with color as it moves. The fill continues until it is blocked by something previously drawn on the screen. If the fill reaches the right side of the screen, it continues on the other side. As a result, you should use this command only when the turtle is in a previously drawn enclosed shape.

BKGD Sets background color of the graphics or text screen.

WINDOW Sets background color of the text window on the graphics screen.

BORDER Sets the border color of the graphics or text screen.

SPRITE Draws a sprite on the text or graphics screen. The command should be followed by the sprite number, a comma, and the coordinates at which the sprite is to be positioned. Unlike the other graphics commands, the SPRITE command uses the standard VIC chip coordinates to specify sprite position—*x* coordinates can be in the range 0–511, and *y* coordinates can be in the range 0–255. However, only *x* coordinates of 24–343 and *y* coordinates of 50–249 place the sprite fully on the visible portion of the screen. (For values out-

side these ranges, the sprite may be partially or completely hidden behind the screen border.) Sprite color is set according to the most recent PEN command.

PRINT Sends a printout of the graphics screen to the printer. This command will work only if your printer is a Commodore 1525 or compatible, or if your printer interface provides 1525 emulation. You should also make sure that the printer is turned on before giving this command. The command should be followed by a 1, 2, or 3 to indicate the size of the image to be printed.

The following program demonstrates a variety of the graphics, sprite, and sound commands:

```
10      G:CLEAR;FULL;PEN RED;BO
   RDER BLACK
20      G:LINE -79,100 TO 80,10
   0
30      G:LINE 80,100 TO 80,-99
40      G:LINE 80,-99 TO -79,-9
   9
50      G:LINE -79,-99 TO -79,1
   00
55      G:PEN WHITE
60      S:ADSR 0,242
70      DS:#0
80      DS:..XXXX
90      DS:.XXXXXX
100     DS:XXXXXXXX
110     DS:XXXXXXXX
120     DS:.XXXXXX
130     DS:..XXXX
140     C:X=180
150     C:Y=150
170     C:CX=RND(1)*10-5
180     C:CY=RND(1)*10-5
190     J(CX<2 AND CY<2):170
200     C:X=X+CX
210     C:Y=Y+CY
220     G:SPRITE 0,X,Y
230     U(X>333 OR X<29):*BOUNC
   E1
240     U(Y>241 OR Y<52):*BOUNC
   E2
250     GET K$
260     J(K$=" "):140
270     J:200
280    *BOUNCE1
290     C:CX=-CX
300     S:8000,3
310     E:
320    *BOUNCE2
330     C:CY=-CY
340     J:300
```

File Commands

The F: instruction provides the following commands for manipulating disk files:

WRITE Sends all output from subsequent T (Type) instructions to a sequential file on the disk. The WRITE command should be followed by the name of the file. No quotation marks are used around the filename.

READ Takes all input for subsequent A (Accept) instructions from a sequential file on the disk. The filename is specified following the READ command.

CLOSE Closes any open disk files and returns input and output to normal. The command requires no following name or number.

Note: The author has available a 36-page Super Turtle PILOT Reference Manual, which includes many programming examples. For information, write:

Alan Poole
610 Crestwood Cir.
Bountiful, UT 84010

For instructions on entering these programs, please refer to "COMPUTE!'s Guide to Typing In Programs" elsewhere in this issue.

Program 1: Super Turtle Pilot

```
0801:0B 08 0A 00 9E 32 30 36 2E
0809:31 00 00 00 A9 05 8D 20 4E
0811:D0 A9 01 8D 21 D0 AD 02 96
0819:DD 09 03 8D 02 DD AD 00 76
0821:DD 29 FC 09 01 8D 00 DD B6
0829:AD 18 D0 29 0F 09 20 8D 2D
0831:18 D0 A9 88 8D 88 02 A9 7B
0839:E0 8D 04 03 A9 0A 8D 05 63
0841:03 A9 C3 8D 02 03 A9 0A 08
0849:8D 03 03 A9 48 8D 08 03 67
0851:A9 09 8D 09 03 A9 5C 8D BF
0859:06 03 A9 0B 8D 07 03 A9 4B
0861:99 8D 18 03 A9 18 8D 19 B6
0869:03 AD 14 03 8D 40 9C AD 6D
0871:15 03 8D 41 9C AD 18 D0 2F
0879:8D 45 9C A9 40 8D 6B 9C 7B
0881:A9 0F 8D 6C 9C A9 10 8D DB
0889:6D 9C A9 AA 8D 6E 9C A9 60
0891:FF 8D 6F 9C A9 00 8D 70 95
0899:9C A9 01 8D 4B 9C A9 01 7C
08A1:8D 4C 9C A9 01 85 2B A9 D7
08A9:00 8D 00 21 8D 46 9C 8D 7B
08B1:66 9C 8D 5F 9C 8D 67 9C 4A
08B9:8D 6A 9C 8D 76 9C 8D 77 50
08C1:9C 8D 80 9C 8D 82 9C A9 B6
08C9:21 85 2C A9 FF 85 37 A9 1A
08D1:83 85 38 A9 18 A0 09 20 1C
08D9:1E AB 20 44 A6 A9 00 85 8D
08E1:22 8D 74 9C A9 84 85 23 4C
08E9:8D 75 9C A0 00 98 91 22 63
08F1:E6 22 D0 02 E6 23 A5 22 69
08F9:C9 FF D0 F1 A5 23 C9 85 FA
0901:D0 EB A2 00 8A 18 69 10 62
0909:9D F8 8B 9D F8 8F E8 8A D5
0911:C9 08 D0 F1 4C EA A7 93 34
0919:13 1F 12 53 55 50 45 52 BC
0921:20 54 55 52 54 4C 45 20 A6
0929:50 49 4C 4F 54 20 20 20 B7
0931:20 20 20 20 20 20 42 59 C0
0939:20 41 4C 41 4E 20 50 4F 2C
0941:4F 4C 45 92 0D 0D 00 20 9C
0949:73 00 85 FB E9 80 90 11 09
0951:A5 FB C9 8A D0 05 A9 00 05
0959:8D 66 9C 20 79 00 4C E7 AD
0961:A7 20 68 09 4C EA A7 A5 EF
0969:FB C9 2A D0 03 4C 32 0A F5
0971:A2 00 BD 8E 09 C5 FB F0 BD
0979:08 E8 E0 0F D0 F4 4C 08 D1
0981:AF 8A 0A AA BD 9E 09 48 BC
0989:BD 9D 09 48 60 52 43 50 AA
0991:47 54 4A 55 45 41 4D 53 18
0999:48 4C 44 46 31 0A 35 0A F5
09A1:40 0A 07 0C FA 16 F4 18 2A
09A9:00 1A 49 1A 5A 1A 36 1B CF
09B1:9B 1C 37 1E 45 1E 6B 1E F8
09B9:6E 1F 20 09 A9 84 FB 20 D6
09C1:06 A9 C4 FB D0 05 68 68 6D
09C9:4C 08 AF 20 73 00 C9 3A 65
09D1:D0 08 A9 19 8D CB 02 4C 01
09D9:1A 0A C9 59 D0 0D 20 73 B8
09E1:00 AE 46 9C D0 05 68 68 06
09E9:4C 32 0A C9 4E D0 0D 20 7C
09F1:73 00 AE 46 9C F0 05 68 13
09F9:68 4C 32 0A C9 28 D0 0D D7
0A01:20 F1 AE 20 2B BC 30 05 2B
0A09:68 68 4C 32 0A 20 48 17 90
0A11:C9 3A F0 05 68 68 4C 08 8C
0A19:AF AD 80 9C 8D C6 02 D0 A6
0A21:04 20 73 00 60 20 42 17 CC
0A29:8D C7 02 A9 00 8D 80 9C A4
0A31:60 20 3B A9 60 20 BB 09 83
0A39:68 68 20 79 00 4C E7 A7 DF
0A41:20 73 00 C9 41 F0 0B C9 8C
0A49:43 F0 36 C9 9B F0 52 4C 30
0A51:08 AF 20 BB 09 20 9E AD C8
0A59:20 F7 B7 A5 A2 18 65 14 21
0A61:85 FF A5 A1 65 15 85 FE 90
0A69:A5 A0 69 00 85 FD A5 A2 B7
0A71:C5 FF D0 FA A5 A1 C5 FE 70
0A79:D0 F4 A5 A0 C5 FD D0 EE A8
0A81:60 20 BB 09 20 EB B7 A5 9B
0A89:D5 85 FB E6 FB 8A C5 FB 69
0A91:B0 0C A8 A6 14 E0 19 B0 87
0A99:05 18 20 F0 FF 60 4C 48 AB
0AA1:B2 A9 00 20 BD FF A9 04 C0
0AA9:A2 04 A0 FF 20 BA FF 20 30
0AB1:C0 FF A2 04 20 C9 FF A9 8C
0AB9:01 8D 77 9C 20 73 00 4C 85
0AC1:9C A6 AD 77 9C D0 03 4C 75
0AC9:83 A4 A9 0D 20 D2 FF 20 3B
0AD1:CC FF A9 04 20 C3 FF A9 7B
0AD9:00 8D 77 9C 4C 83 A4 A5 69
0AE1:7A 48 A5 7B 48 20 79 00 67
0AE9:C9 2A F0 21 C9 54 F0 24 43
0AF1:C9 4A F0 20 C9 55 F0 1C 3F
0AF9:C9 4D F0 18 C9 52 F0 14 73
0B01:C9 46 F0 10 68 85 7B 68 65
0B09:85 7A 4C 7C A5 A9 2A 85 7F
0B11:FB 4C 19 0B A9 3A 85 FB 49
0B19:A9 00 85 FD 20 79 00 C9 45
0B21:00 F0 E1 C9 22 D0 02 E6 8B
0B29:FD C5 FB F0 06 20 73 00 D5
0B31:4C 20 0B A9 22 A0 00 91 96
0B39:7A 68 85 7B 68 85 7A 20 7D
0B41:7C A5 A2 00 E8 BD FF 01 92
0B49:C9 22 D0 F8 A5 FD F0 05 82
0B51:C6 FD 4C 45 0B A5 FB 9D AC
0B59:FF 01 60 84 FB 85 FC A5 99
0B61:FB C9 04 D0 56 A0 02 B1 60
0B69:5F 85 FD C8 B1 5F 85 FE F1
0B71:C9 27 90 08 D0 38 A5 FD 79
0B79:C9 10 B0 32 A9 20 20 47 07
0B81:AB A5 FE C9 03 90 08 D0 8E
0B89:25 A5 FD C9 E8 B0 1F A9 E9
0B91:20 20 47 AB A5 FE D0 16 44
0B99:A5 FD C9 64 B0 10 A9 20 BA
0BA1:20 47 AB A5 FD C9 0A B0 45
0BA9:05 A9 20 20 47 AB A0 04 E0
0BB1:B1 5F C9 2A F0 05 A9 20 63
0BB9:20 47 AB AD 67 9C F0 0A 9B
0BC1:A9 1B 8D 6A 9C A9 2D 4C FD
0BC9:1A A7 AD 6A 9C F0 03 4C 2E
0BD1:C6 0B A0 04 B1 5F C9 2A 2B
0BD9:F0 26 C9 54 F0 17 C9 4A 32
0BE1:F0 13 C9 55 F0 0F C9 4D 68
0BE9:F0 0B C9 52 F0 07 A4 FB 82
0BF1:A5 FC 4C 1A A7 C8 C4 FB 2B
0BF9:B0 F4 B1 5F C9 3A D0 F5 A0
0C01:A4 FB A5 FC 4C F3 A6 20 8E
0C09:BB 09 D0 03 4C 08 AF B0 1E
0C11:20 20 79 00 20 9E B7 8E E9
0C19:62 9C 20 79 00 C9 5B F0 F3
0C21:03 4C 08 AF 20 73 00 A5 3E
0C29:7A 8D 63 9C A5 7B 8D 64 B2
```

```
0C31:9C 20 9B 0C 8A D0 03 4C BD
0C39:08 AF A9 0C 48 A9 4D 48 03
0C41:CA 8A 0A AA BD 76 0C 48 75
0C49:BD 75 0C 48 60 20 79 00 1A
0C51:C9 3B D0 06 20 73 00 4C B2
0C59:32 0C C9 5D D0 15 20 73 2B
0C61:00 CE 62 9C F0 0D AD 63 BD
0C69:9C 85 7A AD 64 9C 85 7B 77
0C71:4C 32 0C 60 26 0D 52 0E DB
0C79:92 0E 98 0E 9E 0E 9E 10 CC
0C81:95 12 8B 13 95 13 9C 13 D0
0C89:A3 13 34 14 F1 14 21 15 37
0C91:55 15 A8 15 B6 15 24 16 68
0C99:32 16 A9 01 85 FB A2 00 F6
0CA1:A0 00 B1 7A DD CB 0C D0 EE
0CA9:10 C8 E8 BD CB 0C D0 F2 18
0CB1:20 73 00 88 D0 FA A6 FB FA
0CB9:60 E8 BD CB 0C D0 FA E6 31
0CC1:FB E8 BD CB 0C D0 D9 A2 80
0CC9:00 60 43 4C 45 41 52 00 FA
0CD1:51 55 49 54 00 46 55 4C 66
0CD9:4C 00 4D 49 58 00 50 4F 09
0CE1:B5 00 4C 49 4E 45 00 50 CA
0CE9:45 4E 00 42 4B 47 44 00 5C
0CF1:57 49 4E 44 4F 57 00 42 30
0CF9:B0 44 45 52 00 44 52 41 40
0D01:57 A4 00 44 52 41 57 00 7A
0D09:54 55 52 4E A4 00 54 55 F4
0D11:52 4E 00 89 00 CB 00 53 03
0D19:50 52 49 54 45 00 46 49 5E
0D21:4C 4C 00 99 00 00 AD 15 7E
0D29:03 C9 0D F0 06 AD 18 D0 CF
0D31:8D 45 9C 20 53 0E A9 01 20
0D39:8D 4B 9C A9 01 8D 4C 9C 8E
0D41:A9 00 8D 21 D0 8D 5F 9C 0C
0D49:A9 05 8D 20 D0 A9 13 20 A0
0D51:D2 FF A9 93 20 D2 FF A9 39
0D59:00 85 FB A9 A0 85 FC A0 A4
0D61:00 98 91 FB C8 D0 FB E6 FB
0D69:FC A5 FC C9 C0 D0 F2 A9 80
0D71:00 85 FB A9 8C 85 FC A9 25
0D79:00 85 FD A9 D8 85 FE A9 D3
0D81:00 91 FB 91 FD E6 FB D0 EC
0D89:02 E6 FC E6 FD D0 02 E6 8A
0D91:FE A5 FC C9 8F D0 E8 A5 08
0D99:FB C9 E7 D0 E2 A9 3C 8D F1
0DA1:18 D0 AD 11 D0 09 20 8D 3B
0DA9:11 D0 AD 16 D0 09 10 8D EF
0DB1:16 D0 78 AD 1A D0 09 01 1C
0DB9:8D 1A D0 AD 11 D0 29 7F B3
0DC1:8D 11 D0 A9 D9 8D 44 9C C5
0DC9:8D 12 D0 AD 14 03 8D 40 2C
0DD1:9C AD 15 03 8D 41 9C A9 CC
0DD9:FB 8D 14 03 A9 0D 8D 15 B9
0DE1:03 A9 7F 8D 0D DC A9 00 DF
0DE9:8D 42 9C A9 03 8D 43 9C FA
0DF1:58 A2 15 A0 00 18 20 F0 1F
0DF9:FF 60 AD 19 D0 8D 19 D0 33
0E01:AD 12 D0 CD 44 9C F0 26 0C
0E09:A9 3C 8D 18 D0 AD 16 D0 76
0E11:09 10 8D 16 D0 AD 11 D0 F8
0E19:09 20 29 7F 8D 11 D0 AD DE
0E21:42 9C 8D 21 D0 AD 44 9C AB
0E29:8D 12 D0 6C 40 9C AD 43 84
0E31:9C 8D 21 D0 AD 45 9C 8D 79
0E39:18 D0 AD 11 D0 29 5F 8D D3
0E41:11 D0 AD 16 D0 29 EF 8D C9
0E49:16 D0 A9 FF 8D 12 D0 4C 7C
0E51:BC FE 78 A9 81 8D 0D DC 6E
0E59:AD 1A D0 29 FE 8D 1A D0 B2
0E61:AD 40 9C 8D 14 03 AD 41 1A
0E69:9C 8D 15 03 20 7C 0E 58 71
0E71:A9 05 8D 20 D0 A9 01 8D 14
0E79:21 D0 60 AD 11 D0 29 DF 3F
0E81:8D 11 D0 AD 16 D0 29 EF D3
0E89:8D 16 D0 AD 45 9C 8D 18 B6
0E91:D0 60 A9 FA 8D 44 9C 60 2A
0E99:A9 D9 8D 44 9C 60 20 BD 5B
0EA1:0E A5 04 F0 03 4C 48 B2 4A
0EA9:A5 03 C9 A0 B0 F7 A5 06 53
0EB1:D0 F3 A5 05 C9 C8 B0 ED F8
0EB9:20 EB 0E 60 20 9E AD 20 9F
0EC1:BF B1 A5 65 18 69 4F 85 BF
0EC9:03 A5 64 69 00 85 04 20 32
0ED1:FD AE 20 9E AD 20 BF B1 A5
0ED9:A5 65 49 FF 18 69 65 85 02
0EE1:05 A5 64 49 FF 69 00 85 36
0EE9:06 60 A5 04 F0 01 60 A5 08
0EF1:03 C9 A0 B0 F9 A5 06 D0 64
0EF9:F5 A5 05 C9 C8 B0 EF 20 C0
0F01:3D 10 A5 03 29 03 85 FB 03
0F09:A9 FC 85 FC A9 03 38 E5 6B
0F11:FB 85 FB AD 4B 9C A6 FB FE
0F19:F0 0A 0A 0A 38 26 FC 26 8E
0F21:FC CA D0 F6 85 FB A5 01 62
0F29:29 FE 85 01 A5 FC A0 00 BE
0F31:31 22 05 FB 91 22 A5 01 32
0F39:09 01 85 01 AD 4B 9C D0 81
0F41:01 60 A5 05 29 F8 85 FB 31
0F49:85 FD A9 00 85 FC 85 FE 09
0F51:06 FD 26 FE 06 FD 26 FE 1A
0F59:A5 FD 18 65 FB 85 FB A5 B6
0F61:FE 65 FC 85 FC A5 03 4A 1F
0F69:4A 18 65 FB 85 FB A9 00 8E
0F71:65 FC 85 FC A9 00 18 65 E4
0F79:FB 85 FB A9 8C 65 FC 85 8A
0F81:FC AD 4B 9C C9 01 F0 0A FA
0F89:B1 FB 29 F0 0D 4C 9C 91 18
0F91:FB 60 AD 4C 9C 0A 0A 0A 6B
0F99:0A 85 FD B1 FB 29 0F 05 A0
0FA1:FD 91 FB 60 A5 03 48 A5 18
0FA9:04 48 A5 05 48 A5 06 48 0E
0FB1:E6 03 D0 02 E6 04 A5 04 D4
0FB9:D0 06 A5 03 C9 A0 90 06 9E
0FC1:A9 00 85 03 85 04 A5 06 23
0FC9:D0 11 A5 05 C9 C8 B0 0B 77
0FD1:20 E9 0F D0 06 20 EB 0E FF
0FD9:4C B1 0F 68 85 06 68 85 8D
0FE1:05 68 85 04 68 85 03 60 4D
0FE9:A5 04 D0 4D A5 03 C9 A0 38
0FF1:B0 47 A5 06 D0 43 A5 05 33
0FF9:C9 C8 B0 3D 20 3D 10 A5 D4
1001:03 29 03 85 FB A9 03 38 6A
1009:E5 FB 85 FB A9 03 A6 FB 2E
1011:F0 05 0A 0A CA D0 FB A0 FE
1019:00 85 FD A5 01 29 FE 85 E4
1021:01 A5 FD 31 22 85 FD A5 C6
1029:01 09 01 85 01 A6 FB F0 10
1031:05 4A 4A CA D0 FB A5 FD 1C
1039:60 A9 00 60 A9 00 85 22 74
1041:A9 A0 85 23 A5 05 29 F8 CD
1049:85 FB A9 00 06 FB 2A 06 DA
1051:FB 2A 06 FB 2A 85 FC A5 81
1059:22 18 65 FB 85 22 A5 23 20
1061:65 FC 85 23 06 FB 26 FC BF
1069:06 FB 26 FC A5 22 18 65 6B
1071:FB 85 22 A5 23 65 FC 85 BD
1079:23 A5 05 29 07 18 65 22 4D
1081:85 22 A5 23 69 00 85 23 4D
1089:A9 00 85 FB A5 03 29 FC 77
1091:0A 26 FB 65 22 85 22 A5 27
1099:23 65 FB 85 23 60 20 BD 15
10A1:0E A5 03 C9 A0 B0 40 85 FC
10A9:C3 A5 04 D0 3A A9 00 85 A0
10B1:C4 85 36 85 1A 85 1C A5 79
10B9:05 C9 C8 B0 2A 85 35 A5 6A
10C1:06 D0 24 A9 A4 20 FF AE 8C
10C9:20 BD 0E A5 03 C9 A0 B0 B6
10D1:16 85 19 A5 04 D0 10 A5 05
10D9:05 C9 C8 B0 0A 85 1B A5 55
10E1:06 D0 04 20 EB 10 60 4C 68
10E9:48 B2 A5 19 38 E5 C3 8D 8F
10F1:55 9C A5 1A E5 C4 8D 56 ED
10F9:9C 30 0C A9 01 8D 59 9C 1E
1101:A9 00 8D 5A 9C F0 08 A9 B1
1109:FF 8D 59 9C 8D 5A 9C A5 38
1111:1B 38 E5 35 8D 57 9C A5 87
1119:1C E5 36 8D 58 9C 30 0C 04
1121:A9 01 8D 5B 9C A9 00 8D D8
1129:5C 9C F0 08 A9 FF 8D 5B 03
1131:9C 8D 5C 9C AD 56 9C 10 6A
1139:18 49 FF 8D 56 9C AD 55 68
1141:9C 49 FF 18 69 01 8D 55 45
1149:9C AD 56 9C 69 00 8D 56 76
1151:9C AD 58 9C 10 18 49 FF 75
1159:8D 58 9C AD 57 9C 49 FF 86
1161:18 69 01 8D 57 9C AD 58 C3
1169:9C 69 00 8D 58 9C A5 C3 51
1171:85 03 A5 C4 85 04 A5 35 D4
1179:85 05 A5 36 85 06 20 EB 28
1181:0E AD 76 9C F0 03 20 A5 28
1189:0F AD 56 9C CD 58 9C 90 CC
1191:10 F0 03 4C 1C 12 AD 55 F6
1199:9C CD 57 9C 90 03 4C 1C 77
11A1:12 AD 55 9C 8D 5D 9C AD 75
11A9:56 9C 8D 5E 9C AD 5D 9C A8
11B1:18 6D 55 9C 8D 5D 9C AD 78
11B9:5E 9C 6D 56 9C 8D 5E 9C B9
11C1:CD 58 9C 90 2C D0 08 AD DF
11C9:5D 9C CD 57 9C 90 22 A5 02
11D1:03 18 6D 59 9C 85 03 A5 65
11D9:04 6D 5A 9C 85 04 AD 5D 63
11E1:9C 38 ED 57 9C 8D 5D 9C 06
11E9:AD 5E 9C ED 58 9C 8D 5E 9B
11F1:9C A5 06 C5 1C D0 07 A5 C0
11F9:05 C5 1B D0 01 60 A5 05 5A
1201:18 6D 5B 9C 85 05 A5 06 53
1209:6D 5C 9C 85 06 20 EB 0E 7D
1211:AD 76 9C F0 03 20 A5 0F 3F
1219:4C AE 11 AD 57 9C 8D 5D B1
1221:9C AD 58 9C 8D 5E 9C AD A0
1229:5D 9C 18 6D 57 9C 8D 5D A2
1231:9C AD 5E 9C 6D 58 9C 8D 38
1239:5E 9C CD 56 9C 90 2C D0 23
1241:08 AD 5D 9C CD 55 9C 90 D7
1249:22 A5 05 18 6D 5B 9C 85 A1
1251:05 A5 06 6D 5C 9C 85 06 5F
1259:AD 5D 9C 38 ED 55 9C 8D 4E
1261:5D 9C AD 5E 9C ED 56 9C DC
1269:8D 5E 9C A5 04 C5 1A D0 16
1271:07 A5 03 C5 19 D0 01 60 AD
1279:A5 03 18 6D 59 9C 85 03 56
1281:A5 04 6D 5A 9C 85 04 20 EF
1289:EB 0E AD 76 9C F0 03 20 13
1291:A5 0F 4C 28 12 20 D5 12 27
1299:C9 10 90 21 C9 10 D0 06 01
12A1:A9 01 8D 5F 9C 60 C9 11 8D
12A9:D0 06 A9 00 8D 5F 9C 60 70
12B1:C9 12 D0 06 A9 00 8D 4B 6D
12B9:9C 60 4C 08 AF 8D 4C 9C 37
12C1:AD 4B 9C 18 69 01 C9 03 8A
12C9:90 06 A9 01 8D 4B 9C 60 30
12D1:8D 4B 9C 60 A9 00 85 FB 7D
12D9:AA A0 00 B1 7A DD 10 13 15
12E1:D0 16 C8 E8 BD 10 13 D0 C0
12E9:F2 98 18 65 7A 85 7A A5 8B
12F1:7B 69 00 85 7B A5 FB 60 51
12F9:E8 BD 10 13 D0 FA E6 FB 71
1301:E8 BD 10 13 D0 D3 20 9E F2
1309:AD 20 BF B1 A5 65 60 42 E6
1311:4C 41 43 4B 00 57 48 49 02
1319:54 45 00 52 45 44 00 43 5E
1321:59 41 4E 00 50 55 52 50 DA
1329:4C 45 00 47 52 45 45 4E BB
1331:00 42 4C 55 45 00 59 45 E8
1339:4C 4C 4F 57 00 B0 41 4E 8B
1341:47 45 00 42 52 4F 57 4E 4D
1349:00 4C 54 20 52 45 44 00 3F
1351:44 41 52 4B 20 47 52 41 EC
1359:59 00 4D 45 44 20 47 52 AD
1361:41 59 00 4C 54 20 47 52 47
1369:45 45 4E 00 4C 54 20 42 83
1371:4C 55 45 00 4C 54 20 47 F6
1379:52 41 59 00 55 50 00 44 74
1381:4F 57 4E 00 45 52 41 53 38
1389:45 00 00 20 D5 12 8D 21 87
1391:D0 8D 42 9C 60 20 D5 12 D6
1399:8D 43 9C 60 20 D5 12 8D FA
13A1:20 D0 60 A5 7A 8D 60 9C D9
13A9:A5 7B 8D 61 9C A9 2F 85 B8
13B1:7A A9 14 85 7B 20 BD 0E 40
13B9:A5 03 85 C3 A5 04 85 C4 6D
13C1:A5 05 85 35 A5 06 85 36 86
13C9:AD 60 9C 85 7A AD 61 9C B4
13D1:85 7B 20 BD 0E A5 03 85 0C
13D9:19 A5 04 85 1A A5 05 85 C5
13E1:1B A5 06 85 1C AD 5F 9C 0B
13E9:D0 03 20 EB 10 A9 2F 85 07
13F1:7A A9 14 85 7B 20 8B B0 BE
```

```
13F9:20 85 B1 A5 47 85 FB A5 10
1401:48 85 FC 20 FD AE 20 8B C6
1409:B0 20 85 B1 AD 60 9C 85 0B
1411:7A AD 61 9C 85 7B 20 9E D0
1419:AD A6 FB A4 FC 20 D4 BB 59
1421:20 FD AE 20 9E AD A6 47 F0
1429:A4 48 20 D4 BB 60 51 58 61
1431:2C 51 59 00 20 9E AD A5 6B
1439:7A 8D 60 9C A5 7B 8D 61 6F
1441:9C A9 B4 85 7A A9 14 85 39
1449:7B 20 8B B0 20 85 B1 A6 D4
1451:47 A4 48 20 D4 BB A9 2F 69
1459:85 7A A9 14 85 7B 20 BD 71
1461:0E A5 03 85 C3 A5 04 85 F4
1469:C4 A5 05 85 35 A5 06 85 28
1471:36 A9 B7 85 7A A9 14 85 96
1479:7B 20 79 00 20 ED A7 20 BE
1481:73 00 20 ED A7 A9 2F 85 0E
1489:7A A9 14 85 7B 20 BD 0E 1A
1491:A5 03 85 19 A5 04 85 1A F1
1499:A5 05 85 1B A5 06 85 1C A4
14A1:AD 5F 9C D0 03 20 EB 10 99
14A9:AD 60 9C 85 7A AD 61 9C 96
14B1:85 7B 60 51 4C 00 51 58 F9
14B9:B2 51 58 AA 28 51 4C AC 11
14C1:BE 28 51 41 AC FF AD 31 83
14C9:38 30 29 AA 2E 30 30 30 AC
14D1:31 29 AC 2E 38 3A 51 59 FB
14D9:B2 51 59 AA 51 4C AC BF 5A
14E1:28 51 41 AC FF AD 31 38 B6
14E9:30 29 AA 2E 30 30 30 31 80
14F1:00 20 9E AD A5 7A 8D 60 63
14F9:9C A5 7B 8D 61 9C A9 1F 12
1501:85 7A A9 15 85 7B 20 8B F8
1509:B0 20 85 B1 A6 47 A4 48 43
1511:20 D4 BB AD 60 9C 85 7A CD
1519:AD 61 9C 85 7B 60 51 41 9F
1521:00 20 9E AD A5 7A 8D 60 94
1529:9C A5 7B 8D 61 9C A9 1F 43
1531:85 7A A9 15 85 7B 20 8B 29
1539:B0 20 85 B1 A5 47 A4 48 6B
1541:20 67 B8 A6 47 A4 48 20 54
1549:D4 BB AD 60 9C 85 7A AD 26
1551:61 9C 85 7B 60 A5 7A 8D D7
1559:60 9C A5 7B 8D 61 9C A9 1C
1561:2F 85 7A A9 14 85 7B 20 3C
1569:8B B0 20 85 B1 A5 47 85 1A
1571:FB A5 48 85 FC 20 FD AE 77
1579:20 8B B0 20 85 B1 A5 47 34
1581:85 FD A5 48 85 FE AD 60 0B
1589:9C 85 7A AD 61 9C 85 7B 91
1591:20 9E AD A6 FB A4 FC 20 20
1599:D4 BB 20 FD AE 20 9E AD E1
15A1:A6 FD A4 FE 20 D4 BB 60 4F
15A9:A9 01 8D 5F 9C 20 35 14 74
15B1:A9 00 8D 5F 9C 60 20 79 78
15B9:00 20 9E B7 E0 08 90 03 86
15C1:4C 48 B2 86 FB 20 73 00 2A
15C9:20 EB B7 A5 FB 0A A8 C8 72
15D1:8A 99 00 D0 A5 15 C9 02 CB
15D9:B0 E6 C9 01 F0 14 A9 FE 89
15E1:A6 FB F0 05 38 2A CA D0 9D
15E9:FB 2D 10 D0 8D 10 D0 4C 07
15F1:03 16 A9 01 A6 FB F0 04 73
15F9:0A CA D0 FC 0D 10 D0 8D 9D
1601:10 D0 A5 FB 0A AA A5 14 38
1609:9D 00 D0 AD 4C 9C A6 FB 17
1611:9D 27 D0 A9 01 A6 FB F0 16
1619:04 0A CA D0 FC 0D 15 D0 47
1621:8D 15 D0 60 A9 01 8D 76 5C
1629:9C 20 35 14 A9 00 8D 76 72
1631:9C 60 20 79 00 20 9E B7 D4
1639:8E 7E 9C E0 04 90 03 4C A2
1641:08 AF A9 00 20 BD FF A9 34
1649:04 A2 04 A0 FF 20 BA FF A0
1651:20 C0 FF A2 04 20 C9 FF 1C
1659:A9 08 20 D2 FF A9 0D 20 6E
1661:D2 FF A9 00 85 03 85 04 73
1669:85 06 85 05 8D 79 9C 8D F3
1671:7C 9C 8D 7F 9C A9 00 8D C5
1679:7A 9C AD 79 9C 85 05 AD 0A
1681:7F 9C 8D 7C 9C A9 07 8D 35
1689:7B 9C 20 DE 16 6E 7A 9C 88
1691:CE 7B 9C D0 F5 AD 7A 9C 9C
1699:38 6A AE 7E 9C 8E 7B 9C EC
16A1:20 D2 FF CE 7B 9C D0 F8 68
16A9:E6 03 A5 03 C9 A0 90 C5 A6
16B1:A9 0D 20 D2 FF A9 00 85 53
16B9:03 A5 05 8D 79 9C AD 7C 60
16C1:9C 8D 7F 9C A5 05 C9 C8 F6
16C9:90 AB A9 0F 20 D2 FF A9 45
16D1:0D 20 D2 FF 20 CC FF A9 C4
16D9:04 20 C3 FF 60 20 E9 0F EE
16E1:A8 EE 7C 9C AD 7C 9C CD DD
16E9:7E 9C D0 07 E6 05 A9 00 A5
16F1:8D 7C 9C 98 F0 02 38 60 81
16F9:18 60 A9 01 8D 80 9C 20 57
1701:BB 09 A2 00 8E 65 9C 20 07
1709:48 17 F0 25 C9 20 D0 0B 0D
1711:20 D2 FF 20 42 17 F0 19 6F
1719:4C 0D 17 C9 23 F0 35 C9 41
1721:24 F0 31 20 1D 18 20 42 91
1729:17 F0 06 C9 5F F0 0B D0 22
1731:EA 20 5D 18 A9 0D 20 D2 9E
1739:FF 60 20 5D 18 20 73 00 81
1741:60 E6 7A D0 02 E6 7B 8C E4
1749:81 9C A0 00 B1 7A AC 81 C5
1751:9C 48 68 60 85 FB 20 5D AC
1759:18 20 42 17 20 13 B1 B0 B6
1761:0B A5 FB 20 1D 18 20 48 D1
1769:17 4C 2A 17 20 48 17 EE 2C
1771:65 9C AE 65 9C 9D A4 9C E6
1779:20 42 17 20 13 B1 B0 EC DA
1781:20 48 17 C9 20 F0 05 20 40
1789:7C 00 90 E0 A4 FB C0 24 D0
1791:D0 0A EE 65 9C AE 65 9C E5
1799:98 9D A4 9C EE 65 9C AE CE
17A1:65 9C A9 00 9D A4 9C A5 3D
17A9:7A 8D 60 9C A5 7B 8D 61 E5
17B1:9C A9 A5 85 7A A9 9C 85 DE
17B9:7B 20 8B B0 20 85 B1 AD 52
17C1:60 9C 85 7A AD 61 9C 85 51
17C9:7B A9 00 8D 65 9C A5 FB DD
17D1:C9 23 F0 2C A0 00 B1 47 3E
17D9:8D 62 9C C8 B1 47 85 22 5F
17E1:C8 B1 47 85 23 A0 00 AD 6B
17E9:62 9C F0 0B B1 22 20 1D B2
17F1:18 C8 CE 62 9C D0 F5 20 92
17F9:5D 18 20 48 17 4C 2A 17 BA
1801:A5 47 A4 48 20 A2 BB 20 12
1809:DD BD 85 22 84 23 A0 00 5C
1811:B1 22 F0 E3 20 1D 18 C8 6D
1819:D0 F6 F0 DB 86 FB 84 FC 75
1821:AE 82 9C F0 06 20 D2 FF 42
1829:4C 58 18 C9 0D F0 20 EE 90
1831:65 9C AE 65 9C 9D A4 9C A8
1839:C9 20 D0 06 20 5D 18 4C C3
1841:58 18 20 88 18 18 6D 65 91
1849:9C C5 D5 90 0A F0 08 A9 CA
1851:0D 20 D2 FF 20 5D 18 A6 B7
1859:FB A4 FC 60 AD 82 9C F0 F7
1861:01 60 AD 65 9C F0 1F A2 BF
1869:00 E8 BD A4 9C C9 20 D0 F2
1871:08 20 88 18 A8 F0 05 A9 FC
1879:20 20 D2 FF EC 65 9C D0 23
1881:E8 A9 00 8D 65 9C 60 A5 6D
1889:D3 C5 D5 F0 0A 90 08 A4 26
1891:D5 C8 84 FD 38 E5 FD 60 05
1899:48 8A 48 98 48 A9 7F 8D 98
18A1:0D DD AC 0D DD 10 03 4C B7
18A9:72 FE 20 02 FD D0 03 6C 9C
18B1:02 80 20 BC F6 20 E1 FF CE
18B9:F0 03 4C 72 FE 20 A3 FD 91
18C1:20 1B E5 20 53 0E AD 02 B7
18C9:DD 09 03 8D 02 DD AD 00 47
18D1:DD 29 FC 09 01 8D 00 DD 87
18D9:AD 18 D0 29 0F 09 20 8D FD
18E1:18 D0 A9 05 8D 20 D0 A9 10
18E9:01 8D 21 D0 A9 06 8D 86 36
18F1:02 6C 02 A0 A9 00 85 FD DE
18F9:A5 7A 8D 60 9C A5 7B 8D 53
1901:61 9C 20 73 00 C9 4D F0 F8
1909:3C AD 60 9C 85 7A AD 61 6D
1911:9C 85 7B 20 BB 09 20 79 20
1919:00 C9 2A F0 0A 20 9E AD CD
1921:20 F7 B7 20 A3 A8 60 20 FA
1929:A8 19 A5 22 D0 07 A5 23 DD
1931:D0 03 4C E3 A8 A5 22 38 AC
1939:E9 01 85 7A A5 23 E9 00 86
1941:85 7B 60 51 4D 20 BB 09 A1
1949:A5 7A 8D 60 9C A5 7B 8D A4
1951:61 9C A9 44 85 7A A9 19 57
1959:85 7B 20 8B B0 20 85 B1 AC
1961:AD 60 9C 85 7A AD 61 9C 58
1969:85 7B A5 47 A4 48 20 A2 8F
1971:BB 20 BF B1 A5 65 F0 16 57
1979:AA CA F0 9A 20 73 00 F0 3B
1981:0D C9 2C D0 F7 CA D0 F4 C0
1989:20 73 00 4C 17 19 A5 FD D3
1991:F0 12 68 AA 68 A8 68 68 97
1999:68 68 68 68 68 98 48 8A 6E
19A1:48 CE 66 9C 4C 32 0A A5 27
19A9:2B 85 22 A5 2C 85 23 A0 CF
19B1:00 B1 22 D0 0C C8 B1 22 AA
19B9:D0 07 A9 00 85 22 85 23 2E
19C1:60 A0 04 B1 22 C9 2A F0 65
19C9:12 A0 00 B1 22 85 FB C8 30
19D1:B1 22 85 23 A5 FB 85 22 92
19D9:4C B0 19 A9 01 85 FB A9 DB
19E1:05 85 FC A4 FB B1 7A F0 6E
19E9:10 C9 2C F0 0C A4 FC D1 E9
19F1:22 D0 D6 E6 FB E6 FC D0 F8
19F9:EA A4 FC B1 22 D0 CA 60 CF
1A01:A5 7A 8D 60 9C A5 7B 8D 5E
1A09:61 9C 20 73 00 C9 4D F0 03
1A11:0A AD 60 9C 85 7A AD 61 5E
1A19:9C 85 7B 20 BB 09 AD 60 2C
1A21:9C 85 7A AD 61 9C 85 7B 33
1A29:A9 03 20 FB A3 A5 7B 48 A9
1A31:A5 7A 48 A5 3A 48 A5 39 B1
1A39:48 A9 8D 48 EE 66 9C A9 26
1A41:01 85 FD 20 F9 18 4C AE 90
1A49:A7 20 BB 09 AD 66 9C D0 72
1A51:03 4C 31 A8 CE 66 9C 4C 60
1A59:D4 A8 20 BB 09 20 A6 B3 AB
1A61:20 60 A5 A9 51 8D 08 9D 7B
1A69:A9 4D 8D 09 9D A9 24 8D 71
1A71:0A 9D A9 B2 8D 0B 9D A9 EF
1A79:22 8D 0C 9D A2 04 E8 E0 55
1A81:55 F0 10 BD FB 01 C9 22 14
1A89:D0 02 A9 27 9D 08 9D A8 3F
1A91:D0 EC E8 A9 22 9D 08 9D 56
1A99:E8 A9 00 9D 08 9D A5 7A 03
1AA1:8D 60 9C A5 7B 8D 61 9C 14
1AA9:A9 08 85 7A A9 9D 85 7B 57
1AB1:20 A5 A9 AD 60 9C 85 7A 6A
1AB9:AD 61 9C 85 7B 20 79 00 58
1AC1:D0 01 60 A9 00 85 FD AA 02
1AC9:20 79 00 9D A4 9C E8 20 CF
1AD1:73 00 F0 0B C9 24 D0 F3 03
1AD9:A0 01 84 FD 4C CC 1A A9 82
1AE1:B2 9D A4 9C A5 FD D0 0C 08
1AE9:A9 C5 E8 9D A4 9C A9 28 6E
1AF1:E8 9D A4 9C A9 51 E8 9D 62
1AF9:A4 9C A9 4D E8 9D A4 9C 55
1B01:A9 24 E8 9D A4 9C A5 FD EC
1B09:D0 06 A9 29 E8 9D A4 9C 94
1B11:A9 00 E8 9D A4 9C A5 7A 70
1B19:8D 60 9C A5 7B 8D 61 9C 8D
1B21:A9 A4 85 7A A9 9C 85 7B F3
1B29:20 A5 A9 AD 60 9C 85 7A E3
1B31:AD 61 9C 85 7B 60 20 BB DB
1B39:09 A9 01 8D 5D 9C A9 00 08
1B41:85 FD 85 FE 8D 46 9C AA C3
1B49:20 79 00 D0 03 4C 08 AF 04
1B51:C9 25 D0 07 A9 01 85 FE 9B
1B59:20 42 17 9D A4 9C E8 C9 20
1B61:24 D0 04 A9 01 85 FD 20 33
1B69:42 17 F0 04 C9 2C D0 E0 66
1B71:A9 00 9D A4 9C 86 FB A9 1B
1B79:A4 85 22 A9 9C 85 23 A5 29
1B81:7A 8D 60 9C A5 7B 8D 61 C5
1B89:9C A5 FD F0 1E A9 A4 85 AC
1B91:7A A9 9C 85 7B 20 8B B0 7F
1B99:20 85 B1 A0 00 B1 47 85 5C
1BA1:FB C8 B1 47 85 22 C8 B1 AA
1BA9:47 85 23 A9 99 85 7A A9 65
1BB1:1C 85 7B 20 8B B0 20 85 AD
1BB9:B1 A0 00 B1 47 85 FC C8 1F
```

```
1BC1:B1 47 85 24 C8 B1 47 85 B6
1BC9:25 AD 60 9C 85 7A AD 61 A6
1BD1:9C 85 7B 20 23 1C AD 46 54
1BD9:9C D0 13 20 79 00 F0 09 AD
1BE1:20 73 00 EE 5D 9C 4C 3F 29
1BE9:1B A9 00 8D 5D 9C AC 5D 05
1BF1:9C A9 00 20 91 B3 A5 7A 04
1BF9:8D 60 9C A5 7B 8D 61 9C 6E
1C01:A9 44 85 7A A9 19 85 7B AF
1C09:20 8B B0 20 85 B1 A6 47 D3
1C11:A4 48 20 D4 BB AD 60 9C F0
1C19:85 7A AD 61 9C 85 7B 4C BC
1C21:32 0A A5 FC C5 FB 90 41 F9
1C29:A5 FB F0 3D A5 FC F0 39 61
1C31:A9 00 8D 68 9C 8D 69 9C 01
1C39:AD 69 9C 18 6D 68 9C A8 A6
1C41:B1 24 AC 68 9C D1 22 D0 B8
1C49:11 EE 68 9C AD 68 9C C5 AA
1C51:FB D0 E5 A5 FE D0 1A 4C 8E
1C59:93 1C EE 69 9C AD 69 9C E1
1C61:18 65 FB C5 FC F0 03 90 1D
1C69:01 60 A9 00 8D 68 9C F0 A7
1C71:C7 AD 69 9C F0 09 A8 88 75
1C79:B1 24 C9 20 F0 01 60 AD C8
1C81:69 9C 18 6D 68 9C C5 FC AD
1C89:B0 08 A8 B1 24 C9 20 F0 C5
1C91:01 60 A9 01 8D 46 9C 60 C6
1C99:51 4D 24 20 BB 09 20 A1 38
1CA1:1D E0 01 D0 03 4C E9 1D 08
1CA9:E0 02 D0 03 4C 00 1E E0 9C
1CB1:03 D0 03 4C 14 1E E0 04 A3
1CB9:D0 03 4C 27 1E A9 00 8D 3C
1CC1:72 9C 8D 73 9C AD 6D 9C 56
1CC9:8D 05 D4 8D 0C D4 8D 13 5F
1CD1:D4 AD 6E 9C 8D 06 D4 8D 33
1CD9:0D D4 8D 14 D4 AD 6F 9C 99
1CE1:8D 02 D4 8D 09 D4 8D 10 9B
1CE9:D4 AD 70 9C 8D 03 D4 8D 7F
1CF1:0A D4 8D 11 D4 20 EB B7 DD
1CF9:A5 14 8D 00 D4 A5 15 8D B0
1D01:01 D4 8E 71 9C 20 79 00 32
1D09:F0 2B 20 73 00 20 EB B7 D1
1D11:A5 14 8D 07 D4 A5 15 8D 3A
1D19:08 D4 8E 72 9C 20 79 00 DD
1D21:F0 13 20 73 00 20 EB B7 E3
1D29:A5 14 8D 0E D4 A5 15 8D C2
1D31:0F D4 8E 73 9C AD 6C 9C 42
1D39:8D 18 D4 AD 6B 9C 18 69 1D
1D41:01 8D 04 D4 AE 72 9C F0 96
1D49:03 8D 0B D4 AE 73 9C F0 84
1D51:03 8D 12 D4 A5 A2 85 FB BE
1D59:A5 A2 C5 FB F0 FA 85 FB 02
1D61:AD 71 9C F0 0B CE 71 9C 84
1D69:D0 06 AD 6B 9C 8D 04 D4 F1
1D71:AD 72 9C F0 0B CE 72 9C D6
1D79:D0 06 AD 6B 9C 8D 0B D4 10
1D81:AD 73 9C F0 0B CE 73 9C 29
1D89:D0 06 AD 6B 9C 8D 12 D4 2E
1D91:AD 71 9C D0 C3 AD 72 9C F5
1D99:D0 BE AD 73 9C D0 B9 60 D4
1DA1:A9 01 85 FB A2 00 A0 00 B7
1DA9:B1 7A DD D1 1D D0 10 C8 49
1DB1:E8 BD D1 1D D0 F2 20 73 E1
1DB9:00 88 D0 FA A6 FB 60 E8 AE
1DC1:BD D1 1D D0 FA E6 FB E8 54
1DC9:BD D1 1D D0 D9 A2 00 60 C1
1DD1:57 41 56 45 00 56 4F 4C 6B
1DD9:55 4D 45 00 41 44 53 52 CE
1DE1:00 50 55 4C 53 45 00 00 4F
1DE9:20 79 00 20 9E B7 E0 05 2F
1DF1:B0 0A A9 08 0A CA D0 FC D6
1DF9:8D 6B 9C 60 4C 48 B2 20 78
1E01:79 00 20 9E B7 E0 10 B0 F9
1E09:07 8E 6C 9C 8E 18 D4 60 A2
1E11:4C 48 B2 20 EB B7 A5 15 7C
1E19:D0 09 A5 14 8D 6D 9C 8E DF
1E21:6E 9C 60 4C 48 B2 20 9E 78
1E29:AD 20 F7 B7 A5 14 8D 6F C6
1E31:9C A5 15 8D 70 9C 60 20 77
1E39:BB 09 A9 93 20 D2 FF A9 F9
1E41:13 20 D2 FF 60 20 73 00 D3
1E49:C9 4F D0 1C 20 73 00 C9 B2
1E51:43 D0 15 20 73 00 C9 4B 82
1E59:D0 0E A9 EA 8D 28 03 A9 22
1E61:01 8D 67 9C 20 73 00 60 67
1E69:4C 08 AF 20 73 00 C9 53 48
1E71:F0 0A C9 49 D0 03 4C 12 B3
1E79:1F 4C 08 AF 20 BB 09 C9 20
1E81:23 D0 30 20 9B B7 E0 08 11
1E89:90 03 4C 48 B2 8A 0A 0A BA
1E91:0A 0A 0A 8D 74 9C A9 00 D8
1E99:8D 75 9C 0E 74 9C 2E 75 56
1EA1:9C A9 00 18 6D 74 9C 8D 1C
1EA9:74 9C A9 84 6D 75 9C 8D CC
1EB1:75 9C 60 AD 74 9C 85 22 F9
1EB9:AD 75 9C 85 23 A0 00 84 36
1EC1:FD A2 18 20 FD 1E 26 FD 5D
1EC9:CA E0 10 F0 08 E0 08 F0 79
1ED1:04 E0 00 D0 EE A5 FD 91 F0
1ED9:22 A9 00 85 FD C8 E0 00 BE
1EE1:D0 E1 AD 74 9C 18 69 03 17
1EE9:8D 74 9C AD 75 9C 69 00 69
1EF1:8D 75 9C 20 79 00 F0 03 98
1EF9:20 73 00 60 20 79 00 D0 E0
1F01:02 18 60 E0 18 F0 03 20 0B
1F09:73 00 C9 58 D0 F3 38 60 E6
1F11:02 20 73 00 C9 52 F0 03 43
1F19:4C 08 AF 20 73 00 A9 02 68
1F21:A2 6D A0 1F 20 BD FF A9 B3
1F29:01 A2 08 A0 00 20 BA FF 91
1F31:20 C0 FF A2 01 20 C6 FF EF
1F39:20 CF FF 20 CF FF 20 CF 0C
1F41:FF 20 CF FF D0 09 20 CC 39
1F49:FF A9 01 20 C3 FF 60 20 13
1F51:CF FF AA 20 CF FF 20 CD 5B
1F59:BD 20 CF FF F0 06 20 D2 2B
1F61:FF 4C 5A 1F A9 0D 20 D2 84
1F69:FF 4C 3F 1F 24 30 20 BB 72
1F71:09 20 79 00 C9 52 D0 1A BE
1F79:20 73 00 C9 45 D0 13 20 F4
1F81:73 00 C9 41 D0 0C 20 73 31
1F89:00 C9 44 D0 05 A9 01 4C EC
1F91:DE 1F C9 43 D0 1F 20 73 2B
1F99:00 C9 4C D0 18 20 73 00 09
1FA1:C9 4F D0 11 20 73 00 C9 5C
1FA9:53 D0 0A 20 73 00 C9 45 7D
1FB1:D0 03 4C 50 20 C9 57 D0 4F
1FB9:21 20 73 00 C9 52 D0 1A 52
1FC1:20 73 00 C9 49 D0 13 20 5D
1FC9:73 00 C9 54 D0 0C 20 73 AA
1FD1:00 C9 45 D0 05 A9 02 4C 57
1FD9:DE 1F 4C 08 AF 85 FB A0 85
1FE1:00 20 73 00 99 00 02 F0 58
1FE9:06 C8 20 42 17 D0 F5 A9 17
1FF1:2C 99 00 02 A9 53 C8 99 92
1FF9:00 02 A9 2C C8 99 00 02 5F
2001:A5 FB C9 01 F0 28 A9 57 2F
2009:C8 99 00 02 C8 98 A2 00 22
2011:A0 02 20 BD FF A9 03 A2 51
2019:08 A0 02 20 BA FF 20 C0 9E
2021:FF A2 03 20 C9 FF A9 01 0F
2029:8D 82 9C 4C 4F 20 A9 52 C9
2031:C8 99 00 02 C8 98 A2 00 4A
2039:A0 02 20 BD FF A9 05 A2 7D
2041:08 A0 02 20 BA FF 20 C0 C6
2049:FF A2 05 20 C6 FF 60 20 EB
2051:73 00 A9 03 20 C3 FF A9 6A
2059:05 20 C3 FF 20 CC FF A9 7A
2061:00 8D 82 9C 60 00 00 00 22
```

Program 2: Turtle Graphics Demo Program

```
10      G:CLEAR; FULL; BORDER 1
    4
20     *LOOP
30      C:L=L+3
40      G:DRAW L; TURN 122
50      J(L<325):*LOOP
60      G:GOTO 0,0; TURNTO 0
70      C:R=RND(1)*16
80      G:PEN R; BORDER 15-R
90      C:L=0
100     J:*LOOP
```

©

Rid
A File Deleter

Paul W. Carlson

If you frequently find yourself deleting files to obtain more space on your disks, you'll appreciate this machine language program that takes the drudgery out of the process. "Rid" is written for the IBM PC/PCjr and compatibles with BASIC and version 2.0 or higher of PC-DOS or MS-DOS. Cartridge BASIC is required for the PCjr.

There is no way around the problem of disks becoming cluttered with unwanted files. How does this happen? Here are a few contributing causes:

- Most text editors and word processors automatically make backup copies so you can recover accidental (or intentional) deletions.
- Programmers often save different versions of a program as they work.
- Some compilers and assemblers create intermediate files.
- Files with time-sensitive information eventually become out-of-date.

With a collection of unnecessary files on your disk, you may have trouble locating the ones you really need. "Rid" is a program that allows you to quickly and safely delete all unwanted files from your floppy or hard disk.

Normally, deleting the unwanted files on a floppy disk (or in the subdirectory of a hard disk) is a dull, time-consuming chore in which typing mistakes can spell disaster. It's all too easy to type DEL BATFILE.BAT instead of DEL BATFILE.BAK or DEL VERSION4.BAS instead of DEL VERSION3.BAS. If you need to delete many files, you may use the wildcards * and ? to speed up deleting files with similar names. Unfortunately, using wildcards can easily result in the unintentional deletion of files.

Rid takes a completely different approach. To make the chore as pleasant as possible, it displays the names of all your files on the screen, allowing you to easily select and delete them.

Type in the BASIC program "Rid Creator." Since it creates a machine language file that will perform disk access, be sure to use the IBM Proofreader program, found elsewhere in this issue, to enter it. Be sure to save a copy to disk, but don't use the name RID.COM—that name is reserved for the machine language program that will be created later.

To start the program, type RUN. You're asked to choose between a hard disk and floppy disk version of the program. If you enter F for floppy, the program creates a version of RID.COM that allows you to change disks in the active drive before the directory is read and displayed. If you enter H, the program creates a version of RID.COM that immediately displays the current directory. If you're going to be using Rid with both floppy and hard disks, choose F. After you enter your preference, the program will create a program file named RID.COM. Hard disk users should put a copy of RID.COM in the same directory that contains the other DOS commands that are loaded from disk (the nonresident external commands such as FORMAT, DISKCOPY, and so on), or in any other directory that is included in the DOS search path. Floppy disk users should put a copy of RID.COM on their working copy of the system disk.

A New DOS Command

RID.COM works just like any other DOS command. First, be sure to set the active drive to the drive that will hold the disk with the files you wish to delete.. Then, enter RID at the DOS prompt, preceded by the drive designation of the disk containing RID.COM, if RID.COM is in a drive other than the active drive. If you created the version of RID.COM for floppy disks, you'll be told to insert a disk in the active drive. If the disk you want to use is not in the drive, place it there and press a key to continue. The active drive and current directory are displayed at the top of the screen, followed by a list of the first 100 files. RID.COM will let you delete only those files found among the first 100 in a directory, but this should seldom, if ever, be a limiting factor.

After the files are displayed, a line of instructions explaining the keyboard controls appears at the bottom of the screen. Notice that one of the files is highlighted. Use

the arrow keys to move the highlight from one filename to another. When you press the D key (upper- or lowercase), the chosen file will be deleted *immediately* from the disk, and the name of the file will disappear from the screen. You can now move the highlight to the name of the next file to be deleted. Be very careful to delete the correct file or files. If you make a mistake and delete a file that you wanted to keep, you may be able to recover your lost files with a disk utility program.

To exit RID.COM and return to DOS, press the Escape key. Once you've become accustomed to using RID.COM, you'll wonder how you got along without it.

Rid Creator

For instructions on entering these programs, please refer to "COMPUTE!'s Guide to Typing In Programs" elsewhere in this issue.

```
NA 1 ' Program to create RID.COM
NB 2 ' Copyright 1987 Compute! P
     ublications, Inc.
IK 3 ' All Rights Reserved.
GI 4 CLS:PRINT "COPYRIGHT 1987 "
     :PRINT "COMPUTE! PUBLICATIO
     NS, INC."
DC 5 PRINT "ALL RIGHTS RESERVED."
     :"
KE 6 FOR ZZ=1 TO 1500:NEXT:CLS
MP 10 INPUT"Create RID.COM for H
     ard or Floppy disk (H or F
     )";D$
LM 20 IF D$="H" OR D$="h" THEN D
     =&H7A:GOTO 50
NE 30 IF D$="F" OR D$="f" THEN D
     =&H6E:GOTO 50
FF 40 GOTO 10
DN 50 OPEN "RID.COM" FOR OUTPUT
     AS 1
PJ 60 PRINT#1,CHR$(&HE9);CHR$(D)
     ;CHR$(&H8);
HL 70 FOR N=1 TO 1927:PRINT#1,CH
     R$(0);:NEXT
FP 80 T=0:FOR J=1 TO 910:READ A$
     :N=VAL("&H"+A$)
FG 90 T=T+N:PRINT#1,CHR$(N);:NEX
     T:CLOSE 1
NI 100 IF T=84241! THEN PRINT"RI
     D.COM SUCCESSFULLY CREATE
     D!":END
KH 110 PRINT CHR$(7);"***** ERRO
     R IN DATA STATEMENTS ****
     *":END
LN 120 DATA 00,00,00,00,00,00,2A
     ,2E,2A,00
JM 130 DATA 0A,0A,0D,49,6E,73,65
     ,72,74,20
QK 140 DATA 64,69,73,6B,20,69,6E
     ,20,61,63
GB 150 DATA 74,69,76,65,20,64,72
     ,69,76,65
PN 160 DATA 2C,20,74,68,65,6E,20
     ,0A,0D,20
KA 170 DATA 20,20,70,72,65,73,73
     ,20,61,6E
DL 180 DATA 79,20,6B,65,79,20,74
     ,6F,20,63
NH 190 DATA 6F,6E,74,69,6E,75,65
     ,2E,2E,2E
CB 200 DATA 24,44,45,4C,45,54,49
     ,4E,47,20
FF 210 DATA 46,49,4C,45,53,20,46
     ,52,4F,4D
LL 220 DATA 20,20,20,3F,3A,5C,24
     ,44,69,73
GG 230 DATA 6B,20,61,63,63,65,73
     ,73,20,65
LF 240 DATA 72,72,6F,72,20,2D,20
     ,70,72,65
LC 250 DATA 73,73,20,61,6E,79,20
     ,6B,65,79
MP 260 DATA 20,74,6F,20,63,6F,6E
     ,74,69,6E
BK 270 DATA 75,65,2E,2E,2E,0A,0A
     ,0D,24,0A
KL 280 DATA 0A,0D,20,20,20,20,20
     ,20,20,20
AE 290 DATA 20,55,73,65,20,61,72
     ,72,6F,77
EK 300 DATA 20,6B,65,79,73,20,74
     ,6F,20,6D
KC 310 DATA 6F,76,65,2C,20,5B,44
     ,5D,20,74
MA 320 DATA 6F,20,64,65,6C,65,74
     ,65,20,66
PA 330 DATA 69,6C,65,73,2C,20,5B
     ,45,73,63
CF 340 DATA 5D,20,74,6F,20,71,75
     ,69,74,2E
KJ 350 DATA 24,8D,16,94,08,B4,09
     ,CD,21,B4
JD 360 DATA 07,CD,21,B4,0F,CD,10
     ,A2,43,07
EC 370 DATA 88,3E,44,07,B4,03,8A
     ,3E,44,07
OH 380 DATA CD,10,88,2E,45,07,88
     ,0E,46,07
JK 390 DATA B4,08,CD,10,88,26,4F
     ,07,B8,00
PN 400 DATA 06,B7,07,33,C9,BA,4F
     ,18,CD,10
NJ 410 DATA B8,00,02,33,DB,33,D2
     ,CD,10,B8
JM 420 DATA 00,00,A0,43,07,3C,07
     ,74,02,B0
BI 430 DATA 02,CD,10,C6,06,47,07
     ,00,B4,19
EJ 440 DATA CD,21,04,41,8D,36,DB
     ,08,83,C6
JO 450 DATA 16,88,04,B4,47,8D,36
     ,50,08,B2
ND 460 DATA 00,CD,21,8D,36,50,08
     ,8A,04,46
BA 470 DATA 3C,00,75,F9,4E,B0,0A
     ,88,04,46
CC 480 DATA 88,04,B0,0D,46,88,04
     ,46,B0,24
EJ 490 DATA 88,04,B4,09,8D,16,DB
     ,08,CD,21
AB 500 DATA 8D,16,50,08,CD,21,8D
     ,16,50,07
LA 510 DATA B4,1A,CD,21,8D,3E,03
     ,01,B4,4E
LC 520 DATA 8D,16,90,08,B9,00,00
     ,CD,21,72
EA 530 DATA 3F,FE,06,47,07,B9,11
     ,00,8D,1E
QO 540 DATA 50,07,83,C3,1E,49,8A
     ,17,80,FA
BK 550 DATA 00,74,0A,88,15,47,B4
     ,02,CD,21
QO 560 DATA 43,EB,EE,B4,02,B2,20
     ,CD,21,47
QC 570 DATA E2,FB,80,3E,47,07,64
     ,75,03,EB
BP 580 DATA 40,90,8D,16,90,08,B4
     ,4F,CD,21
GN 590 DATA 72,35,EB,C1,8D,16,F5
     ,08,B4,09
DE 600 DATA CD,21,B4,07,CD,21,B4
     ,00,A0,43
HP 610 DATA 07,CD,10,B4,05,A0,44
     ,07,CD,10
EC 620 DATA 8A,2E,45,07,8A,0E,46
     ,07,CD,10
PL 630 DATA B8,00,06,33,C9,BA,4F
     ,18,8A,3E
IL 640 DATA 4F,07,CD,10,C3,8D,16
     ,29,09,B4
OP 650 DATA 09,CD,21,C6,06,4D,07
     ,02,C6,06
EH 660 DATA 4C,07,02,C6,06,4B,07
     ,00,C6,06
MC 670 DATA 4A,07,00,C6,06,49,07
     ,00,E8,3A
JD 680 DATA 01,B4,07,CD,21,3C,00
     ,74,0D,3C
LA 690 DATA 1B,74,A9,0C,20,3C,64
     ,75,EE,E9
MJ 700 DATA B6,00,B4,07,CD,21,3C
     ,48,75,21
NG 710 DATA 80,3E,4D,07,03,72,DC
     ,8A,0E,4D
JM 720 DATA 07,FE,C9,88,0E,4C,07
     ,8A,0E,49
BN 730 DATA 07,80,E9,05,88,0E,48
     ,07,E8,FE
BN 740 DATA 00,EB,C2,3C,4B,75,20
     ,80,3E,4B
JF 750 DATA 07,00,74,B7,8A,0E,4B
     ,07,FE,C9
HK 760 DATA 88,0E,4A,07,8A,0E,49
     ,07,FE,C9
JG 770 DATA 88,0E,48,07,E8,DA,00
     ,EB,9E,3C
OG 780 DATA 50,75,2A,8A,16,47,07
     ,2A,16,49
EP 790 DATA 07,80,FA,05,77,02,EB
     ,8B,8A,0E
EA 800 DATA 4D,07,FE,C1,88,0E,4C
     ,07,8A,0E
CI 810 DATA 49,07,80,C1,05,88,0E
     ,48,07,E8
QK 820 DATA AD,00,E9,70,FF,3C,4D
     ,74,03,E9
NL 830 DATA 69,FF,80,3E,4B,07,04
     ,75,03,E9
EA 840 DATA 5F,FF,8A,16,47,07,2A
     ,16,49,07
LD 850 DATA 80,FA,01,77,03,E9,4F
     ,FF,8A,0E
QM 860 DATA 4B,07,FE,C1,88,0E,4A
     ,07,8A,0E
EG 870 DATA 49,07,FE,C1,88,0E,48
     ,07,E8,72
FE 880 DATA 00,E9,35,FF,E8,1F,00
     ,8A,04,3C
HA 890 DATA 20,75,03,E9,29,FF,8B
     ,D6,B4,41
GE 900 DATA CD,21,B0,20,B9,0C,00
     ,88,04,46
FC 910 DATA E2,FB,E8,50,00,E9,13
     ,FF,A0,49
KC 920 DATA 07,B4,00,D1,E0,D1,E0
     ,D1,E0,D1
HI 930 DATA E0,BE,03,01,03,F0,C3
     ,8A,16,4B
QO 940 DATA 07,D0,E2,D0,E2,D0,E2
     ,D0,E2,8A
LK 950 DATA 36,4D,07,B7,00,B4,02
     ,CD,10,C3
GO 960 DATA E8,D3,FF,E8,E3,FF,B7
     ,00,8A,1E
LO 970 DATA 4E,07,B9,01,00,B4,09
     ,8A,04,46
DH 980 DATA 3C,00,74,0A,CD,10,FE
     ,C2,B4,02
KI 990 DATA CD,10,EB,ED,C3,C6,06
     ,4E,07,07
NM 1000 DATA E8,D5,FF,C6,06,4E,0
     7,70,A0,4C
DK 1010 DATA 07,A2,4D,07,A0,4A,0
     7,A2,4B,07
BH 1020 DATA A0,4B,07,A2,49,07,E
     8,BB,FF,C3
```

Atari ST Menu Driver

Brian Flynn

Are your BASIC programs cluttering up your disks, hiding in folders, and straying beyond your file-window's view? If clicking scroll bars in an attempt to locate that elusive BASIC program is getting tiresome, try "Atari ST Menu Driver." This short program provides easy menu-driven access to all your ST BASIC programs.

"Atari ST Menu Driver" makes loading and running ST BASIC programs a lot easier. Extracted from the disk accompanying the COMPUTE! book *ST Applications*, written by Brian and John Flynn, the menu driver handles up to ten programs and works with either a monochrome monitor set to high resolution or a color monitor set to medium resolution.

Atari ST Menu Driver was written in modular style for ease of comprehension. Subroutines are identified by labels, such as SETSCREEN:, DRAW.MENU:, and the like. This feature of ST BASIC is very handy, and you may want to use it in some of your own projects.

This program also takes advantage of the ST's VDI (Virtual Device Interface) routines. These high-powered PEEK and POKE procedures enable the use of such tasks as reading the location of the mouse pointer, drawing boxes and octagons, and writing text to the screen. Each VDI routine is clearly labeled and is highly transportable to other programs.

Getting Started

As noted, Menu Driver can be used with either a monochrome or a color monitor. If you are using a color monitor, set the screen resolution to medium. Choose Set Preferences from the Options pull-down menu on the GEM desktop. You will be offered two choices: Low Resolution and Medium Resolution. Select the latter. Then load ST BASIC as usual.

If you are using a monochrome monitor, don't worry about screen resolution. It's automatically set to high, and it cannot be changed.

Now, type in and save the program.

Using The Program

By way of example, the program is currently set up to run ten BASIC programs found in the chapter on games in *ST Applications*. Upon running the menu driver, a list of program names appears, beginning with "Enigma" and ending with "Bunny's Tic-Tac-Toe." Assuming your disk contained these programs, all you would need to do is click the mouse to load and run a program.

To modify the driver to accommodate other sets of programs, follow these simple steps:

1. Choose a title for your group of programs, and place it in line 360. The program will automatically center your entry.

2. Indicate in line 670 the number of programs in your group. Up to ten programs are allowed.

3. At the end of the driver (lines 1850–onward), enter two names for each program:

- the full program title to be displayed on the screen.
- the actual filename, with .BAS included.

Finally, save your menu-driver with a suitable name like GAMES.BAS for games, or perhaps MONEY.BAS for a group of financial applications. For convenience, the menu driver should be saved on the same disk as the ST BASIC programs that it is set up to load.

If you follow the example set by the game programs already entered, you should have no problem customizing Atari ST Menu Driver for your own purposes. In fact, you may want to create a menu-driver for each category of BASIC programs that you have occasion to access.

For instructions on entering this program, please refer to "COMPUTE!'s Guide to Typing In Programs" elsewhere in this issue.

Atari ST Menu Driver

```
100   REM GAMES
110   GOSUB SETSCREEN
120   GOSUB TITLEBAR
130   GOSUB KEYVALUES
140   GOSUB HIDE.ARROW
150   GOSUB DRAW.MENU
160   GOSUB RETURN.BAR
170   GOSUB COPYRIGHT
180   GOSUB SHOW.ARROW
190   GOSUB CHOOSE
200   GOSUB GOODBYE
210   CHAIN TITLE.SHORT$(PIC
      K%)
220   '
230   SETSCREEN:
240   DEFINT A-Z
250   ST = 3-PEEK(SYSTAB): R
      EM SCREEN (1=COLOR, 2=MC
      HROME)
260   '    COLORS (WHITE, YEL
      LOW, ORANGE, BLACK)
270   DATA 1911,1874,1825,0
280   FOR I=0 TO 3: READ KOL
      OR(I): NEXT
290   LC# = 1114: POKE LC#,V
      ARPTR(KOLOR(0))
300   '    ADD-FACTOR FOR COL
      OR SCREEN
310   IF ST = 1 THEN CS = 3
```

```
      ELSE CS = 0
320   RETURN
330   TITLEBAR:
340   FULLW 2: CLEARW 2
350   COLOR 3
360   T$ = "Games of Skill"
370   T$ = SPACE$(2) + T$ +
      SPACE$(2)
380   AT$ = CHR$(14)+CHR$(15
      )
390   T$ = SPACE$(4) + AT$ +
       T$ + AT$ + SPACE$(4)
400   Y=15*ST+4: CN$ = "ON":
       GOSUB TEXT.WRITE
410   RETURN
420   TEXT.WRITE:
430   LN = LEN(T$)
440   IF CN$ = "ON" THEN X =
       312 - LN*8/2
450   POKE CONTRL,8: POKE CO
      NTRL+2,1: POKE CONTRL+6,
      LN
460   FOR Q = 1 TO LN
470   POKE INTIN + (Q-1)*2,A
      SC(MID$(T$,Q,1))
480   NEXT Q
490   POKE PTSIN,X: POKE PTS
      IN+2,Y: VDISYS(0)
500   RETURN
510   KEYVALUES:
520   DIM TITLE.LONG$(10), T
      ITLE.SHORT$(10)
530   '    OCTAGONS
540   DATA -4,-3,-6,-1,-6,1,
      -4,3,4,3,6,1,6,-1,4,-3,-
      4,-3
550   DATA -3,-6,-6,-3,-6,3,
      -3,6,3,6,6,3,6,-3,3,-6,-
      3,-6
560   FOR I=1TO 9: READ XG(I
      ),YG(I): NEXT
570   FOR I=1TO 9
580   READ X,Y
590   IF ST = 2 THEN XG(I) =
       X: YG(I) = Y
600   NEXT
610   '    INDEX & STYLE
620   DATA 1,1,4,2
630   FOR I=1TO 2
640   READ INDEX(I),STYLE(I)
650   NEXT
660   '    PROGRAM NAMES
670   N = 10: REM NUMBER OF
      PROGRAMS
680   FOR I=1TO N
690   READ TITLE.LONG$(I), T
      ITLE.SHORT$(I)
700   NEXT
710   BELL$ = CHR$(7)
720   RETURN
730   DRAW.MENU:
740   COLOR 1,2,1,INDEX(ST),
      STYLE(ST)
750   X1=110: Y1=5*ST: X2=51
      0: Y2=140*ST: GOSUB BAR
760   X1=140: Y1=15*ST: X2=4
      80: Y2=130*ST: GOSUB BAR
770   FILL 310,10*ST
780   PICK% = 1: P = 1: CN$
      = "OFF"
790   FOR I=1TO N
800   X = 250: Y = (10*I+40)
      *ST+CS
810   T$ = TITLE.LONG$(I)
820   GOSUB TEXT.WRITE
830   IF I = PICK% THEN Z =
      9 ELSE Z = 6
840   CHOICE = I: GOSUB DRAW
      .SHAPE
850   NEXT I
860   RETURN
870   BAR:
880   LINEF X1,Y1,X1,Y2: LIN
      EF X1,Y2,X2,Y2
890   LINEF X2,Y2,X2,Y1: LIN
      EF X2,Y1,X1,Y1
900   RETURN
910   DRAW.SHAPE:
920   X = 220: Y = (10*CHOIC
      E+37)*ST + CS
930   POKE CONTRL,Z: POKE CO
      NTRL+2,9: POKE CONTRL+6,
      0
940   CNT = 0
950   FOR J = 1TO 9
960   POKE PTSIN+CNT,X+XG(J)
      : POKE PTSIN+CNT+2,Y+YG(
      J)
970   CNT = CNT + 4
980   NEXT J
990   VDISYS(0)
1000  RETURN
1010  RETURN.BAR:
1020  COLOR 1,1,1,1,1
1030  X1=270: Y1=144*ST: X2=
      350: Y2=154*ST: GOSUB BA
      R
1040  FILL 310,150*ST
1050  V = 1: GOSUB LETTERING
1060  V = 3: GOSUB WRITE.MOD
      E
1070  CN$ = "ON": Y = 171*ST
      +CS
1080  T$ = "Return": GOSUB T
      EXT.WRITE
1090  V = 0: GOSUB LETTERING
1100  V = 1: GOSUB WRITE.MOD
      E
1110  PRINT BELL$;
1120  RETURN
1130  LETTERING:
1140  POKE CONTRL,106: POKE
      CONTRL+2,0: POKE CONTRL+
      6,1
1150  POKE INTIN,V: VDISYS(0
      )
1160  RETURN
1170  WRITE.MODE:
1180  POKE CONTRL,32: POKE C
      ONTRL+2,0: POKE CONTRL+6
      ,1
1190  POKE INTIN,V: VDISYS(0
      )
1200  RETURN
1210  COPYRIGHT:
1220  Y = 183*ST+CS: COLOR 3
1230  T$ = "Copyright 1986,
      COMPUTE! Publications, I
      nc."
1240  GOSUB TEXT.WRITE
1250  RETURN
1260  CHOOSE:
1270  GOSUB CLICKIT
1280  IF X>270 AND X<350 AND
       Y>143*ST AND Y<155*ST T
      HEN 1350
1290  V = INT((Y-13*ST)/(10*
      ST))
1300  IF X>200 AND X<420 AND
       V > 0 AND V <= N THEN P
       = V
1310  GOSUB CHANGE.CHOICE
1320  PICK% = P
1330  GOSUB GURGLE
1340  GOTO CHOOSE
1350  RETURN
1360  CLICKIT:
1370  MOUSE = 0
1380  WHILE MOUSE = 0
1390  POKE CONTRL,124: POKE
      CONTRL+2,0: POKE CONTRL+
      6,0
1400  VDISYS(0)
1410  X=PEEK(PTSOUT): Y=PEEK
      (PTSOUT+2)
1420  Y = Y-19*ST-CS
1430  MOUSE = PEEK(INTOUT)
1440  WEND
1450  RETURN
1460  CHANGE.CHOICE:
1470  GOSUB HIDE.ARROW
1480  COLOR 1,0: Z = 9: CHOI
      CE = PICK%: GOSUB DRAW.S
      HAPE
1490  Z = 6: GOSUB DRAW.SHAP
      E
1500  COLOR 1,2: Z = 9: CHOI
      CE = P: GOSUB DRAW.SHAPE
1510  GOSUB SHOW.ARROW
1520  RETURN
1530  HIDE.ARROW:
1540  POKE CONTRL,123: POKE
      CONTRL+2,0: POKE CONTRL+
      6,0
1550  VDISYS(0)
1560  RETURN
1570  SHOW.ARROW:
1580  POKE CONTRL,122: POKE
      CONTRL+2,0: POKE CONTRL+
      6,1
1590  POKE INTIN,0: VDISYS(0
      )
1600  RETURN
1610  GURGLE:
1620  NOTE = 4
1630  FOR G=1TO 5
1640  NOTE = 6 - NOTE
1650  SOUND 1,8,NOTE,4,1
1660  NEXT G
1670  SOUND 1,0
1680  RETURN
1690  GOODBYE:
1700  '    RESET COLORS
1710  CLEARW 2
1720  KOLOR(1) = 1570: KOLOR
      (2) = 609
1730  POKE LC#, VARPTR(KOLOR
      (0))
1740  '    CLEAR TITLE BAR
1750  CN$ = "ON": Y = 15*ST+
      4
1760  T$ = SPACE$(22): GOSUB
       TEXT.WRITE
1770  '    DISPLAY MESSAGE
1780  COLOR 1
1790  Y = 90*ST: Q$ = CHR$(3
      4)
1800  T$ = "Loading " + Q$ +
       TITLE.LONG$(PICK%) + Q$
1810  GOSUB TEXT.WRITE
1820  RETURN
1830  '
1840  REM PROGRAM NAMES
1850  DATA Enigma, ENIGMA.BA
      S
1860  DATA "Elementary, Wats
      on", WATSON.BAS
1870  DATA Knights Errant, K
      NIGHTS.BAS
1880  DATA Pharaoh's Pyramid
      , PYRAMID.BAS
1890  DATA Roman Checkers, R
      OMAN.BAS
1900  DATA Falstaff, FALSTAF
      F.BAS
1910  DATA Mosaic Puzzle, MO
      SAIC.BAS
1920  DATA Hi-Q, HIQ.BAS
1930  DATA Solitaire Checker
      s, SOLTRE.BAS
1940  DATA Bunny's Tic-Tac-T
      oe, TTT.BAS
```

NRI Trains You At Home—As You Build Your Own IBM-Compatible Computer

GET THE KNOW-HOW TO SERVICE EVERY COMPUTER ON THIS PAGE

Learn the Basics the NRI Way—and Earn Good Money Troubleshooting Any Brand of Computer

The biggest growth in jobs between now and 1995, according to Department of Labor estimates, will occur in the computer service and repair business, where demand for trained technicians will actually *double*.

You can cash in on this opportunity—either as a full-time corporate technician or an independent service-person—once you've learned all the basics of computers the NRI way. NRI's practical combination of "reason-why" theory and "hands-on" building skills starts you with the fundamentals of electronics, then guides you through advanced electronic circuitry and on into computer electronics. You also learn to program in BASIC and machine language, the essential languages for troubleshooting and repair.

IBM is a Registered Trademark of IBM Corporation.

Epson is a Registered Trademark of Epson America, Inc.

Apple and the Apple logo are Registered Trademarks of Apple Computer, Inc.

Compaq is a Registered Trademark of COMPAQ Computer Corporation.

Total Computer Systems Training, Only From NRI

No computer stands alone . . . it's part of a total system. To really service computers, you have to understand computer *systems*. And only NRI includes a powerful computer system as part of your training, centered around the new fully IBM compatible Sanyo 880 Series computer.

You start with the step-by-step assembly of the new, highly rated fully IBM compatible Sanyo 880 Series computer. You install and troubleshoot the "intelligent" keyboard. Then you assemble the power supply, install the disk drive, and add extra memory to give you a powerful 256K RAM system. The new 880 computer has two operating speeds: standard IBM speed of 4.77 MHz and a remarkable turbo speed of 8 MHz, making it almost twice as fast as the IBM PC. Next, you'll interface the high-resolution monitor and begin to use the valuable software also included with your complete computer system.

AND MORE!

It all adds up to confidence-building, real-world experience that includes training in programming, circuit design, and peripheral maintenance. You'll be learning about, working with, servicing, and troubleshooting an entire computer system—monitor, keyboard, computer, disk drive, power supply—to ensure that you have all the essential skills you need to succeed as a professional computer service technician.

No Experience Needed, NRI Builds It In

This is the kind of practical, hands-on experience that makes you uniquely prepared, with the skills and confidence you need for success. You learn at your own convenience in your own home. No classroom pressures, no night school, no need to quit your present job until you're ready to make your move. Your training is backed up by your personal NRI instructor and the NRI technical staff, ready to answer your questions and help you when you need it. You get it all with NRI at-home training.

100-Page Free Catalog Tells More

Send the postage-paid reply card today for NRI's big, 100-page, color catalog on NRI's electronics training, which gives you all the facts about NRI courses in Microcomputers, Robotics, Data Communications, TV/Audio/Video Servicing, and other growing high-tech career fields. If the reply card is missing, write to the address below.

Your NRI total systems training includes: • NRI Discovery Lab® to design and modify circuits • Your four-function digital multimeter with walk-you-through instructions on audio tape • Digital logic probe for visual examination of keyboard circuits • The newest Sanyo 880 Series Computer with "intelligent" keyboard and 360K double-density, double-sided disk drive • High resolution monochrome monitor • 8K ROM, 256K RAM • Bundled software including GW BASIC, MS-DOS, WordStar, CalcStar • Reference manuals, schematics, and bite-sized lessons.

NRI SCHOOLS

McGraw-Hill Continuing Education Center
3939 Wisconsin Avenue, NW
Washington, DC 20016

We'll Give You Tomorrow.

Smooth Out For The Atari

James Hague

This sophisticated graphics technique lets you smooth out the rough lines of your graphics screens in mere seconds. For the Atari 400, 800, XL, and XE with 48K or more of RAM.

One of the strangest of the many graphics modes of the Atari eight-bit computers is GRAPHICS 9. It allows as many as 16 different shades of each color. That's twice the number of intensities available on the Atari ST. But that doesn't mean that you can get ST-quality pictures on the eight-bit machines—the resolution simply isn't high enough. While an ST's "low-resolution" screen offers 320 × 200 pixels, GRAPHICS 9 is limited to 80 × 192 pixels.

When you draw on such a low-resolution screen, you're likely to see *jaggies*, the stairstep effects you see on lines that are drawn at an angle (nonhorizontal or nonvertical lines). One common method used to eliminate jaggies is *antialiasing*, a complex smoothing method used in advanced paint programs such as *Deluxe Paint II*. In "Smooth Out," a related method—*pixel averaging*—is used to smooth the entire screen at once.

Pixel averaging smooths pictures by blurring them. As a result, the process is better suited to landscapes and portraits than to bar charts and boxes.

Typing It In

Type in Program 1 and save it to tape or disk. Do not run the program yet—it is not a complete program until you add the lines from either Program 2 or Program 3.

Program 2 is a very short demonstration of Smooth Out. Load Program 1, type in the lines from Program 2 and then save the resulting program with a unique filename. Program 3 draws an alien landscape. Follow the same procedure with Program 3 as you did with Program 2.

When you're ready to try one of the sample programs, load it and type RUN. A picture is drawn on the screen. When the picture is finished, you'll hear a tone. Press START, and Smooth Out will smooth the screen with a machine language subroutine. Press START again to end the program.

Behind The Scene

Pixel averaging can be done in many ways. The smoothing algorithm used here adds up the values of the surrounding eight pixels and divides the total by 8 to find the new value of the pixel. This must be done for each pixel on the screen. In machine language, the task takes seconds. If I had used BASIC, it would have taken more than 30 minutes.

If you want to use this program with your own graphics screens, insert the lines to draw your GRAPHICS 9 screen between lines 100 and 5000 of Program 1. (Be sure not to remove line 100.)

For an interesting effect, try changing the GRAPHICS 9 statements in the programs to GRAPHICS 11. (See lines 100 and 5040.)

For instructions on entering these programs, please refer to "COMPUTE!'s Guide to Typing In Programs" elsewhere in this issue.

Program 1: Smooth Out

```
BG 90 REM COPYRIGHT 1987 COM
      PUTE! PUBLICATIONS, IN
      C.  ALL RIGHTS RESERVE
      D.
JI 92 PRINT "{CLEAR}COPYRIGH
      T 1987":PRINT "COMPUTE
      ! PUBLICATIONS, INC.":
      PRINT "ALL RIGHTS RESE
      RVED."
FE 94 FOR X=1 TO 1500:NEXT X
LN 96 PRINT "{CLEAR}"
HB 100 MEMTOP=PEEK(106):GRAP
      HICS 9
CJ 5000 IF PEEK(1540)<>205 A
        ND PEEK(1542)<>39 TH
        EN RESTORE 5200:FOR
        A=0 TO 153:READ B:PO
        KE 1536+A,B:NEXT A
AM 5010 GOSUB 5100
KP 5020 POKE 203,PEEK(88):PO
        KE 204,PEEK(89)
GA 5030 POKE 106,PEEK(106)-4
        8
FD 5040 GRAPHICS 9
GC 5050 A=USR(1536)
BB 5060 GOSUB 5100
EL 5070 POKE 106,MEMTOP:GRAP
        HICS 0
KE 5080 END
GC 5100 FOR A=15 TO 0 STEP -
        1:SOUND 0,100,10,A:N
        EXT A
HM 5110 IF PEEK(53279)<>6 TH
        EN 5110
KI 5120 RETURN
OH 5200 DATA 104,169,0,133,2
        05,160
AC 5210 DATA 39,177,203,145,
        88,136
PE 5220 DATA 16,249,160,40,1
        77,203
AF 5230 DATA 145,88,160,79,1
        77,203
AD 5240 DATA 145,88,169,81,1
        33,206
JC 5250 DATA 169,0,133,207,1
        62,7
MB 5260 DATA 165,206,24,125,
        146,6
ND 5270 DATA 74,144,8,168,17
        7,203
```

```
DI 5280 DATA 41,15,76,60,6,1
        68
KG 5290 DATA 177,203,74,74,7
        4,74
BD 5300 DATA 24,101,207,133,
        207,202
PB 5310 DATA 16,224,165,206,
        74,144
KD 5320 DATA 9,168,165,207,7
        4,74
HK 5330 DATA 74,76,88,6,168,
        165
IP 5340 DATA 207,10,41,240,1
        7,88
CO 5350 DATA 145,88,230,206,
        165,206
GA 5360 DATA 201,159,208,186
        ,165,203
DN 5370 DATA 24,105,40,133,2
        03,165
OL 5380 DATA 204,105,0,133,2
        04,165
JO 5390 DATA 88,24,105,40,13
        3,88
JK 5400 DATA 165,89,105,0,13
        3,89
EM 5410 DATA 230,205,165,205
        ,201,190
CE 5420 DATA 208,136,160,40,
        177,203
PM 5430 DATA 145,88,200,192,
        80,208
EF 5440 DATA 247,96,175,176,
        177,255
NO 5450 DATA 1,79,80,81
```

Program 2: Smooth Out Demo

```
DC 110 FOR A=1 TO 25
LL 120 X=INT(RND(0)*57)+1
OH 130 Y=INT(RND(0)*150)+1
EG 140 COLOR INT(RND(0)*16)
OE 150 FOR B=X TO X+19
KI 160 PLOT X+10,Y:DRAWTO B,
       Y+39
ND 170 NEXT B:NEXT A
```

Program 3: Alien Landscape

```
CL 110 COLOR 12:PLOT 0,0:DRA
       WTO 79,0:DRAWTO 79,19
       1:DRAWTO 0,191:DRAWTO
       0,0
PO 120 FOR A=1 TO 2
DF 130 IF A=1 THEN X=30:Y=30
       :GOTO 150
MI 140 X=55:Y=40
DP 150 FOR B=1 TO 78
NL 160 COLOR INT(RND(0)*5)+2
       :PLOT X,Y:DRAWTO B,13
       5
ND 170 NEXT B:NEXT A
OI 180 FOR A=130 TO 190:COLO
       R INT(RND(0)*7)+5:PLO
       T 1,A:DRAWTO 39,A-5:D
       RAWTO 78,A:NEXT A
AF 190 FOR A=1 TO 2
DN 200 IF A=1 THEN X=55:Y=80
       :GOTO 220
NC 210 X=39:Y=95
KL 220 FOR B=1 TO 78 STEP 2
AI 230 COLOR INT(RND(0)*10)+
       4:PLOT X,Y:DRAWTO B,1
       90
NB 240 NEXT B:NEXT A
DH 250 FOR A=1 TO 25
KM 260 COLOR INT(RND(0)*10)+
       5:PLOT 1,70:DRAWTO A,
       190:NEXT A
```


Nim:
The Ultimate Binary Game

Jim Butterfield, Associate Editor

The best way to beat the computer in this classic strategy game is to know how to convert numbers from decimal to binary. The accompanying program was written in Commodore BASIC and runs on any Commodore eight-bit computer. With minor modifications, it could also be adapted to work with other versions of Microsoft BASIC.

"Nim" is one of the simplest games ever invented, yet successful play requires at least an intuitive grasp of binary numbers, the system used by all digital computers.

Here's how Nim is played. You take a bunch of objects (toothpicks, matches, coins, whatever) and arrange them randomly in three or more piles. You and your opponent alternate taking turns. During a turn, a player may take as many objects as he or she wishes from any one pile (but *only* from one pile). The player taking the last object wins.

That's a description of the standard game of Nim. Several variations exist. In one variation, the player taking the last object loses. In another, there is a maximum number of objects that may be taken during a turn. The program presented here plays all three versions of the game, switching the rules from one game to the next. Type the program in and save a copy before you run it. The program should run as listed on any eight-bit Commodore computer, and with minor modifications on any eight-bit computer.

Simple Play Theory

The classic game of Nim (last object wins, pick any number) has a very elegant playing strategy that requires knowledge of binary numbers. (If you are not familiar with the binary numbering system, see "From Decimal to Binary," accompanying this article.)

Count the number of objects in each pile, and write down each number, one above the other, in binary. For example, if there are three piles of objects, containing three, four, and five objects, respectively, you would write

```
3      1  1
4   1  0  0
5   1  0  1
```

Note that the binary numbers are lined up on the right side, just as

we would arrange conventional numbers. Now, you should ask, does every column (not row) have an *even* number of 1's in it?

If the answer is yes, you're stuck. The best you can do is make some random move and hope your opponent stumbles when he or she plays.

But if the answer is no, you have a winning play. The play is to take from a pile in such a way so that all columns have *even parity*—an even number of 1's.

Let's look at the example given above. The right column has two bits set, so that's even parity. The middle column has only one bit set. That's an odd number, so you have a winning play.

It takes some time, at first, to examine the possible moves. In this case, it turns out there's only one move that produces even parity. Here's the move: Take two from the first column, leaving

```
1          1
4   1   0  0
5   1   0  1
```

Examine the columns of the binary numbers, and you'll see that they all contain an even number of bits (zero or two in this case). Your opponent now has no satisfactory play.

Let's carry this game through to its conclusion. Suppose your opponent takes four objects from the largest column, leaving one. Line up the numbers again:

```
1          1
4   1   0  0
1          1
```

Your play to restore even parity is obvious. Grab the entire pile of four to leave:

```
1   1
0   0
1   1
```

What can your opponent do now? Not much. On the next turn, he or she must take the lone piece from either one of the piles, after which you will take the last object from the remaining pile and win.

Variations

The classic game has a clear and easy strategy. The task becomes more complex when we limit the number of objects that may be taken on each play. Extra difficulties arise when we decide that the player taking the last object will lose instead of win. But the basic game strategy remains, built upon a foundation of binary numbers.

I won't go into the extra theory and strategy here. If you're interested, you can examine the program to see what makes it such a good player.

The Good And The Bad

You might find it dull to play against a computer that wins every time, so the computer has been given an IQ. The computer asks at the beginning of the game if it should play the best it can. If you you reply *N* for no, it will sometimes make mistakes, giving even an inexperienced player a chance to win.

Even if you don't know binary, you can become skilled at this game—you'll begin to spot winning combinations. Be forewarned: Every time the computer loses a game, it becomes a better player. So the next game may not be as easy.

Eventually, if you master the theory of play, you will be able to beat the computer every time. That's because the computer, after setting up a random board, asks you whether you want the first move. If you have a good play, take the first move and make it. If you don't seem to have a good move, pass the first move to the computer.

Program Notes

One odd expression in the program implements *exclusive OR*. Unlike the standard OR operator, exclusive OR is true only when one operand is true and the other is false (the standard OR is also true when both operands are true). Exclusive OR can be simulated with the AND, OR, and NOT operators:

```
X = (A OR B) AND NOT(A AND B)
```

If your specialty is machine language, the 6502 processor has an exclusive OR (EOR) command built in.

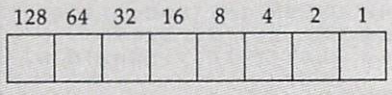
If you are interested in figuring out how the program works, you'll need to understand the roles of some variables. R is the playing rule—if it is 1, the last player wins, if it is 0, the last player loses. N9 is the maximum number of objects you are allowed to pick up. If you are allowed to choose any number of objects, N9 is set to 99.

From Decimal To Binary

The binary numbering system is used by computers because each bit—the smallest unit of storage in a computer's memory—can have only two states, *0* or *1*. These two states correspond to the *off* and *on* states of the electronic switches that make up the brain of the computer. Just as it is easy for us to do decimal (base 10) math with our fingers, so it is easy for computers to do binary (base 2) math.

The binary and decimal systems are just two different ways of looking at the same thing: numeric quantities. Instead of saying, "Look! There are seven (base 10) trees," we could just as correctly state, "Look! There are 111 (base 2) trees." The trick is learning how to convert from one base to another.

Perhaps the simplest way to convert from base 10 to base 2 is to construct a value box (at first on paper, but eventually in your head). Here's a value box that will convert numbers from 0 to 255 in decimal.



Now let's pick a number to be converted. We'll use 20.

Going from left to right, find the first number in the value box that is less than or equal to 20. In this case, the number is 16. Color in the box under the number 16.

We have represented 16 of the 20 objects that we wish to represent. Subtract 16 from 20—we have 4 more objects to represent. Continue scanning across from left to right. We pass 8, but since 4 is less than or equal to 4, we color in the box under the 4. When we subtract 4 from 4, we get 0, so we have now completed our conversion. All of the empty boxes represent 0s, and all of the filled boxes represent 1s. The result is 00010100. We can write binary numbers without leading zeros (those to the left of the first significant digit), so decimal 20 is 10100 in binary.

The array A( ) holds the playing board. P( ) is the original (previous) board, in case you want to play the same game over again. S( ) is a scrambling array, to give the computer's strategy a little variety. If the computer player has more than one possible winning move, it might pick either one, depending on the contents of S( ).

Playing Against Humans

If you play this game against another person, remember that psychology is an important part of your playing style. You should make your moves without hesitation if possible. This is especially true when you don't have a winning move—you don't want to tip off your opponent that you may be in trouble.

And don't take a piece of paper and start writing out numbers in binary. Learn to do it in your head. It's a good step to computer literacy.

Nim

For instructions on entering this program, please refer to "COMPUTE!'s Guide to Typing In Programs" elsewhere in this issue.

```
XQ 100 PRINT"{CLR}{2 DOWN}":PR
       INT TAB(17)"{RVS} NIM "
KE 110 J=RND(-TI)
ED 120 DIM A(5),P(5),S(5),N(5)
JK 130 FOR J=1 TO 4:S(J)=J:NEX
       T J
ME 140 H$="{HOME}{13 DOWN}"
EG 150 C$=H$+"{35 SPACES}"+H$
EP 160 PRINT CHR$(8);CHR$(142)
RH 170 PRINT"RULES:"
MG 180 PRINT"{2 SPACES}EACH PL
       AYER PICKS AS MANY ITEM
       S"
SG 190 PRINT"{2 SPACES}AS DESI
       RED FROM ANY ONE ROW."
HD 200 PRINT
GF 210 PRINT"THE WINNER IS THE
        PLAYER WHO PICKS"
RB 220 PRINT"THE LAST ITEM."
PG 230 PRINT
ED 240 PRINT"WE EACH PLAY IN T
       URN."
DH 250 PRINT
PX 260 X$="N":INPUT"SHOULD I P
       LAY MY BEST GAME";X$
QE 270 I=.5:IF X$="Y" OR X$="Y
       ES" THEN I=1
HK 280 PRINT
CD 290 IF I=1 THEN PRINT"PUNY
       {SPACE}HUMAN!{2 SPACES}
       YOU HAVEN'T A HOPE."
XX 300 IF I<1 THEN PRINT"MAYBE
        I'LL LET YOU WIN ONE O
       R TWO GAMES.
FC 310 FOR J=1 TO 500:NEXT
RK 320 R=1:N9=99
CB 330 T=0
HQ 340 FOR J=1 TO 4
XC 350 A(J)=INT(RND(1)*7):T=T+
       A(J)
XR 360 NEXT J
PS 370 IF T<4 THEN 330
EB 380 FOR J=1 TO 4
BA 390 P(J)=A(J):K=INT(RND(1)*
       4)+1
DG 400 T=S(K):S(K)=S(J):S(J)=T
GX 410 NEXT J
BG 420 M=0:M0=0
MH 430 PRINT"{CLR}{DOWN}PLAYER
        TAKING LAST ITEM ";
SS 440 IF R=0 THEN PRINT"LOSES
       "
DB 450 IF R=1 THEN PRINT"WINS"
CE 460 PRINT
GF 470 IF N9=99 THEN PRINT"PIC
       K AS MANY AS YOU LIKE"
EQ 480 IF N9<99 THEN PRINT"YOU
        MAY PICK NO MORE THAN"
       ;N9
SH 490 PRINT
XJ 500 R9=0:T6=0:T8=0:T9=0
SG 510 FOR J=1 TO 4
EB 520 PRINT CHR$(J+64);": ";
FQ 530 FOR K=1 TO 9
PR 540 C=209:IF K>A(J) THEN C=
       32
PF 550 PRINT CHR$(C);" ";
MM 560 NEXT K
JE 570 PRINT:PRINT
RM 580 K=S(J)
MB 590 N=INT(A(K)/(N9+1)):N(K)
       =A(K)-N*(N9+1)
KJ 600 IF N(K)=1 THEN T8=T8+1:
       T3=K
QF 610 IF N(K)>1 THEN T9=T9+1:
       T4=K
DF 620 IF A(K)>T6 THEN T6=A(K)
       :T5=K
DR 630 NEXT J
BB 640 IF M>0 THEN 680
PE 650 PRINT C$;
JF 660 X$="N":INPUT"DO YOU WAN
       T THE FIRST MOVE";X$
BM 670 IF X$="Y" OR X$="YES" T
       HEN M0=1
RF 680 M=M+1:IF M0=1 THEN 900
QJ 690 T=0
EM 700 IF R=1 THEN 770
KC 710 IF T9=0 AND T8=0 THEN 8
       00
QC 720 J1=T3:IF T9>0 THEN J1=T
       4
JA 730 X=N(J1):IF T8=INT(T8/2)
       *2 THEN X=X-1
AX 740 IF X=0 THEN 770
BJ 750 IF T9<2 THEN 870
BE 760 REM CONVENTIONAL ANALYS
       IS
RK 770 FOR J=1 TO 4
KJ 780 T=(T OR N(J)) AND NOT(T
        AND N(J))
BJ 790 NEXT J
JR 800 J1=T5:X=T6:IF X>N9 THEN
        X=N9
HC 810 IF T=0 THEN 870
EM 820 FOR J=1 TO 4
KK 830 K=S(J)
BM 840 T0=(T OR N(K)) AND NOT(
       T AND N(K))
EF 850 IF T0<N(K) AND (N(K)-T0
       )<N9 THEN J1=K:X=N(K)-T
       0
AQ 860 NEXT J
KX 870 IF RND(1)>I THEN X=INT(
       RND(1)*T6)+1:J1=T5:IF X
       >N9 THEN X=N9
BK 880 PRINT C$;"I TAKE";X;"FR
       OM ROW ";CHR$(J1+64)
GP 890 GOTO 1000
EE 900 PRINT C$;"CHOOSE A ROW
       {SPACE}(OR {RVS} G
       {OFF} TO GIVE UP):";
XE 910 GET X$:IF X$="" THEN 91
       0
KX 920 IF X$="G" THEN M0=R:GOT
       O 1140
HC 930 J1=ASC(X$)-64:IF J1<1 O
       R J1>4 THEN 900
HG 940 IF A(J1)=0 THEN 900
FS 950 PRINT C$;"HOW MANY FROM
        ROW ";CHR$(64+J1);": "
       ;
BX 960 GET X$:IF X$="" THEN 96
       0
BB 970 X=ASC(X$)-48:IF X<0 OR
       {SPACE}X>9 THEN 960
GP 980 IF X=0 OR X>A(J1) OR X>
       N9 THEN PRINT"????":FOR
       J=1 TO 500:NEXT:GOTO 9
       00
CK 990 PRINT X
EE 1000 PRINT"{HOME}{2 DOWN}"
FJ 1010 FOR K=1 TO J1:PRINT:PR
       INT:NEXT K
CX 1020 PRINT TAB(3);
BS 1030 FOR K=1 TO X:PRINT"- "
       ;:NEXT K
JK 1040 A(J1)=A(J1)-X
GG 1050 T=0
PX 1060 FOR J=1 TO 4
FR 1070 IF A(J)<>0 THEN T=1
HS 1080 NEXT J
PM 1090 M0=1-M0
PK 1100 IF T=0 THEN 1140
GX 1110 FOR J=1 TO 750:NEXT J
MB 1120 PRINT"{HOME}{4 DOWN}"
AD 1130 GOTO 500
JC 1140 PRINT C$;
DR 1150 IF R=M0 THEN W0=W0+1:P
       RINT"I WIN!"
GK 1160 IF R<>M0 THEN W1=W1+1:
       PRINT"YOU WIN!"
JH 1170 PRINT
MG 1180 PRINT"THAT MAKES";W1;"
       GAME";
DX 1190 IF W1<>1 THEN PRINT"S"
       ;
PC 1200 PRINT" FOR YOU"
CD 1210 PRINT"AND";W0;"FOR ME.
       "
BG 1220 IF I<1 AND W1>W0 AND R
       <>M0 AND R1=0 THEN GOS
       UB 1380
MJ 1230 PRINT
JP 1240 R1=0
GQ 1250 X$="Y":INPUT"PLAY AGAI
       N";X$
QB 1260 IF X$="N" OR X$="NO" T
       HEN END
SC 1270 X$="N":INPUT"SAME GAME
        AS LAST TIME";X$
QJ 1280 IF X$<>"N" AND X$<>"NO
       " THEN 1350
CD 1290 IF RND(1)>.35 THEN 330
SM 1300 IF RND(1)>.6 THEN 1330
CC 1310 R=1-R:R9=5:PRINT:PRINT
       "{RVS} RULE CHANGE "
SG 1320 FOR J=1 TO 500:NEXT J
CG 1330 IF RND(1)<.6 THEN N9=I
       NT(RND(1)*4)+3:IF N9>5
        THEN N9=99
QB 1340 GOTO 330
CC 1350 R1=1
JC 1360 FOR J=1 TO 4:A(J)=P(J)
       :NEXT J
MD 1370 GOTO 420
RC 1380 I=I↑.75
HG 1390 PRINT"DON'T FEEL TOO S
       MART."
SE 1400 PRINT"I CAN DO BETTER!
       "
RK 1410 RETURN
```

Double-Duty DOS

Jason Coleman

The Apple II has two popular operating systems, DOS 3.3 and ProDOS. If you've ever been trapped between them, you'll appreciate this utility—it adds new commands to let you read and write DOS 3.3 files from within ProDOS, and move ProDOS files to DOS 3.3.

Most Apple users have two stacks of disks—one stack ProDOS and one stack DOS 3.3. That's not a problem until you want to use a program in one operating system that's written for the other one. Some people convert all of their 3.3 files to ProDOS format with a conversion utility. For the user with a large number of DOS 3.3 disks, this can be a trying experience.

"Double-Duty DOS" offers a different approach—it lets you view DOS 3.3 catalogs and load and save BASIC and machine language programs, all from within ProDOS.

Since this program is a ProDOS utility, be sure to boot with ProDOS when you are ready to type it in. The program must be entered with the Apple version of the "MLX" machine language entry program, which can be found elsewhere in this issue. When MLX asks for a starting and ending address, respond with these values:

STARTING ADDRESS? 3000
ENDING ADDRESS? 3A17

Type in the program. Before leaving MLX, be sure to save a copy to disk.

A Door To The Old World

Let's give Double-Duty DOS a test drive. Boot up in ProDOS and BRUN the program. In addition to the normal commands that are available in ProDOS, you have several new ones designed especially for accessing DOS 3.3 disks.

Here is a list of the new commands:

DCAT *,Ss ,Dd*
DLOAD filename *,Amemloc ,Ss ,Dd*
DSAVE filename *,Ss ,Dd*
DSAVE filename, A$hhhh, **L**$hhhh *,Ss ,Dd*

Parameters in *italics* are optional. The *s* stands for slot number, and *d* is for drive number. Place a DOS 3.3 disk in the drive and type DCAT. You'll see a list of all the files on the disk. You can then load any of these DOS 3.3 files using the DLOAD command, and save them to a ProDOS disk with the standard SAVE or BSAVE commands.

In the DLOAD command, the load address can be specified as a decimal or hexadecimal number. Precede hexadecimal values with a $. For example, DLOAD TEST,S6,D0,A$900 loads a DOS 3.3 binary file named TEST into location $900 from slot 6, drive 0.

You can also move files from ProDOS disks to DOS 3.3 disks. Simply load a program from a ProDOS disk and save it to a DOS 3.3 disk with the DSAVE command. Because of a limitation in DOS 3.3, you cannot save programs larger than 122 sectors (30K).

While the DLOAD command mimics both the normal LOAD command and the BLOAD commands, DSAVE has two different syntaxes. The first form listed above is for use with BASIC files, similar to SAVE. The second form is for binary files, and is used like the BSAVE command. Note that in the second form, the address and length parameters must both be specified in hexadecimal (preceded with a $).

Double-Duty DOS comes in handy any time you need to move from one world to another.

Double-Duty DOS

Please refer to the "Apple MLX" article in this issue before entering the following program.

```
3000: A9 0A 20 F5 BE 90 05 A9 07
3008: 0E 4C 09 BE AA A0 00 8A F1
3010: 99 5E 30 E8 C8 C0 07 90 51
3018: F6 8E DB 38 8E F0 38 E8 28
3020: 8A 99 5E 30 E8 C8 8A 99 14
3028: 5E 30 A0 00 B9 00 31 A2 AA
3030: 00 DD 55 30 F0 17 E8 E0 4C
3038: 09 90 F6 C8 D0 EE EE 2E FA
3040: 30 EE 52 30 AD 2E 30 C9 12
3048: 3B 90 E1 B0 1A BD 5E 30 66
3050: 99 00 31 D0 E6 31 32 33 44
3058: 34 35 36 37 39 3A 00 00 0D
3060: 00 00 00 00 00 00 00 00 A9 6A
3068: 31 85 3D A9 37 8D B7 36 9A
3070: AD 07 BE 8D 57 31 AD 08 FC
3078: BE 8D 58 31 A9 3A 85 3F 3A
3080: A0 FF 84 3E C8 84 3C 84 FA
3088: 42 8C 07 BE AD 5E 30 85 C6
3090: 43 8D 08 BE 20 2C FE A0 33
3098: 00 B9 A5 30 F0 06 20 ED EC
30A0: FD C8 D0 F5 60 C4 C3 C1 0B
30A8: D4 AC A0 C4 D3 C1 D6 C5 18
30B0: AC A0 C1 CE C4 A0 C4 CC B3
30B8: CF C1 C4 A0 CE CF D7 A0 1A
30C0: C9 CE D3 D4 C1 CC CC C5 22
30C8: C4 8D D7 D2 C9 D4 D4 C5 28
30D0: CE A0 C2 D9 A0 CA C1 D3 3E
30D8: CF CE A0 C3 CF CC C5 CD 30
30E0: C1 CE AC A0 CA D5 CC D9 96
30E8: AC A0 B1 B9 B8 B6 8D 8D E2
30F0: 00 00 00 00 00 00 00 00 51
30F8: 00 00 00 00 00 00 00 00 59
3100: D8 A0 00 A2 00 B9 00 02 0A
3108: C8 C9 A0 F0 F8 29 DF DD 6E
3110: 1B 31 D0 0B E8 E0 04 F0 DA
3118: 4A D0 EA C4 C3 C1 D4 A0 EC
3120: 00 A2 00 B9 00 02 C8 C9 2A
3128: A0 F0 F8 29 DF DD 5E 31 2D
3130: D0 08 E8 E0 05 D0 EC 4C B9
3138: D4 32 A0 00 A2 00 B9 00 2E
3140: 02 C8 C9 A0 F0 F8 29 DF B6
3148: DD 59 31 D0 08 E8 E0 05 CD
3150: D0 EC 4C 40 35 38 4C 00 07
3158: 00 C4 D3 C1 D6 C5 C4 CC A6
3160: CF C1 C4 8E 55 BE CA 8E 66
3168: 52 BE A9 31 8D 51 BE A9 C4
3170: 80 8D 50 BE A9 00 8D 53 28
3178: BE A9 10 8D 54 BE 18 60 AD
3180: AD 61 BE 8D 3C BE 0A 0A BD
3188: 0A 0A AE 62 BE 8E 3D BE D7
3190: CA F0 02 09 80 8D 84 39 E1
3198: A9 80 8D 83 39 A9 11 A0 0D
31A0: 00 8D 87 39 8C 88 39 A5 89
31A8: 74 8D 86 39 A9 00 8D 85 FA
31B0: 39 A0 39 A9 83 20 18 39 9F
31B8: 90 03 4C 7F 39 20 62 FC B1
31C0: A0 06 B1 73 85 02 A0 0A E1
31C8: B9 9C 32 20 ED FD 88 10 FF
31D0: F7 A9 A0 20 ED FD A5 02 64
31D8: 20 A7 32 20 62 FC 20 62 27
31E0: FC A0 0F 8C 88 39 A0 39 38
31E8: A9 83 20 18 39 90 03 4C E4
31F0: 7F 39 A5 74 85 01 A5 73 4C
31F8: 85 00 A0 0B 84 03 B1 00 76
3200: D0 03 4C 92 32 C9 FF F0 E9
3208: 76 C8 C8 B1 00 30 08 A9 88
```

```
3210: A0 20 ED FD 4C 1C 32 A9 4B
3218: AA 20 ED FD B1 00 29 7F D6
3220: F0 0E C9 01 F0 0E C9 02 1F
3228: F0 0E C9 04 F0 0E D0 4F B2
3230: A9 D4 D0 0A A9 C9 D0 06 75
3238: A9 C1 D0 02 A9 C2 20 ED A2
3240: FD A9 A0 20 ED FD A5 03 D9
3248: 18 69 21 A8 B1 00 20 A7 37
3250: 32 A9 A0 20 ED FD A4 03 02
3258: C8 C8 C8 A2 00 B1 00 20 7D
3260: ED FD C8 E8 E0 1D D0 F5 F5
3268: 20 62 FC AD 00 C0 10 0F 22
3270: A9 00 8D 10 C0 AD 00 C0 D9
3278: 10 FB A9 00 8D 10 C0 A5 EC
3280: 03 18 69 23 A8 B0 03 4C 26
3288: FC 31 CE 88 39 F0 03 4C F9
3290: E6 31 20 62 FC 20 62 FC 09
3298: 18 A9 00 60 C5 CD D5 CC 57
32A0: CF D6 A0 CB D3 C9 C4 8D 50
32A8: CF 32 A0 02 A9 00 48 AD 41
32B0: CF 32 D9 D1 32 90 0E F9 CB
32B8: D1 32 8D CF 32 68 18 69 0E
32C0: 01 48 4C AF 32 68 09 B0 32
32C8: 20 ED FD 88 10 DE 60 00 BD
32D0: 00 01 0A 64 A2 00 B9 00 85
32D8: 02 C8 C9 8D F0 1C C9 A0 AE
32E0: F0 F4 C9 8D F0 16 C9 AC 2D
32E8: F0 12 C9 E0 90 02 29 DF 50
32F0: 9D 01 02 E8 B9 00 02 C8 CD
32F8: D0 E8 38 60 E0 1F 90 02 B3
3300: A2 1E 8E 00 02 88 8C 52 AE
3308: BE A9 24 8D 50 BE A9 33 99
3310: 8D 51 BE A9 00 8D 53 BE 9F
3318: A9 10 8D 54 BE A9 84 8D 81
3320: 55 BE 18 60 AD 61 BE 8D E7
3328: 3C BE 0A 0A 0A 0A AE 62 76
3330: BE 8E 3D BE CA F0 02 09 54
3338: 80 8D 84 39 A5 74 8D 86 07
3340: 39 A9 00 8D 85 39 A9 11 FB
3348: 8D 87 39 A9 0F 8D 88 39 12
3350: A0 39 A9 83 20 18 39 90 27
3358: 03 4C 7F 39 A9 0E A8 A2 50
3360: 00 84 02 B1 73 F0 20 DD C0
3368: 01 02 D0 09 E8 EC 00 02 77
3370: F0 19 C8 D0 EE A5 02 18 E5
3378: 69 23 90 E2 AC 88 39 88 1F
3380: F0 05 8C 88 39 D0 C9 A9 05
3388: 06 38 60 E0 1E F0 0A C8 AB
3390: B1 73 C9 A0 D0 DF E8 D0 98
3398: F2 A4 02 88 88 88 B1 73 A7
33A0: 30 E5 8D 87 39 C8 B1 73 86
33A8: 8D 88 39 AE 86 39 E8 E8 DD
33B0: 8E 86 39 86 07 A2 00 86 D8
33B8: 06 C8 B1 73 29 7F C9 02 9E
33C0: F0 0B C9 04 D0 03 4C 90 97
33C8: 34 38 A9 0D 60 A9 83 A0 AE
33D0: 39 20 18 39 90 03 4C 7F 1B
33D8: 39 A5 67 A4 68 38 E9 02 76
33E0: B0 01 88 8D 85 39 8C 86 7A
33E8: 39 A0 00 8C 8B 34 C8 B1 4D
33F0: 06 8D 8F 34 C8 B1 06 8D 99
33F8: 8E 34 A0 0C B1 06 F0 29 39
3400: 8D 87 39 C8 B1 06 C8 8D 89
3408: 88 39 84 02 A9 83 A0 39 89
3410: 20 18 39 90 03 4C 7F 39 40
3418: A4 02 F0 26 EE 86 39 AD 85
3420: 86 39 C5 74 90 D6 A9 0E 5B
3428: 60 AD 85 39 85 69 85 AF FC
3430: AD 86 39 85 6A 85 B0 A0 FB
3438: 00 98 C6 67 91 67 E6 67 75
3440: 18 60 AD 85 39 8D 8C 34 28
3448: AD 86 39 8D 8D 34 A5 06 B7
3450: 8D 85 39 A5 07 8D 86 39 17
3458: AD 8F 34 8D 87 39 AD 8E E5
3460: 34 8D 88 39 A9 83 A0 39 C0
3468: 20 18 39 90 03 4C 7F 39 98
3470: AD 8C 34 8D 85 39 AD 8D 2C
3478: 34 8D 86 39 A0 0C AD 8B DE
3480: 34 D0 03 4C FC 33 84 02 1C
3488: 4C 0E 35 00 00 00 00 00 41
3490: A9 83 A0 39 20 18 39 90 BA
3498: 03 4C 7F 39 A0 01 B1 06 8B
34A0: 8D 8F 34 C8 B1 06 8D 8E 16
34A8: 34 A0 0C 8C 8B 34 B1 06 34
34B0: 8D 87 39 C8 B1 06 C8 84 31
34B8: 02 8D 88 39 A5 74 8D 86 CA
34C0: 39 A0 00 8C 85 39 A9 83 9E
34C8: A0 39 20 18 39 90 03 4C B3
34D0: 7F 39 AE 59 BE AD 58 BE CE
34D8: AC 57 BE 30 0C A0 01 B1 DE
34E0: 73 85 04 88 AA B1 73 85 F5
34E8: 03 E4 74 90 03 A9 0E 60 DE
34F0: 38 E9 04 B0 01 CA 85 08 C1
34F8: 86 09 A0 00 B1 73 91 08 81
3500: C8 D0 F9 A5 08 8D 85 39 57
3508: A4 09 C8 8C 86 39 A4 02 4D
3510: D0 03 4C 42 34 B1 06 F0 B6
3518: 25 8D 87 39 C8 B1 06 8D A3
3520: 88 39 C8 84 02 A9 83 A0 DC
3528: 39 20 18 39 90 03 4C 7F 76
3530: 39 EE 86 39 AD 86 39 C5 17
3538: 74 90 D3 A9 0E 60 18 60 98
3540: A2 00 B9 00 02 C8 C9 A0 9A
3548: F0 F8 C9 8D F0 0E C9 AC 7B
3550: F0 0A 9D 01 02 E8 B9 00 A0
3558: 02 C8 D0 EE 8E 00 02 88 FF
3560: 8C 52 BE A9 7E 8D 50 BE A1
3568: A9 35 8D 51 BE A9 00 8D E5
3570: 53 BE A9 10 8D 54 BE A9 4F
3578: 94 8D 55 BE 18 60 AE 00 C6
3580: 02 D0 04 A9 10 38 60 E0 3E
3588: 1F 90 05 A9 1E 8D 00 02 0B
3590: AD 61 BE 8D 3C BE 0A 0A D5
3598: 0A 0A AE 62 BE 8E 3D BE EF
35A0: CA F0 02 09 80 8D 84 39 F9
35A8: A9 80 8D 83 39 AD 57 BE DF
35B0: 10 0D AE 60 BE AD 5F BE 6C
35B8: 18 69 04 90 1C B0 19 A5 8E
35C0: B0 38 E5 68 AA A5 AF 38 58
35C8: E5 67 B0 01 CA 8E 60 BE 36
35D0: 8D 5F BE 18 69 02 90 01 A8
35D8: E8 E8 C9 00 F0 01 E8 8E 17
35E0: FB 37 E0 7B 90 04 A9 0E E0
35E8: 38 60 A4 74 C8 C8 C8 8C EA
35F0: 86 39 84 07 A9 00 8D C2 19
35F8: 36 85 06 8D 85 39 8D 88 2E
3600: 39 A9 11 8D 87 39 A0 39 0A
3608: A9 83 20 18 39 90 03 4C 0E
3610: 7F 39 A0 01 B1 06 C9 11 F8
3618: F0 03 4C 7F 39 C8 B1 06 95
3620: C9 0F D0 F6 20 FC 37 90 B2
3628: 01 60 E6 74 E6 74 AE FB B3
3630: 37 A0 00 98 91 73 C8 D0 A6
3638: FB A0 0C 84 08 A0 C1 B1 8C
3640: 06 F0 72 98 29 01 F0 07 F9
3648: A9 00 8D C6 36 F0 05 A9 D0
3650: 08 8D C6 36 98 38 E9 38 12
3658: 29 FE 4A 4A 8D C4 36 B1 A4
3660: 06 48 84 09 A0 07 2A B0 29
3668: 03 88 10 FA 98 18 6D C6 F0
3670: 36 8D C6 36 A4 08 AD C2 FA
3678: 36 D0 0F AD C4 36 8D C2 CD
3680: 36 AD C6 36 8D C3 36 68 FF
3688: D0 0F AD C4 36 91 73 C8 CA
3690: AD C6 36 91 73 C8 84 08 35
3698: 68 A0 08 2A B0 03 88 D0 78
36A0: FA 18 6A C0 08 F0 03 C8 BC
36A8: D0 F8 A4 09 91 06 CA F0 0C
36B0: 16 C9 00 D0 8A 88 C0 37 D6
36B8: D0 85 C6 74 C6 74 A9 09 73
36C0: 38 60 00 00 00 00 00 C6 28
36C8: 74 C6 74 A9 81 8D 83 39 CC
36D0: A9 11 8D 87 39 A9 00 8D 7E
36D8: 88 39 A5 07 8D 86 39 A9 9F
36E0: 83 A0 39 20 18 39 90 03 2A
36E8: 4C 7F 39 AD C2 36 8D 87 EE
36F0: 39 AD C3 36 8D 88 39 CE 11
36F8: 86 39 A9 83 A0 39 20 18 A6
3700: 39 B0 E5 CE 83 39 A5 07 34
3708: 8D 86 39 A9 11 8D 87 39 A7
3710: A9 0F 8D 88 39 A0 39 A9 B9
3718: 83 20 18 39 90 03 4C 7F 8F
3720: 39 A9 0B A8 B1 06 F0 11 1A
3728: 30 0F 98 18 69 23 90 F3 F3
3730: CE 88 39 D0 E0 A9 11 38 64
3738: 60 AD C2 36 91 06 C8 AD E1
3740: C3 36 91 06 C8 AE 57 BE 1F
3748: 10 04 A9 04 D0 02 A9 02 19
3750: 91 06 C8 A2 00 BD 01 02 47
3758: 91 06 C8 E8 EC 00 02 D0 F4
3760: F4 E0 1E F0 0B A9 A0 91 0E
3768: 06 E8 C8 D0 F4 AD FB 37 C7
3770: 91 06 C8 A9 00 91 06 A9 D8
3778: 81 8D 83 39 A9 83 A0 39 E4
3780: 20 18 39 90 03 4C 7F 39 B6
3788: C6 07 AE 57 BE 10 20 AE 8C
3790: 59 BE AD 58 BE 38 E9 04 45
3798: B0 01 CA 85 08 86 09 A0 5E
37A0: 00 AD 58 BE 91 08 C8 AD 5D
37A8: 59 BE 91 08 C8 D0 0A A9 6D
37B0: FF 85 08 A9 07 85 09 A0 1D
37B8: 00 AD 5F BE 91 08 C8 AD 56
37C0: 60 BE 91 08 A0 0C A5 08 4A
37C8: 8D 85 39 A5 09 8D 86 39 A5
37D0: B1 06 F0 20 C8 8D 87 39 7E
37D8: B1 06 8D 88 39 C8 8C FB DD
37E0: 37 A9 83 A0 39 20 18 39 83
37E8: 90 03 4C 7F 39 E6 09 AC 06
37F0: FB 37 D0 D2 A9 00 8D 00 DA
37F8: 08 18 60 00 A4 74 C8 C8 CE
3800: 8C 86 39 84 1A A9 00 8D CC
3808: 85 39 85 19 A9 11 8D 87 FF
3810: 39 A9 0F 8D 88 39 A9 83 42
3818: A0 39 20 18 39 90 03 4C 0B
3820: 7F 39 A0 01 B1 19 C9 11 59
3828: F0 03 4C 7F 39 A9 0E 48 28
3830: A8 A2 00 B1 19 D0 03 68 33
3838: 18 60 DD 01 02 D0 13 E8 FA
3840: C8 EC 00 02 D0 ED E0 1E 8E
3848: F0 15 B1 19 E8 C8 C9 A0 DC
3850: F0 F4 68 18 69 23 90 D7 D5
3858: CE 88 39 D0 B9 18 60 68 DD
3860: A8 88 B1 19 30 11 C9 04 6C
3868: F0 18 C9 02 D0 05 AD 57 FD
3870: BE 10 14 A9 0D 38 60 A9 15
3878: 0A 38 60 C8 C8 C8 98 48 77
3880: D0 D0 AD 57 BE 10 EC 88 51
3888: 88 98 48 18 69 21 A8 B1 C0
3890: 19 C9 7B 90 05 68 A9 0E A3
3898: 38 60 68 A8 B1 19 30 DB 03
38A0: 8D C4 36 A9 FF 91 19 C8 AB
38A8: B1 19 8D C5 36 A9 81 8D 2F
38B0: 83 39 A9 83 A0 39 20 18 E0
38B8: 39 90 03 4C 7F 39 CE 83 11
38C0: 39 AD C4 36 8D 87 39 AD DF
38C8: C5 36 8D 88 39 A9 83 A0 FB
38D0: 39 20 18 39 90 03 4C 7F 25
38D8: 39 20 F7 38 A0 0C B1 19 22
38E0: F0 13 8D C4 36 C8 B1 19 DD
38E8: 8D C5 36 C8 84 08 20 F7 61
38F0: 38 A4 08 D0 E9 18 60 AD D2
38F8: C4 36 0A 0A 18 69 38 A8 BA
3900: AD C5 36 C9 08 B0 01 C8 EB
3908: 29 07 AA A9 01 CA 30 03 57
3910: 0A D0 FA 11 06 91 06 60 0F
3918: 85 00 84 01 A0 01 B1 00 5A
3920: 8D 8A 39 C8 B1 00 8D 90 E8
3928: 39 C8 B1 00 8D 91 39 C8 8D
3930: B1 00 48 C8 B1 00 A8 68 58
3938: 20 B8 39 8D 8D 39 8C 8E E1
3940: 39 8E 8F 39 A5 74 8D 8C 1F
3948: 39 20 00 BF 80 89 39 B0 A8
3950: 2E A0 00 B1 00 C9 81 F0 38
3958: 3B A5 74 18 6D 8F 39 8D 8B
3960: 77 39 A9 00 8D 76 39 AD 78
3968: 90 39 8D 79 39 AD 91 39 97
3970: 8D 7A 39 A0 00 B9 00 96 F6
3978: 99 00 20 C8 D0 F7 60 38 A7
3980: A9 08 60 80 60 00 00 00 E0
3988: 00 03 00 00 00 00 00 00 BB
3990: 00 00 00 00 A5 74 18 6D 9F
3998: 8F 39 8D 7A 39 A9 00 8D 78
39A0: 79 39 AD 90 39 8D 76 39 03
39A8: AD 91 39 8D 77 39 20 73 AA
39B0: 39 20 00 BF 81 89 39 60 C8
39B8: 8C 93 39 AA A0 00 98 CA 29
39C0: 30 08 18 69 08 90 F8 C8 24
39C8: D0 F5 8D 92 39 A2 00 AD FD
39D0: 93 39 D0 04 AD 92 39 60 40
39D8: C9 01 D0 0A AD 92 39 18 6D
39E0: 69 07 90 01 C8 60 C9 0E 55
39E8: D0 05 AD 92 39 E8 60 C9 DB
39F0: 0F D0 0B AD 92 39 18 69 6E
39F8: 07 90 01 C8 E8 60 AD 93 77
3A00: 39 4A B0 05 A9 0E E8 D0 32
3A08: 02 A9 0F 38 ED 93 39 4A C7
3A10: 18 6D 92 39 90 01 C8 60 4C
```

©

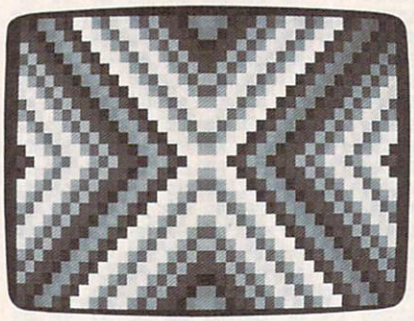
Dazzlers

Paul W. Carlson

Grab your sunglasses—here come the dazzlers! Several demos are included along with information on how to create your own brilliant, rapidly changing patterns. For the IBM PC/PCjr or compatible with a color/graphics adapter or equivalent hardware. Requires BASICA for the PC, GW-BASIC for compatibles, and Cartridge BASIC for the PCjr.

Everyone has their own favorite graphics demos. "Dazzlers" is a collection of programs, each featuring flowing colors and simulated movement. After seeing these kaleidoscopic displays, you may decide to put a couple of them onto your own favorite demo disk. After you've tried the patterns I've included, you may even want to design some of your own.

Program 1 is the main program for Dazzlers. It cannot be used independently, but must be combined with one of five subroutines (Programs 2–6). Type in and save a copy of Program 1. Then add the lines from one of the subroutines to the program and save it with a unique filename. Run the program. The screen clears; then it's filled with solid white. Next, blocks of color appear on the screen, forming an intricate pattern. And finally, the colors are repeatedly cycled. Press any key after the colors begin changing to clear the screen and end the program.



Waves of color ripple across the screen in "Dazzlers," a set of animated graphics demos.

The Subroutines

Any of the subroutines (Programs 2–5) can be used by merging them with Program 1. There are a number of elements common to each of these subroutines. And for a reason: Once an interesting pattern is created, making only minor changes in the subroutine will create an entirely different and unique pattern. This is an important point to remember when designing your own dazzler subroutines.

Here are a few more points to keep in mind: The subroutine should POKE values from 0 to 15 into every odd-numbered memory location in video page 0 from 1 to 1999. The same numbers must also be POKEd into the odd locations of video page 1 (2049–4047). To make the colors flow properly, adjacent characters (adjacent horizontally, vertically, or diagonally) should have consecutive color numbers. One method that can be used to generate consecutive color numbers is to enclose a statement such as C=(C+1) AND 15 in a loop, where the variable C is the color number. This statement will assign to C the repeating sequence of consecutive numbers 0–15.

An almost endless variety of patterns is possible. You can lay out a pattern on graph paper and then write a subroutine that will reproduce the pattern on the screen, or you can simply alter a previous subroutine until you get something totally new. Both methods were used to write the subroutines in this article.

How It Works

At the heart of the program is a short, very fast machine language routine that increments every character's foreground color. Those characters with a foreground color number of 15 receive a foreground color number of zero. For machine language programmers, the figure shows the source code for this program.

Line 10 in the BASIC program reserves an area of memory for the machine language routine. Lines 20–80 POKE the machine language routine into the reserved memory. Line 90 sets up a 40-column color screen with video page 0 the active page and video page 1 the display page. Each page is each 2048 bytes long, although only the first 2000 bytes of each page are used. The

Source Code for Color-Changing Routine

```
; This subroutine increments the foreground color of every
; character on video page 0 or page 1.  The subroutine is called
; with the video page offset as the only argument.
;
CSEG    SEGMENT
DAZZLE  PROC    FAR
        ASSUME  CS:CSEG
        PUSH    ES                  ;Save the extra segment
        MOV     BP,SP               ;Make BP point to stack
        MOV     AX,0B800H           ;Set extra segment to start of
        MOV     ES,AX               ;     video RAM
        MOV     SI,6[BP]            ;SI is address of page offset
        MOV     DI,WORD PTR [SI]    ;DI is page offset
        INC     DI                  ;Bump DI to attribute byte
        MOV     CX,1000             ;1000 bytes to move
MLOOP:  MOV     AL,ES:[DI]          ;Get attribute byte
        MOV     DL,AL               ;Copy byte into DL
        INC     DL                  ;Increment attribute byte
        AND     DL,15               ;DL is new foreground color
        AND     AL,240              ;Mask out old foreground color
        OR      AL,DL               ;OR in new foreground color
        MOV     ES:[DI],AL          ;Move new attribute byte to video RAM
        INC     DI                  ;Bump DI to next attribute byte
        INC     DI
        LOOP    MLOOP               ;Get next attribute byte
        POP     ES                  ;Restore the extra segment
        RET     2                   ;Clean up the stack
DAZZLE  ENDP
CSEG    ENDS
        END
```

even-numbered bytes contain the ASCII code that determines which character will appear on the screen. Line 100 of the program POKEs the number 219, the ASCII code for a solid block, into the even-numbered bytes of both video pages 0 and 1. This is what causes the screen to fill with solid white. Then line 100 executes a GOSUB 1000 which POKEs the foreground color numbers into the odd-numbered bytes of both pages.

The odd-numbered bytes contain the color information for each character. Before the subroutine returns, it assigns a value to the variable TD, which controls the time delay. In line 110, the variable A is the address of the machine language routine, and the variables F% and S% are the number of bytes from the beginning of video RAM for video page 0 and page 1. Lines 120 and 130 form a loop in which the foreground colors of the video page not being displayed are incremented. That page is then displayed, a time delay is executed, and the keyboard is checked for a keypress. Line 140 is executed when a key is pressed, clearing the screen and returning the display to 80-column video mode.

For instructions on entering these programs, please refer to "COMPUTE!'s Guide to Typing In Programs" elsewhere in this issue.

Program 1: Dazzlers

```
CJ 1 'Copyright 1987 COMPUTE! Pu
     blications, Inc.  All Right
     s Reserved.
KC 3 KEY OFF:CLS:LOCATE 10,34:PR
     INT"Copyright 1987":PRINT T
     AB(28)"COMPUTE! Publication
     s, Inc.":PRINT TAB(31)"All
     Rights Reserved.":FOR I = 1
      TO 1500:NEXT I
FA 10 DEF SEG:CLEAR,&H3FF0:N=&H4
     000
HM 20 READ A$:IF A$="/" THEN 90
DC 30 POKE N,VAL("&H"+A$):N=N+1:
     GOTO 20
MF 40 DATA 06,8B,EC,B8,00,B8,8E,
     C0,8B,76
OK 50 DATA 06,8B,3C,47,B9,E8,03,
     26,8A,05
BD 60 DATA 8A,D0,FE,C2,80,E2,0F,
     24,F0,0A
IK 70 DATA C2,26,88,05,47,47,E2,
     EB,07,CA
HN 80 DATA 02,00,/
FP 90 WIDTH 40:SCREEN 0,1,0,1:CL
     S:DEF SEG=&HB800
PN 100 FOR N=0 TO 2046 STEP 2:PO
     KE N,219:POKE N+2048,219:
     NEXT:GOSUB 1000
OB 110 DEF SEG:A=&H4000:F%=0:S%=
     2048
HF 120 CALL A(F%):SCREEN ,,,0:FO
     R N=1 TO TD:NEXT:CALL A(S
     %):SCREEN ,,,1
ON 130 FOR N=1 TO TD:NEXT:IF INK
     EY$="" THEN 120
BM 140 WIDTH 80:CLS:END
```

Program 2: Subroutine 1

```
GA 998 ' Subroutine 1
KJ 999 '
KF 1000 FOR P=0 TO 1:FOR R=0 TO
     12:C=R:K=2048*P
ND 1010 FOR L=0 TO 38 STEP 2:C=(
     C+1) AND 15
AE 1020 POKE 1+80*R+L+K,C:POKE 7
     9+80*R-L+K,C
IH 1030 POKE 1921-80*R+L+K,C:POK
     E 1999-80*R-L+K,C:NEXT:N
     EXT:NEXT
MO 1040 TD=50:RETURN
```

Program 3: Subroutine 2

```
HC 998 ' Subroutine 2
KJ 999 '
KF 1000 FOR P=0 TO 1:FOR R=0 TO
     12:C=R:K=2048*P
ND 1010 FOR L=0 TO 38 STEP 2:C=(
     C+1) AND 15
NK 1020 POKE 41+80*R+L+K,C:POKE
     39+80*R-L+K,C
DD 1030 POKE 1961-80*R+L+K,C:POK
     E 1959-80*R-L+K,C:NEXT:N
     EXT:NEXT
MO 1040 TD=50:RETURN
```

Program 4: Subroutine 3

```
IE 998 ' Subroutine 3
KJ 999 '
BH 1000 FOR P=0 TO 1:FOR R=0 TO
     6:C=R:K=2048*P
LB 1010 FOR L=0 TO 18 STEP 2:C=(
     C+1) AND 15
MI 1020 D=1+80*R+L+K:POKE D,C:PO
     KE D+40,C:POKE D+960,C:P
     OKE D+1000,C
CL 1030 D=39+80*R-L+K:POKE D,C:P
     OKE D+40,C:POKE D+960,C:
     POKE D+1000,C
EG 1040 D=961-80*R+L+K:POKE D,C:
     POKE D+40,C:POKE D+960,C
     :POKE D+1000,C
OH 1050 D=999-80*R-L+K:POKE D,C:
     POKE D+40,C:POKE D+960,C
     :POKE D+1000,C:NEXT:NEXT
     :NEXT
ME 1060 TD=50:RETURN
```

Program 5: Subroutine 4

```
JG 998 ' Subroutine 4
KJ 999 '
MC 1000 FOR P=0 TO 1:FOR C=0 TO
     11:N=2048*P+82*C-1
EE 1010 FOR K=C TO 39-C:N=N+2:PO
     KE N,C:NEXT
DF 1020 FOR K=C TO 23-C:N=N+80:P
     OKE N,C:NEXT
HK 1030 FOR K=C TO 38-C:N=N-2:PO
     KE N,C:NEXT
GF 1040 FOR K=C TO 22-C:N=N-80:P
     OKE N,C:NEXT:NEXT
HL 1050 FOR N=985 TO 1015 STEP 2
     :POKE N+2048*P,12:NEXT:N
     EXT
NK 1060 TD=100:RETURN
```

Program 6: Subroutine 5

```
LI 998 ' Subroutine 5
KJ 999 '
PB 1000 FOR P=0 TO 1:FOR C=0 TO
     11:N=2048*P+82*C-1:B=1
PE 1010 FOR K=C TO 39-C:N=N+2:B=
     (B+1) AND 15:POKE N,B:NE
     XT:B=2
LE 1020 FOR K=C TO 23-C:N=N+80:B
     =(B+1) AND 15:POKE N,B:N
     EXT:B=1
CK 1030 FOR K=C TO 38-C:N=N-2:B=
     (B+1) AND 15:POKE N,B:NE
     XT:B=2
ML 1040 FOR K=C TO 22-C:N=N-80:B
     =(B+1) AND 15:POKE N,B:N
     EXT:NEXT:B=17
NK 1050 FOR N=985 TO 1015 STEP 2
     :B=(B+1) AND 15:POKE N+2
     048*P,B:NEXT:NEXT
NK 1060 TD=100:RETURN
```

Word War

Peter Crosby

This simple but challenging word game pits you against a friend and is bound to provide hours of entertainment. The game requires 512K of RAM, Amiga Basic, and Kickstart version 1.2.

"Word War" is a quick-paced word game that takes advantage of the Amiga's advanced features—its mouse, many colors, and excellent sound. The object is simple—players guess letters and words to take over the *Power Bar* gauge located at the top of the screen. Whenever a player correctly guesses a letter or word, his or her color eats up more of the Power Bar. When one player's color overwhelms the other, the game is over. In essence, Word War combines the playing concepts of both hangman and tug-of-war.

Typing It In

Word War is written in Amiga Basic. Type it in and save a copy to disk. When you're ready to play, load and run the program.

Enter the names of the two players. The game screen will appear. Take a moment to familiarize yourself with the screen. Near the top is the Power Bar. At the beginning of the game, the left half of the bar is red and the right is blue. As the players score points, the border between the colors moves to the left or to the right. Eventually, one player wins the game by taking over the entire bar.

Below the Power Bar are two boxes which hold the names of the players. Player 1 always plays red, while player 2 plays blue. During player 1's turn, the red box lights up. During player 2's turn, the blue name box is lit. Both players should watch these boxes to be sure that they do not play their opponent's turn.

"Language" is the secret word in this round of "Word War," a challenging game for the Amiga.

The *Action Board* can be found directly below the names of the players. At the bottom of this board is the alphabet. A mystery word appears in the small box which appears to hang from the Board. At first the word is made up of black squares in the place of the mystery word letters. Players use the mouse to point at and click on the letter of the alphabet that they wish to guess. If the chosen letter can be found in the word, each occurrence of the letter lights up in the word and the player gets another turn. In addition, the Power Bar moves slightly in favor of the player. If the letter cannot be found in the word, the other player gets a turn. In the center of the board is a box labeled *I Know It*. A player who wants to take a guess at the mystery word can use the mouse to point at and click on this box. After selecting this box, the player types in his or her guess. If it is correct, the Power Bar changes to favor the player. If the guess is incorrect, the player's opponent gains points instead.

When one player needs to make a desperate recovery, or wants to jump far ahead in score, he or she clicks on one of his or her three *Revival Squares*. (This can be done only during the player's turn.) These squares are located on the extreme left (for the red player) and the extreme right (for the blue player) of the Action Board. During the revival, a series of scrambled words appears at the top of the screen. The player must quickly unscramble these words and type them in. When time runs out, the game returns to the main screen and the revival points are placed into the power bar. During the revival, the Delete key is inactive. To correct a typing error, use the space bar as a Delete key.

The accompanying listing has a vocabulary of 50 words. If you wish to use more, you must make a few

THE CBS COMPACT DISC CLUB HAS THE MUSIC YOU WANT

TAKE ANY 3 FOR $1 WITH MEMBERSHIP

354902 354449 353946 345751

349985. Johnny Mathis/Henry Mancini—The Hollywood Musicals (Columbia)

348979. Tina Turner—Break Every Rule (Capitol)

352633. Dolly Parton/Linda Ronstadt/Emmylou Harris—Trio (Warner Bros.)

336396-396390. Billy Joel's Greatest Hits, Vol. 1 & 2. (Columbia)

346643. Andreas Vollenweider—Down To The Moon. (CBS)

346478. Madonna—True Blue. (Sire)

343319. Janet Jackson—Control (A&M)

349571. Boston—Third Stage (MCA)

290916. The Best Of Earth, Wind & Fire, Vol. I (Columbia/Arc)

257279. Bruce Springsteen—Born to Run (Columbia)

138586. Bob Dylan's Greatest Hits (Columbia)

319541. Elton John—Greatest Hits. (MCA)

342105. Bangles—Different Light. (Columbia)

287003. Eagles Greatest Hits 1971-1975 (Asylum)

347492. Glenn Miller Orchestra—In The Digital Mood. (*Digital*—GRP)

293597. Led Zeppelin—Houses Of The Holy. (Atlantic)

350736. Rolling Stones—Rewind. (Rolling Stones Records)

346957. Steve Winwood—Back In The High Life. (Island)

344622. Anita Baker—Rapture. (Elektra)

319996-399998. Motown's 25 #1 Hits From 25 Years. (Motown)

291278. The Doobie Brothers—Best of the Doobies. (Warner Bros.)

345777. Peter Gabriel—So. (Geffen)

246868. Jim Croce—Photographs And Memories—His Greatest Hits. (Saja)

334391. Whitney Houston. (Arista)

314443. Neil Diamond's 12 Greatest Hits, Vol. 2. (Columbia)

308049. Creedence Clearwater Revival Featuring John Fogerty/Chronicle. 20 greatest hits. (Fantasy)

343582. Van Halen—5150. (Warner Bros.)

326629. Bruce Springsteen—Born In The U.S.A. (Columbia)

342097. Barbra Streisand—The Broadway Album. (Columbia)

219477. Simon & Garfunkel's Greatest Hits. (Columbia)

348649. Pachelbel Canon & Other Digital Delights—Toronto Chamber Orch. (*Digital*—Fanfare)

353771. Bolling/Rampal: Suite #2 for Flute & Jazz Piano Trio (*Digital*—CBS)

348318. The Police—Every Breath You Take—The Singles (A&M)

346312. Billy Joel—The Bridge. (*Digital*—Columbia)

336222. Dire Straits—Brothers In Arms. (Warner Bros.)

341073. Steely Dan—A Decade of Steely Dan. (MCA)

349373. Beethoven: Symphony No. 9 (Choral) Bernstein, NY. Phil. (*Digitally Remastered*—CBS Masterworks)

314997-394999. Stevie Wonder's Original Musiquarium I. (Tamla)

348110. Buddy Holly—From The Original Master Tapes. (*Digitally Remastered*—MCA)

352245. David Sanborn—A Change Of Heart. (Warner Bros.)

351692. Beastie Boys—Licensed To Ill. (Def Jam/Columbia)

346544. Kenny G—Duotones. (Arista)

344721. Lionel Richie—Dancing On the Ceiling. (Motown)

355156. Vladimir Horowitz Plays Favorite Chopin (*Digitally Remastered*—CBS Masterworks)

354985. Billie Holiday—From The Original Decca Masters. (*Digitally Remastered*—MCA)

323899. The Best Of The Alan Parson's Project. (Arista)

286740. Linda Ronstadt's Greatest Hits. (Asylum)

355115-395111. Prince—Sign 'O' The Times. (Paisley Park)

357186. Hooters—One Way Home. (Columbia)

356873. The Outfield—Bangin'. (Columbia)

355636. Ozzy Osbourne—Randy Rhoads Tribute. (CBS Assoc.)

355529. Philip Glass—Dance Pieces. (*Digital*—CBS)

355362. Whitesnake. (Geffen)

354829. Lisa—Lisa And Cult Jam With Full Force—Spanish Fly. (Columbia)

354167. Dan Fogelberg—Exiles. (Full Moon/Epic)

354100. Crowded House. (Capitol)

354035. Dwight Yoakam—Hillbilly Deluxe. (Reprise)

260638. Chicago's Greatest Hits. (Columbia)

356279. Gloria Estefan And Miami Sound Machine—Let It Loose. (Epic)

352948

355990

353821

347955

356154

356675

COMPACT disc DIGITAL AUDIO

The age of CD sound is here—and you have a practical new way to find the CDs you want. As your introduction to the CBS Compact Disc Club, you can choose any 3 CDs listed in this ad for just $1.00. Fill in and mail the application—we'll send your CDs and bill you for $1. You simply agree to buy 2 more CDs (at regular Club prices) in the next year—and you may then cancel your membership anytime after doing so.

How the Club works. About every four weeks (13 times a year) you'll receive the Club's music magazine, which describes the Selection of the Month for your musical interest...plus many exciting alternates. In addition, up to six times a year, you may receive offers of Special Selections, usually at a discount off regular Club prices, for a total of up to 19 buying opportunities.

If you wish to receive the Selection of the Month, you need do nothing—it will be shipped automatically. If you prefer an alternate selection, or none at all, fill in the response card always provided and mail it by the date specified. You will always have at least 10 days in which to make your decision. If you ever receive any Selection without having 10 days to decide, you may return it at our expense.

The CDs you order during your membership will be billed at regular Club prices, which currently are $14.98 to $15.98—plus shipping and handling. (Multiple-unit sets may be somewhat higher.) After completing your enrollment agreement you may cancel membership at any time; if you decide to continue as a member, you'll be eligible for our money-saving bonus plan. It lets you buy one CD at half price for each CD you buy at regular Club prices.

10-Day Free Trial: We'll send details of the Club's operation with your introductory shipment. If you are not satisfied for any reason whatsoever, just return everything within 10 days and you will have no further obligation. So why not choose 3 CDs for $1 right now.

ADVANCE BONUS OFFER: As a special offer to new members, take one additional Compact Disc right now and pay *only* $6.95. It's a chance to get a fourth selection at a super low price!

Selections with two numbers contain 2 CDs and count as 2—so write in both numbers.

CBS COMPACT DISC CLUB: Terre Haute, IN 47811

CBS COMPACT DISC CLUB, 1400 N. Fruitridge
P.O. Box 1129, Terre Haute, Indiana 47811-1129

Please accept my membership application under the terms outlined in this advertisement. Send me the 3 Compact Discs listed here and bill me only $1.00 for all three. I agree to buy two more selections at regular Club prices in the coming year—and may cancel my membership at any time after doing so.

SEND ME THESE 3 CDs:

My main musical interest is (check one): *(But I may always choose from any category)*

☐ ROCK/POP ☐ CLASSICAL

Mr.
Mrs.
Miss _____
Print First Name Initial Last Name

Address _____ Apt. _____

City _____

State _____ Zip _____

Do you have a VCR? (Check one.) ☐ Yes ☐ No

Do you have a credit card? (Check one.) ☐ Yes ☐ No

489/F87

ADVANCE BONUS OFFER: Also send me a fourth CD right now at the super low price of just $6.95, which will be billed to me.

This offer is not available in APO, FPO, Alaska, Hawaii, Puerto Rico; Please write for details of alternative offer. Canadian residents will be serviced from Toronto. 3WV/C2

Note: We reserve the right to reject any application or cancel any membership. 3WW/C3

changes to the program. The arrays Word$( ) and RevWord( ) must be dimensioned to the number of words in your word data. (The words are in DATA statements at the end of the program.) The 49 in the FOR X=0 TO 49 statement of the InitWords subroutine needs to be changed to the number of words you are using minus one. The number 50 in the RND statements of lines 100 and 550 should also be changed to reflect the revised number of words. At the start of each game, BASIC reads words from the word data into the array word, checking for duplication as it proceeds. The greater the number of words used, the longer the delay at the start of a game.

Word War

For instructions on entering this program, please refer to "COMPUTE!'s Guide to Typing In Programs" elsewhere in this issue.

```
' WordWar
' Copyright 1987 COMPUTE! Public
ations, Inc.
' All Rights Reserved.
Header:
CLS:CLEAR
DIM waveform%(255),Word$(50),Rev
Word(50),PaLCoL(12,3),UsedLetter
(26),Jum$(12),jm(12),Used(12),Rg
uess$(12)
GOSUB InitWords
GOSUB MenuSet
FOR n=0 TO 127
waveform%(n)=127:NEXT
WAVE 0,waveform%
SCREEN 2,320,200,5,1:WINDOW 2,"
Word War",(0,0)-(311,186),16,2
PRINT TAB(13)"Copyright 1987":PR
INT TAB(7)"COMPUTE! Publications
, Inc.":
PRINT TAB(11)"All Rights Reserve
d.":FOR t=1 TO 10000:NEXT
GOSUB ReadCoLor
GOSUB RestCoLor
GOSUB PLayers
LINE(30,20)-(282,32),31,bf:LINE(
23,10)-(149,22),21,bf:LINE(149,1
0)-(275,22),23,bf
LINE(22,9)-(276,23),30,b:LINE(21
,8)-(277,24),20,b
LINE(40,39)-(110,52),31,bf:LINE(
207,39)-(278,52),31,bf
LINE(30,29)-(100,42),24,bf:LINE(
29,28)-(101,43),20,b
LINE(198,29)-(268,42),22,bf:LINE
(197,28)-(269,43),20,b
LOCATE 5,5:COLOR 30,21:PRINT PLa
y$(1):COLOR 30,23:LOCATE 5,26:PR
INT PLay$(2):COLOR 20,0
LINE(30,60)-(282,145),31,bf:LINE
(23,50)-(275,135),25,bf:LINE(23,
50)-(275,135),20,b
LINE(35,136)-(40,150),28,bf:LINE
(38,136)-(40,150),2,bf:LINE(258,
136)-(263,150),28,bf:LINE(261,13
6)-(263,150),2,bf
LINE(45,136)-(50,150),31,bf:LINE
(268,136)-(273,150),31,bf
LINE(30,158)-(282,168),31,bf:LIN
E(23,150)-(275,161),29,bf:LINE(2
2,149)-(276,162),20,b
COLOR 20,0:LINE(99,81)-(199,103)
,27,b
LINE(98,80)-(198,102),2,bf:COLOR
30,2:LOCATE 12,15:PRINT"I Know I
t"
FOR y= 66 TO 106 STEP 15
LINE(35,y+1)-(45,y+11),27,bf:LIN
E(33,y)-(43,y+10),24,bf
LINE(257,y+1)-(267,y+11),27,bf:L
INE(255,y)-(265,y+10),22,bf
NEXT y
COLOR 20,4
er$="          "
Marker=149
PfLag=2:GOSUB ChangePLay
WordNum=-1:BeepFLag=0

SetUp:
COLOR 30,0:WordNum=WordNum+1
LINE(45,110)-(259,122),27,b
LINE(44,109)-(258,121),0,bf:LINE
(44,109)-(258,121),2,b
FOR x=65 TO 90:LOCATE 15,x-58:PR
INT CHR$(x);:NEXT
COLOR 20,4
L=LEN(Word$(WordNum))
LetterLoc=INT((39-(L*2))/2)
FOR x=1 TO L
LOCATE 20,LetterLoc+x*2:COLOR ,2
+x:PRINT MID$(Word$(WordNum),x,1
)
NEXT
high=3:NumRight=0:FOR x=1 TO 26:
UsedLetter(x)=0:NEXT

MouseCLick:
IF MOUSE (0)=0 THEN MouseCLick
XLetter=MOUSE(5)
YLetter=MOUSE(6)
IF YLetter>110 AND YLetter<121 A
ND XLetter>49 AND XLetter<256 TH
EN GOSUB SeLectLetter
IF YLetter>80 AND YLetter<102 AN
D XLetter>98 AND XLetter<198 THE
N GOSUB GuessIt
IF YLetter>65 AND YLetter<120 AN
D XLetter>33 AND XLetter<43 THEN
GOSUB Revive1
IF YLetter>65 AND YLetter<120 AN
D XLetter>255 AND XLetter<265 TH
EN GOSUB Revive2
GOTO MouseCLick

GuessWord:
COLOR 2,0
Guess$=CHR$(SeLection)
FOR g=1 TO L
IF MID$(Word$(WordNum),g,1)=UCAS
E$(Guess$) THEN high=g+2:GOSUB H
ighLight
NEXT g
IF BeepFLag=0 THEN SOUND 80,5,15
0,0:GOSUB ChangePLay
BeepFLag=0
RETURN

HighLight:
NumRight=NumRight+1
IF NumRight=L THEN Winner
FOR x=500 TO 1000 STEP 50
SOUND x,1,150,1:NEXT
FOR x=.1 TO 1 STEP .1
PALETTE high,x,x,0:FOR y=0 TO 25
0:NEXT y,x
Score=3:GOSUB Score
BeepFLag=1
RETURN

SeLectLetter:
SeLection=(INT(XLetter/8)-5)+64
IF UsedLetter(SeLection-64)=1 TH
EN GOTO CLearMouse
UsedLetter(SeLection-64)=1
COLOR 0,0:LOCATE 15,SeLection-58
:PRINT CHR$(SeLection)
GOSUB GuessWord
GOTO CLearMouse

CLearMouse:
IF MOUSE(0)<>0 THEN CLearMouse
RETURN

GuessIt:
COLOR 2,25
a$=INKEY$:IF a$<>"" THEN GuessIt

IF a$=CHR$(13) THEN GuessIt
LOCATE 8,7:INPUT"Your Guess? ",g
w$
IF UCASE$(gw$)=Word$(WordNum) TH
EN Score=(L-NumRight)*5:GOTO Win
ner
FOR x=200 TO 50 STEP -10:SOUND x
,1,150,2:NEXT
LINE(24,51)-(274,65),25,bf
Score=(L-NumRight)*5:GuessWrong=
1:GOSUB ChangePLay
GOSUB CLearMouse
RETURN

Winner:
s1=1600:s2=1000:s3=800:s4=600:st
=-100
FOR Loop= 1 TO 4
FOR x=s1 TO s2 STEP st
SOUND x,1,150,1:NEXT
FOR x=s3 TO s4 STEP st
SOUND x,1,150,1:NEXT
FOR high=3 TO L+3
FOR x=.1 TO 1 STEP .1
PALETTE high,x,x,0:NEXT x
PALETTE high,.4,.4,0
NEXT high
s1=400:s2=600:s3=800:s4=1400:st=
100
NEXT Loop
GOSUB Score
COLOR 0,0:LINE(24,51)-(274,65),2
5,bf
LINE(23,150)-(275,161),29,bf
GOSUB RestCoLor
GOSUB CLearMouse
BeepFLag=0
GOTO SetUp

RestCoLor:
FOR x=3 TO 16:PALETTE x,0,0,0:NE
XT x:PALETTE 20,0,0,0
RETURN

InitWords:
RESTORE WordData
RANDOMIZE TIMER
FOR x=0 TO 49
100 y=(INT(RND*50)):IF Word$(y)<
>"" THEN 100
READ Word$(y)
NEXT x
RETURN

ReadCoLor:
PALETTE 0,0,.3,.3
PALETTE 1,0,.3,.3
RESTORE CoLorData
FOR x=0 TO 11
FOR y=0 TO 2
READ PaLCoL(x,y)
NEXT y
NEXT x
FOR x=0 TO 11
PALETTE x+20,PaLCoL(x,0),PaLCoL(
x,1),PaLCoL(x,2)
NEXT x
RETURN

PLayers:
COLOR 2,21
LINE(34,61)-(284,76),31,bf:LINE(
24,51)-(274,66),21,bf:LINE(24,51
```

```
)-(274,66),20,b
LOCATE 8,5:PRINT"Player #1: ";:C
OLOR 30,21:LINE INPUT"",PLay$(1)
COLOR 2,23
LINE(34,92)-(284,108),31,bf:LINE
(24,82)-(274,98),23,bf:LINE(24,8
2)-(274,98),20,b
LOCATE 12,5:PRINT"Player #2: ";:
COLOR 30,23:LINE INPUT"",PLay$(2
)
COLOR 20,0
FOR x=1 TO 2
IF LEN(PLay$(x))>8 THEN PLay$(x)
=MID$(PLay$(x),1,8)
Lp=LEN(PLay$(x))
PLay$(x)=SPACE$((8-Lp)/2) + PLay
$(x)
NEXT
CLS
RETURN

ChangePLay:
IF PfLag=1 THEN Switch1
IF PfLag=2 THEN Switch2

Switch2:
PALETTE 22,0,0,.5:PALETTE 24,1,0
,0
COLOR 30,24:LOCATE 5,5:PRINT PLa
y$(1):COLOR 20,22:LOCATE 5,26:PR
INT PLay$(2)
COLOR 20,0:PfLag=1
IF GuessWrong=1 THEN GOSUB Score

RETURN

Switch1:
PALETTE 22,0,0,1:PALETTE 24,.3,0
,0
COLOR 20,24:LOCATE 5,5:PRINT PLa
y$(1):COLOR 30,22:LOCATE 5,26:PR
INT PLay$(2)
COLOR 20,0:PfLag=2
IF GuessWrong=1 THEN GOSUB Score

RETURN

Score:
IF PfLag=1 THEN GOSUB PLay1 ELSE
GOSUB PLay2
Score=0:GuessWrong=0
RETURN

PLay2:
IF Marker-Score=<23 THEN
LINE(23,10)-(275,22),23,bf
GOTO EndGame
END IF
Marker=Marker-Score:LINE(Marker,
10)-(275,22),23,bf
RETURN

PLay1:
IF Marker+Score=> 275 THEN
LINE(23,10)-(275,22),21,bf
GOTO EndGame
END IF
Marker=Marker+Score:LINE(23,10)-
(Marker,22),21,bf
RETURN

EndGame:
FOR x=1 TO 15
FOR y=0 TO 1 STEP .1
PALETTE 0,y,y,y:FOR z=1 TO 50:NE
XT
NEXT
NEXT
PALETTE 0,0,.3,.3
LINE(23,10)-(275,22),20,bf:COLOR
30,29:LOCATE 20,9:PRINT "  " PLa
y$(PfLag) " has WON!!!    "
200 LINE(24,51)-(274,65),25,bf
COLOR 20,25:LOCATE 8,7:INPUT"Ano
ther Game? ",An$
IF UCASE$(An$)="N" THEN 300
IF UCASE$(An$)="Y" THEN GOSUB CL
earMouse:GOTO Header
GOTO 200
300 MENU RESET
WINDOW CLOSE 2
SCREEN CLOSE 2
SCREEN CLOSE 1
WINDOW CLOSE 1
END

NewGame:
GOSUB CLearMouse
GOTO Header

Revivel:
IF (PfLag=2) OR (Rev(1)>2) THEN
RETURN
IF YLetter>65+(Rev(1)*15) AND YL
etter<75+(Rev(1)*15) THEN 400
RETURN
400 LINE(33,65+(Rev(1)*15))-(45,
77+(Rev(1)*15)),25,bf
COLOR 20,0:Rev(1)=Rev(1)+1
GOSUB CLearMouse
GOSUB ReviveMe
RETURN

Revive2:
IF (PfLag=1) OR (Rev(2)>2) THEN
RETURN
IF YLetter>65+(Rev(2)*15) AND YL
etter<75+(Rev(2)*15) THEN 500
RETURN
500 LINE(255,65+(Rev(2)*15))-(26
7,77+(Rev(2)*15)),25,bf
COLOR 20,0:Rev(2)=Rev(2)+1
GOSUB CLearMouse
GOSUB ReviveMe
RETURN

ReviveMe:
RANDOMIZE TIMER
FOR x=0 TO 49:RevWord(x)=0:NEXT
GOSUB ResetVaL:Score=0:rw=1:pit=
0
WINDOW 3,"",(0,0)-(311,186),16,2
:IF PfLag=1 THEN CoL=24 ELSE CoL
=22
LINE(0,0)-(311,186),20,bf:LINE(1
5,0)-(296,182),0,bf
LINE(46,21)-(286,32),31,bf:LINE(
39,14)-(279,25),29,bf:LINE(38,13
)-(280,26),20,b
FOR x=1 TO 5
550 rn=INT(RND*50)
FOR y=1 TO 5:IF rn=Used(y) THEN
550
NEXT y
IF RevWord(rn)=1 THEN 550
Used(x)=rn:RevWord(rn)=1:Rev$(x)
=Word$(rn)
NEXT x
GOSUB TimeBar
COLOR 20,0:LOCATE 13,4:PRINT "TI
ME":COLOR 30,0
LINE(83,122)-(225,174),31,bf:LIN
E(73,112)-(215,164),27,bf:LINE(7
2,111)-(216,165),20,b
LINE(240,120)-(290,153),31,bf:LI
NE(230,110)-(280,143),CoL,bf
LINE(229,110)-(281,144),20,b:LIN
E(229,120)-(281,120),20
LOCATE 15,30:COLOR 20,CoL:PRINT
"POINTS":COLOR 30,0
GOTO RevLoop

ResetVaL:
Ln=35:FOR x=1 TO 12:jm(x)=0:Used
(x)=0:Jum$(x)="":NEXT
JumbLe$="":Unjum$="":Ln=35
560 c$=INKEY$
IF c$<>"" THEN 560
IF c$=CHR$(13) THEN 560
RETURN

TimeBar:
LINE(40,40)-(50,92),31,bf:LINE(3
5,35)-(45,90),CoL,bf
RETURN

RevLoop:
Lw=LEN(Rev$(rw)):LOCATE 1,10:PRI
NT "Press A Key To Begin.":b$=IN
KEY$
IF b$="" THEN RevLoop ELSE LINE(
40,0)-(260,10),0,bf:LINE(70,48)-
(260,100),0,bf
FOR x=1 TO Lw
570 rn=INT(RND*Lw+1):IF rn=0 THE
N 570
FOR y=1 TO Lw:IF rn=jm(y) THEN 5
70
NEXT y

jm(x)=rn:Jum$(x)=MID$(Rev$(rw),r
n,1)
NEXT x
FOR x=1 TO Lw:JumbLe$=JumbLe$+Ju
m$(x)+" ":NEXT:Cent=INT((24-(Lw*
2))/2)
COLOR 30,29:LOCATE 3,9+Cent:PRIN
T JumbLe$:COLOR 30,0
FOR x=1 TO Lw:s=70
600 a$=INKEY$
IF a$=CHR$(32) THEN
LOCATE 11,13+x:PRINT a$
x=x-1:IF x<1 THEN x=1
LOCATE 11,13+x
PRINT a$:a$="":GOTO 600
END IF
IF a$=CHR$(13) THEN a$="":GOTO 6
00
LOCATE 11,13+x:PRINT UCASE$(a$);
:COLOR 25,0:PRINT "_":COLOR 30,0

IF a$="" THEN
LINE(35,INT(Ln/25)+32)-(45,INT(L
n/25)+32),0:Ln=Ln+1
LINE(46,INT(Ln/25)+37)-(50,INT(L
n/25)+37),0
IF Ln>1460 THEN 660 ELSE 600
END IF
Rguess$(x)=a$:a$="":NEXT
FOR x= 1 TO Lw:Unjum$=Unjum$+Rgu
ess$(x):NEXT
IF UCASE$(Unjum$)=Rev$(rw) THEN
650 ELSE 670
650 LOCATE 8,11:PRINT" That's Co
rrect! ":FOR x=1000 TO 2500 STEP
100:SOUND x,1,150,1:NEXT:Score=S
core+20:UcoL=30:GOSUB PrintIt:GO
TO 680
660 LOCATE 8,11:PRINT"Sorry, Out
Of Time!":SOUND 100,5,250,1:UcoL
=20:GOSUB PrintIt:GOTO 680
670 LOCATE 8,11:PRINT"That Is In
correct.":SOUND 100,5,250,1:UcoL
=20:GOSUB PrintIt:GOTO 680
680 LOCATE 17,31:COLOR 30,CoL:PR
INT Score
700 GOSUB ResetVaL:COLOR 30,0:GO
SUB TimeBar:rw=rw+1:IF rw<=5 THE
N LINE(39,14)-(279,25),29,bf:GOT
O RevLoop

EndLoop:
FOR x= 1 TO 3000:NEXT
CLS:LOCATE 7,3:PRINT "Total Numb
er of Revival Points":LOCATE 9,1
6:PRINT Score
FOR x= 1 TO 5000:NEXT
WINDOW CLOSE 3
WINDOW OUTPUT 2
GOSUB Score
GOSUB ChangePLay
```

```
RETURN
PrintIt:
LINE(39,14)-(279,25),29,bf
FOR pit=0 TO Lw-1
COLOR 30,29:LOCATE 3,9+Cent+(pit
*2):It$=MID$(Rev$(rw),pit+1,1):I
t$=It$+" "
PRINT It$:NEXT
FOR j=1 TO 4000:NEXT
LOCATE 15+rw,13:COLOR UcoL,27:PR
INT Rev$(rw)
pit=0
RETURN

MenuSet:
MENU 1,0,1, " "
MENU 2,0,1, " "
MENU 3,0,1, " "
MENU 4,0,1, " "
RETURN

CoLorData:
DATA 0,0,0,1,0,0,.3,0,0,0,0,1,0,
0,.3,1,.5,0,.7,0,1,.5,.2,0,.6,.6
,.6
DATA .7,.1,0,1,1,1,0,.1,0.02

WordData:
DATA "FORTHRIGHT","CONSULTANT","
REMEMBER","HIGHLIGHT","AUDIBLE"
DATA "PERSONAL","ADVANCED","EXEC
UTE","VARIATION","RESULTS"
DATA "FREQUENCY","EVALUATION","I
NDICATE","DURATION","ARGUMENT"
DATA "GENERATE","PRECISE","INFOR
MED","SYSTEM","ENCOUNTERED"
DATA "FOREIGN","CONSUMER","WASHT
UB","SUMMER","FEATURE"
DATA "TELEPHONE","PROGRAM","MICR
OWAVE","TEMPERATURE","CONCLUDE"
DATA "RECOMMEND","MOTIVATION","E
CONOMICS","CAPTAIN","DOUBTLESS"
DATA "CONVENIENCE","MINIMIZE","T
RANSMISSION","MECHANICAL","CONDI
TION"
DATA "ELECTRICAL","CHEMICAL","EN
GINEER","EFFECTIVE","HORIZONTAL"

DATA "CONCRETE","LANGUAGE","FUNC
TION","STARTLE","KNOWLEDGE"
                                    ©
```


Stringing The Atari Screen

Rhett Anderson

If you envy the speed with which machine language programmers can manipulate the graphics screen, type in this program to see just how fast BASIC can be.

For years, Atari BASIC programmers have been using strings to manipulate player-missile graphics. Atari BASIC strings are ideally suited for this task. While most BASICs limit the size of strings to 255 characters, Atari BASIC allows huge strings. In addition, Atari BASIC makes substring manipulation very easy.

Even if you are familiar with this trick, you may not have realized that it is possible to treat the entire display memory (or *playfield*, as Atari calls it) as a string. This technique allows you to do some things that are otherwise very slow in Atari BASIC. Among these are

- Erasing or filling the screen or part of the screen with any solid color or pattern.
- Duplicating part of the screen.
- Fast swapping of full or partial screen displays.

Although I have used GRAPHICS 7 for this program, these techniques are easily applied to all the graphics modes—even the text modes.

Typing It In

The program found at the end of this article is written entirely in BASIC. This program assumes that the first variable used will be placed in the first slot in the variable name table. To insure that this is the case, be sure to type NEW before entering the program. Type it in and save it to tape or disk. If you forget to type NEW, or if the program does not operate properly, save the program to tape or disk with the LIST command (see your BASIC manual for an example). Then type NEW and use the ENTER command to retrieve the program. You may now save the program again with the SAVE command.

The program is broken up into several sections. You can use any of these sections in your own programs. Just be sure that lines 10–60 are at the beginning of your program.

You must be sure to dimension the strings you use to the proper size. The amount of memory used by the screen (and therefore by the string) depends upon the screen mode. The size of the screen can be found with this formula:

size = (number of bytes needed per scan line) * (number of pixels available vertically)

The following table shows how many bytes are needed per scan line for the most common graphics modes:

Modes	Bytes Per Scan Line
0,7,8,9,10,11,15	40
1,2,5,6	20
3,4	10

For example, a GRAPHICS 0 screen would require 40 * 24 bytes of storage, while a GRAPHICS 7 screen requires 40 * 80.

As written, the program points

PRINTER PACKAGE SPECTACULAR

ALL PRINTER PACKAGES INCLUDES - NO SURCHARGES ON CREDIT CARDS - FREE SHIPPING IN CONTINENTAL USA
APPROPRIATE INTERFACE DEVICE FROM THE HONEST MAIL ORDER COMPANY

ComputAbility
Consumer Electronics

Panasonic

PANASONIC 1080i PRINTER PKG. WITH APPROPRIATE INTERFACE DEVICE

IBM PC/PARALLEL CABLE	$175
ATARI ST/PARALLEL CABLE	$175
AMIGA/PARALLEL CABLE	$189
C-64/128/XETEC SUPER GRAPHIC	$209
ATARI 8-BIT/SUPRA 1150	$199
APPLE 2E/PRO GRAPPLER	$249

NO SURCHARGES ON CREDIT CARDS
FREE SHIPPING IN CONTINENTAL USA

PANASONIC 1091i PRINTER PKG WITH APPROPRIATE INTERFACE DEVICE

IBM PC/PARALLEL CABLE	$209
ATARI ST/PARALLEL CABLE	$209
AMIGA/PARALLEL CABLE	$225
C-64/128/XETEC SUPER GRAPHIC	$249
ATARI 8-BIT/SUPRA 1150	$235
APPLE 2E/PRO GRAPPLER	$299

NO SURCHARGES ON CREDIT CARDS
FREE SHIPPING IN CONTINENTAL USA

PANASONIC 1092i PRINTER PKG WITH APPROPRIATE INTERFACE DEVICE

IBM PC/PARALLEL CABLE	$329
ATARI ST/PARALLEL CABLE	$329
AMIGA/PARALLEL CABLE	$339
C-64/128/XETEC SUPER GRAPHIC	$369
ATARI 8-BIT/SUPRA 1150	$359
APPLE 2E/PRO GRAPPLER	$399

NO SURCHARGES ON CREDIT CARDS
FREE SHIPPING IN CONTINENTAL USA

star

Lower Prices

NX-10 PRINTER PKG WITH APPROPRIATE INTERFACE DEVICE

IBM PC/PARALLEL CABLE	~~$199~~	$185
ATARI ST/PARALLEL CABLE	~~$199~~	$185
AMIGA/PARALLEL CABLE	~~$269~~	$199
C-64/128/XETEC SUPER GRAPHIC	~~$249~~	$219
ATARI 8-BIT/SUPRA 1150	~~$269~~	$209
APPLE 2E/PRO GRAPPLER	~~$269~~	$259

NO SURCHARGES ON CREDIT CARDS
FREE SHIPPING IN CONTINENTAL USA

NX-15 PRINTER PKG WITH APPROPRIATE INTERFACE DEVICE

IBM PC/PARALLEL CABLE	$349
ATARI ST/PARALLEL CABLE	$349
AMIGA/PARALLEL CABLE	$359
C-64/128/XETEC SUPER GRAPHIC	$369
ATARI 8-BIT/SUPRA 1150	$379
APPLE 2E/PRO GRAPPLER	$409

NO SURCHARGES ON CREDIT CARDS
FREE SHIPPING IN CONTINENTAL USA

NB-24/10 PRINTER (24 WIRE) PKG WITH APPROPRIATE INTERFACE DEVICE

IBM PC/PARALLEL CABLE	$459
ATARI ST/PARALLEL CABLE	$459
AMIGA/PARALLEL CABLE	$469
C-64/128/XETEC SUPER GRAPHIC	$499
ATARI 8-BIT/SUPRA 1150	$489
APPLE 2E/PRO GRAPPLER	$529

NO SURCHARGES ON CREDIT CARDS
FREE SHIPPING IN CONTINENTAL USA

commodore

64C COMPUTER
1541C DISK DRIVE
1802C MONITOR
1581C DISK DRIVE
C-128 COMPUTER
1571 DISK DRIVE
1902 MONITOR
C-128 D COMPUTER

CALL FOR INDIVIDUAL & SUPER PACKAGE PRICES

1750 RAM EXPANDER
1764 RAM EXPANDER
1351 MOUSE
AVATEX 1200 HC MODEM
AVATEX 2400 HC MODEM
I CON TROLLER

CALL FOR CURRENT PRICES

COMMODORE PC-10
PC COMPATIBLE COMPUTER

CALL FOR CURRENT PRICES

AMIGA

COMPUTER SYSTEM
CALL FOR CURRENT PRICES & AVAILABILITY

EPSON®

PRINTER PACKAGES
PRICES TO LOW TO ADVERTISE

ATARI

520 ST & 1040 ST
SYSTEM PACKAGES
CALL FOR LASTEST PRICES

IBM

Amnisia	29.95
Art Gallery 1	24.95
Art Gallery2	18.95
Balance of Power	31.95
Bop-N-Wrestle	19.95
Certificate Maker	38.95
Championship Baseball	25.95
Championship Golf	25.95
Chessmaster 2000	25.95
Chuck Yeager F.S.	26.95
Clip Art 1,3	19.95
Clip Art 2	25.95
Commando	25.95
Conflict in Vietnam	25.95
Crusade in Europe	25.95
Decision in the Desert	25.95
Defender of the Crown	25.95
Destroyer	25.95
F-15 Strike Eagle	21.95
Flight Simulator II	38.95
Grand Slam Bridge	39.95
Graphics Library 1 or 2	21.95
Hitchhikers Guide to the Galaxy	19.95
Hollywood Hijinx	25.95
Infiltrator	19.95
Isqur Portfolio	149.95
Jet	33.95
Kings Quest I, II, or III	31.95
Leather Goddesses of Phobos	25.95
Newsmaster	62.95
Newsroom	37.95
One on One	10.95
Orbiter	31.95
Pawn	29.95
P.C. Powerpack	77.95
Phantasie	25.95
Police Quest	CALL
Printmaster	36.95
Print Shop	37.95
P.S. Companion	31.95
Radio Baseball	25.95
Roadwar 2000	25.95
Seven Cities of Gold	10.95
Shanghai	22.95
Silent Service	21.95
Space Quest	31.95
Starflight	32.95
Sub Battle	25.95
Super Star Hockey	25.95
Tag Team Wrestling	25.95
Tenth Frame	29.95
Two on Two Basketball	25.95
Universe II	47.95
Where in the U.S.A is Carmen Sandiego	29.95
Where in the World is Carmen Sandiego	25.95
World Tour Golf	32.95

C64

Bard's Tale	25.95
Bard's Tale 2	25.95
Basic Compiler 64	25.95
Cad Pak 64	25.95
Carmen U.S.A.	29.95
Certificate Maker	30.95
Champ. Baseball	19.95
Chart Pak	25.95
Chessmaster 2000	25.95
Copy II 64	27.95
Data Manager II	25.95
Defender of the Crown	22.95
Desk Pak L	22.95
Desk Pak 2	32.95
Famous Courses	14.95
Fast Load	22.95
Final Cartridge	39.95
Flight Simulator II	33.95
Font Master II	29.95
Geocalc	34.95
Geo Chart	27.95
Geodex	25.95
Geofile	34.95
Geopublish	34.95
Geos	37.95
Geos Inside & Out	16.95
Gunship	22.95
Hardball	18.95
Hollywood Hijinx	22.95
Leaderboard	24.95
Leather Goddess	22.95
Legacy/Ancients	20.95
Mail Order Monster	10.95
Mind Mirror	10.95
Movie Maker	10.95
Newsroom	30.95
Partner 64	39.95
Pawn	25.95
Pegasus	23.95
Pirates	25.95
Printmaster Plus	21.95
Printshop	26.95
P.S. Companion	22.95
S.D.L	22.95
Skyfox	10.95
Starglider	25.95
Sub Battle	22,95
ST. Sports Baseball	25.95
ST. Sports Basketball	25.95
Super C Compiler	39.95
Super Pascal	39.95
Super Pak 2/64	59.95
Super Star Hockey	22.95
Swiftcalc Sideways	25.95
2 on 2 Basketball	22.95
Ultima 1	25.95
War In S. Pacific	30.95
Wordwriter/Spell	32.95
Wid Class Leader	25.95
World Tour Golf	23.95

C128

Abacus Book	CALL
Basic Compiler 128	39.95
Bureaucracy	22.95
Cad Pak 128	39.95
Chart Pak	25.95
Cobol 128	39.95
Datamanager128	42.95
Fleet System 4	46.95
Fontmaster 128	36.95
Geos 128	44.95
Mach 128	32.95
Merlin 128	69.95
Multiplan 128	27.95
Paperclip IV128	CALL
Partner 128	42.95
Perfect Writer	49.95
Pocket Filer 2	34.95
Pocket Planner 2	36.95
Pocket Writer 2	36.95
Speed Term 128	25.95
Superbase 128	57.95
Super C Compiler	39.95
Super Pak2/128	59.95
Super Pascal 128	39.95
Superscript 128	46.95
Swiftcalc 128	42.95
Sylvia Porter 128	42.95
Trinity	25.95
Visastar 128	54.95
Visawrite 128	54.95
Wordwriter 128	42.95

ATARI 8-BIT

Animation Station	59.95
Archon	10.95
Archon II	10.95
Atariwriter Plus	39.95
Bop-N-Wrestle	19.95
Chessmaster 2000	25.95
F-15 Strike Eagle	22.95
Gettysburg	39.95
Hardball	18.95
Home Planetarium	27.95
Leaderboard	24.95
Leather Goddess	22.95
Mail Order Monsters	10.95
Megafont II	16.95
Movie Maker	10.95
Music Studio	22.95
Page Designer	18.95
Print Shop	28.95
P.S. Companion	22.95
P.S. Interface	18.95
Racing Destruction	10.95
Rubber Stamp	18.95
Silent Service	22.95
Syncalc	32.95
Synfile	32.95
Top Gunner	16.95
Typesetter	21.95
Ultima II	31.95
Ultima IV	38.95
Wizards Crown	27.95
Zork Trilogy	39.95

APPLE

Airheart	22.95
Animate	44.95
Autoduel	32.95
Bards Tale	29.95
Bards Tale II	32.95
Bop-N-Wrestle	19.95
Certificate Maker	31.95
Certificate Maker Lib #1	22.95
Clip Art 1 or 3	19.95
Clip Art 2	25.95
Destroyer	25.95
F-15 Strike Eagle	22.95
Fight Night	21.95
Graphics Lib 1,2or3	16.95
Hardball	21.95
Infiltrator	19.95
Jet	27.95
Newsroom	37.95
On Balance	64.95
Pawn	29.95
Pegasus	25.95
Phantasie 1,2 or 3	25.95
Print Shop	31.95
Silent Service	22.95
Starglider	29.95
Street Sports Baseball	25.95
Street Sports Basketball	25.95
Tag Team Wrestling	22.95
Where in the U.S.A. is Carmen Sandiego	29.95
Where in the World is Carmen Sandiego	25.95

ATARI ST

Abacus Books	CALL
Aegis Animator	48.95
Art Gallery 1or2	18.95
Artic Fox	26.95
Assempro	39.95
Back-up	24.95
Balance of Power	32.95
Certificate Maker	31.95
Chessmaster 2000	32.95
Colonial Conquest	25.95
CZ Droid	69.95
Dac Easy Accounting	44.95
Dac Payroll	32.95
Datamanager ST	48.95
Datatrieve	32.95
DB Man	96.95
Deep Space	29.95
Defender of the Crown	32.95
Degas Elite	52.95
Desk Cart	72.95
Easy Draw	48.95
EZ Track	39.95
Flash	27.95
Flight Simulator	33.95
GFA Basic	48.95
GFA Compiler	48.95
GFA Draft	62.95
GFA Vector	36.95
Gold Runner	24.95
Hardball	27.95
Karate Kid II	24.95
Kid Talk	31.95
Labelmaster Elite	27.95
LDW Basic	44.95
Major Motion	24.95
Mark Williams	114.95
Math Talk	31.95
Megafont ST	24.95
Music Studio	32.95
Pawn	29.95
P.C. Ditto	84.95
Personal Pascal	49.95
Phantasie 1,2or3	25.95
PM Interface	18.95
Printmaster Plus	24.95
Publishing Partner	99.95
Roadwar 2000	25.95
Rubber Stamp	24.95
S.D.L	32.95
Shanghai	25.95
Silent Service	25.95
Skyfox	29.95
Speller Bee	31.95
Starglider	29.95
Sundog	24.95
Swiftcalc ST	48.95
Thunder	26.95
Time Bandits	24.95
Typesetter Elite	31.95
Wordwriter ST	48.95

AMIGA

Aegis Animator	84.95
Aegis Draw	159.95
Analyze 2.0	94.95
Bards Tale	32.95
Bureaucracy	25.95
Cambridge Lisp	129.95
CBM Textcraft	69.95
Championship Baseball	29.95
Championship Golf	25.95
Chessmaster 2000	29.95
Defender of the Crown	32.95
Deja Vu	32.95
Deluxe Music 2.0	64.95
Deluxe Print II	84.95
Deluxe Paint	84.95
Deluxe Video 2.0	84.95
Earl Weaver Baseball	32.95
Fairy Tale Adventure	31.95
First Shapes	31.95
Flight Simulator II	33.95
Flip Side	31.95
Fortran 77	169.95
GFL Football	29.95
Goldspell	27.95
Grand Slam Tennis	31.95
Grid Iron	44.95
Impact	54.95
Instant Music	32.95
Kampgruppe	39.95
Kid Talk	31.95
Kinderama	31.95
Kings Quest 1,2 or 3	32.95
Logistix	89.95
Marble Madness	32.95
Math Talk	31.95
Mean 18	27.95
Microleague Baseball	39.95
Modula 2	64.95
Music Studio	32.95
On Line	42.95
Organize	62.95
Page Setter	89.95
Pawn	29.95
Phantasie	25.95
Print Master Plus	31.95
Publisher	119.95
Scribble	62.95
S.D.L	32.95
Shanghai	25.95
Silent Service	25.95
Sinbad	32.95
Skyfox	26.95
Sonix	54.95
Starglider	29.95
Superbase Personal	89.95
Tass Times	25.95
Two on Two Basketball	29.95
VIP Professional	169.95
Winter Games	25.95
World Games	25.95

SINCE 1982

ComputAbility
Consumer Electronics

P.O. BOX 17882, MILWAUKEE, WI 53217
ORDER LINES OPEN
Mon-Fri 9am-9pm CST Sat 11am-5pm
To Order Call Toll Free

800-558-0003

For Technical info, Order Inquiries, or for Wisc. Orders

414-357-8181

TELEX NUMBER 9102406440
(ANSERBACK=COMPUT MILW UQ)

No surcharge for Mastercard or Visa

ORDERING INFORMATION: Please **specify system.** For fast delivery send cashier's check or money order. Personal and company checks allow 14 business days to clear. School PO's welcome. **C.O.D. charges are $3.00.** In Continental U.S.A. include $3.00 for software orders. 5% shipping for hardware, minimum $5.00. MasterCard and Visa orders please include card #, expiration date and signature. WI residents please include 5% sales tax. HI, AK, FPO, APO, Puerto Rico and Canadian orders please add **6% shipping, minimum $6.00.** All other **foreign orders add 15% shipping, minimum $15.00.** All orders shipped outside the Continental U.S.A. are shipped first class insured U.S. mail. If foreign shipping charges exceed the minimum amount you will be charged the additional amount to get your package to you quickly and safely. All goods are new and include factory warranty. Due to our low prices all sales are final. **All defective returns must have a return authorization number.** Please call (414) 357-8181 to obtain an R.A. # or your return will not be accepted. Prices and availability subject to change without notice.

the variable A$ to the screen area. You may change the name of the variable in your own programs, but it must be the first variable used in the program.

Once the string is reassigned to the screen, any operations you perform on the string will show up on the screen.

Addressing the screen as a string may seem awkward at first, but it is incredibly fast. Once you understand it, you'll be able to write programs that do screen manipulations with machine language speed. In the case of the GRAPHICS 7 screen, a byte represents four pixels. This makes color selection rather odd. Instead of using the COLOR statement to select colors 0–3, the values 0, 85, 170, and 255 represent the four different colors.

Although the memory representation makes working with solid colors rather arcane, it makes working with patterns a breeze (for an example, see the Atari version of *Laser Chess*™ in the July 1987 issue of COMPUTE!). In some graphics modes such as mode 15 (available only on XLs and XEs), the two colors blend well enough to appear to be a new color. A patterned screen fill is done in the demo program in lines 320–360. Use this table to replace the numbers in lines 320 and 340 for other color combinations.

Colors	Values
0 and 1	17, 68
0 and 2	34, 136
0 and 3	51, 204
1 and 2	102, 153
1 and 3	119, 221
2 and 3	187, 238

For instance, if you want to use colors 2 and 3 in the demo, replace the 102 in line 320 with 187 and replace the 153 in line 340 with 238.

For instructions on entering this program, please refer to "COMPUTE!'s Guide to Typing In Programs" elsewhere in this issue.

Stringing The Atari Screen

```
NO 1 REM COPYRIGHT 1987  COM
     PUTE! PUBLICATIONS, INC
     .  ALL RIGHTS RESERVED.
LA 2 PRINT "{CLEAR}":POSITIO
     N 12,8:PRINT "Copyright
      1987":POSITION 7,10:PR
     INT "COMPUTE! Publicati
     ons, Inc."
GN 3 POSITION 10,12:PRINT "A
     ll Rights Reserved"
PN 4 FOR I=1 TO 1500:NEXT I
JO 10 GRAPHICS 7:CLR :DIM A$
     (40*80),COL(4),B$(40*8
     0),TEMP$(40*80):A$(40*
     80)=CHR$(0)
NL 20 VTAB=PEEK(134)+PEEK(13
     5)*256
MB 30 ATAB=PEEK(140)+PEEK(14
     1)*256
ND 40 OFFS=PEEK(88)+PEEK(89)
     *256-ATAB
CO 50 HI=INT(OFFS/256):LO=OF
     FS-HI*256
JH 60 POKE VTAB+2,LO:POKE VT
     AB+3,HI
JE 70 REM ** SOLID COLOR FIL
     L **
NJ 75 PRINT "SOLID COLOR FIL
     L":GOSUB 1000
PD 80 FOR T=0 TO 4:READ COL:
     COL(T)=COL
BK 90 A$(1,1)=CHR$(COL)
BL 100 A$(2)=A$
NH 105 GOSUB 1000
CF 110 NEXT T
FO 120 DATA 0,85,170,255,0
PB 200 REM ** FILL PART OF S
     CREEN **
HE 210 PRINT "PARTIAL SCREEN
      FILL":FOR T=0 TO 3:W
     HERE=400*T+1:REM USE
     400 BECAUSE 10 LINES
     OF 40 BYTES EACH
HO 220 A$(WHERE,WHERE)=CHR$(
     COL(T))
AK 230 A$(WHERE+1,WHERE+399)
     =A$(WHERE,WHERE+398):
     REM DO 10 LINES
NL 235 GOSUB 1000
CJ 240 NEXT T
ME 300 REM ** PATTERN FILL *
     *
PD 310 PRINT "PATTERNED SCRE
     EN FILL"
NK 315 GOSUB 1000
PL 320 A$(1,1)=CHR$(102)
KO 330 A$(2,40)=A$:REM 40 BE
     CAUSE USING GR. 7
CI 340 A$(41,41)=CHR$(153):R
     EM NEXT LINE DOWN
KA 350 A$(42,80)=A$(41)
AH 360 A$(81)=A$:REM DUPLICA
     TE FIRST 2 LINES DOWN
      ENTIRE SCREEN
NL 370 GOSUB 1000
MM 400 REM ** SCREEN DUPLICA
     TION **
IN 410 PRINT "DUPLICATE PART
      OF SCREEN"
NL 415 GOSUB 1000
CN 420 FOR X=0 TO 159:COLOR
     X/54
OK 430 PLOT X,0:DRAWTO X/2,7
     9
CP 440 NEXT X
AN 450 FOR I=1 TO 3
MO 560 A$(I*800+1,I*800+800)
     =A$:REM 800 FOR 20 LI
     NES
CE 570 NEXT I
MO 580 GOSUB 1000
PJ 600 REM ** FAST SCREEN SW
     APPING **
IB 610 PRINT "SCREEN FLIPPIN
     G"
KA 620 B$=A$
CE 630 FOR Y=1 TO 30:COLOR R
     ND(0)*3+1
JG 640 PLOT RND(0)*159,RND(0
     )*79:DRAWTO RND(0)*15
     9,RND(0)*79
DD 650 NEXT Y
EC 660 FOR K=1 TO 30
BC 670 TEMP$=A$:A$=B$:B$=TEM
     P$
CI 680 NEXT K
OA 690 GOSUB 1000
FJ 700 REM ** PARTIAL SCREEN
     SWAPPING **
II 710 PRINT "PARTIAL SCREEN
      SWAPPING":? " (TOP W
     ITH BOTTOM"
EG 720 FOR K=1 TO 30:B$=A$(1
     ,40*40):A$(1,40*40)=A
     $(40*40+1):A$(40*40+1
     )=B$:NEXT K
IE 800 REM ** ANIMATION **
AK 810 PRINT "ANIMATION"
GB 820 A$(1,1)=CHR$(0):A$(2)
     =A$
BJ 830 FOR T=0 TO 3
LE 840 COLOR 1:FOR Y=1 TO 10
     :COLOR Y:PLOT T+4,Y+T
     *20:DRAWTO T+30+4,Y+T
     *20:NEXT Y
DA 850 NEXT T
DD 855 GOSUB 1000
KG 860 B$=A$
GG 870 A$(1,1)=CHR$(0):A$(2)
     =A$
FA 880 X=10:Y=10:DX=1:DY=1
EG 890 X=X+DX:Y=Y+DY
LE 900 IF X=100 THEN DX=-1
CM 910 IF X=5 THEN DX=1
IN 920 IF Y=60 THEN DY=-1
DB 930 IF Y=6 THEN DY=1
AH 940 H=INT(X/4):L=X-H*4
LG 950 A$(H+40*Y+1,H+40*Y+48
     0)=B$(L*800+1,L*800+4
     80)
HJ 960 GOTO 890
KB 1000 FOR DELAY=1 TO 100:N
     EXT DELAY
KC 1010 RETURN
```

©

SOFTWARE DISCOUNTERS OF AMERICA

For Orders Only—1-800-225-7638
PA Orders—1-800-223-7784
Customer Service 412-361-5291

- Free shipping on orders over $100 in continental USA
- No Surcharge for VISA/MasterCard
- Your card is not charged until we ship

IBM SOFTWARE

ACCESS
10th Frame Bowling ... $29
ACCOLADE
Ace of Aces ... $24
Mean 18 Golf ... $30
Mean 18 Famous Course Disk #2 ... $14
Mean 18 Famous Course Disks #3 & 4 ... $23
ACTIVISION
Hacker ... $9.88
Hacker 2 ... $24
Portal ... $27
Shanghai ... $23
Tass Times ... $23
ARTWORX
Bridge 5.0 ... $19
Strip Poker ... $25
Data Disk #1 Female ... $14
Data Disk #2 Male ... $14
Data Disk #3 Female ... $14
Linkword Languages:
Dutch ... $19
French ... $19
German ... $19
Italian ... $19
Russian ... $19
Spanish ... $19
AVALON HILL
Dr. Ruth's Game ... $19
Super Bowl Sunday ... $21
S.B.S. Champs Disk ... $14
S.B.S. '85 Team Disk ... $14
S.B.S. '86 Team Disk ... $14
BRODERBUND
Ancient Art of War ... $27
Ancient Art of War at Sea ... $27
Bank St. Writer Plus ... $49
Business Week's Business Advantage ... $32
Carmen Sandiego USA ... $29
Carmen Sandiego World ... $25
Karateka ... $21
Memory Mate ... $44
Print Shop ... $36
P.S. Companion ... $32
P.S. Graphics Library Holiday Edition ... $21
P.S. Graphics Library #1 or #2 ... $21 Ea.
P.S. Paper Refill ... $14
Search & Destroy ... $9.88
Toy Shop ... $32
Toy Shop Refill ... $19
Type! ... $30
CENTRAL POINT
Copy 2 ... $23
PC Option Board ... $79
PC Tools ... $23
DAC
Easy Accounting 2.0 ... $44
Easy Payroll 2.0 ... $32
DATA EAST
Commando ... $24
Tag Team Wrestling ... $24
DAVIDSON
Algeblaster ... $30
Grammar Gremlins ... $30
Homeworker ... $54
Math Blaster ... $30
Spell It ... $30
Word Attack ... $30
ELECTRONIC ARTS
Software Classic Series:
Archon ... $9.88
Mind Mirror ... $9.88
Music Const. Set ... $9.88
One-on-One ... $9.88
Pinball Const. Set ... $9.88
Seven Cities Gold ... $9.88
Super Boulder Dash ... $9.88
ELECTRONIC ARTS
Arctic Fox ... $23
Chessmaster 2000 ... $26
Chuck Yeager's Advanced Flight Simulator ... $26
Empire ... Call
Grand Slam Bridge ... $39
Marble Madness ... $23
Scrabble ... $26
Starfleet 1 or 2 ... $32 Ea.
Starflight ... $32
World Tour Golf ... $32
ENLIGHTENMENT
Guide to Chess Openings $39
Paul Whitehead Teaches Chess ... $39
EPYX
Create A Calendar ... $19
Destroyer ... $24
Pitstop 2 ... $9.88
P.S. Graphics Scrapbook:
#1 Sports ... $16
#2 Off the Wall ... $16
#3 School ... $16
Sub Battle Simulator ... $24
Summer Games 2 ... $24
World Games ... $24
GAMESTAR
Champ. Baseball ... $24
Champ. Basketball ... $24
Champ. Golf ... $24
GFL Ch. Football ... $24
HAYDEN
Sargon 3 Chess ... $24
HI-TECH EXPRESSIONS
Award Ware ... $9.88
Card Ware ... $6.88
Party Ware ... $9.88
INFOCOM
Bureaucracy ... $24
Hitchhikers Guide ... $19
Hollywood Hijinx ... $24
Leather Goddesses ... $24
Stationfall ... $24
The Lurking Horror ... $24
Zork 1 ... $24
Zork Trilogy ... $44
INTRACORP
Business Card Maker ... $39
MECA
Andrew Tobias: Managing Your Money 2.0 ... $129
MICROPROSE
F-15 Strike Eagle ... $21
Gunship ... $32
Silent Service ... $21
MINDSCAPE
Balance of Power ... $30
Bop & Wrestle ... $19
Crossword Magic ... $30
Defender of the Crown $24
Infiltrator ... $19
Perfect Score SAT ... $44
Super Star Ice Hockey ... $24
MONOGRAM
Dollars & Sense ... $109
SHARE DATA
Jeopardy ... $8.88
Wheel of Fortune ... $8.88
SIERRA
Black Cauldron ... $24
King's Quest 1, 2, or 3 ... $30 Ea.
Leisure Suit Larry ... $24
Space Quest ... $30
SIMON & SHUSTER
Chem Lab ... $24
Promethean Prophecy ... $24
Real Estate Investment & Management ... $65
Typing Tutor 4 ... $30
Websters:
Spelling Checker ... $39
Thesaurus ... $44
SPECTRUM HOLDBYTE
Falcon F-16 Sim ... $32
SPINNAKER
T-Shirt Shop ... $32
T-Shirt Art #1 ... $14
SPRINGBOARD
Certificate Maker ... $36
C.M. Library #1 ... $21
Early Games ... $21
Newsroom Pro ... $89
N.R. Clip Art #1 or 3 $19 Ea.
N.R. Clip Art #2 ... $24
P.S. Graphics Expander $24
SSI
Battle of Antietam ... $30
Gettysburg ... $36
Kampfgruppe ... $36
Phantasie ... $24
Rings of Zilfin ... $24
Roadwar 2000 ... $24
Shard of Spring ... $24
Wizard's Crown ... $24
SUBLOGIC
Jet ... $32
TELARIUM
Amazon ... $9.88
Dragonworld ... $9.88
Fahrenheit 451 ... $9.88
Nine Princes ... $9.88
Perry Mason ... $9.88
Rendezvous ... $9.88
THUNDER MOUNTAIN
Top Gun ... $6.88
TIMEWORKS
*Data Manager ... $39
Evelyn Wood Reader ... $39
Partner ... $39
*Swiftcalc w/Sideways ... $39
Sylvia Porter's Financial Planner ... $59
*Word Writer ... $39
*All 3 in 1 Power Pack $79
UNISON WORLD
Fonts & Borders ... $23
Newsmaster ... $59
Printmaster Plus ... $36
Art Gallery 1 or 2 ... $19 Ea.
WEEKLY READER
Stickybear Math ... $23
Stickybear Numbers ... $23
Stickybear Reading ... $23
Stickybear Spellgrabber $23
Stickybear Typing ... $23

APPLE II SERIES

ACCESS
Famous Course Disk #1 for WCLB ... $14
Triple Pack: BH1, BH2, Raid Over Moscow ... $14
World Class Leader Board ... $25
ACCOLADE
Hardball ... $19
Hardball GS ... $29
Killed Until Dead ... $19
Mean 18 Golf GS ... $29
ACTIVISION
Aliens ... $23
Draw GS ... $47
Gamemaker Library Disks: Sci-Fi or Sports ... $16 Ea.
Gamemaker ... $30
Ghostbusters ... $9.88
Hacker 2 ... $24
Hacker 2 GS ... $29
Might & Magic ... $30
Music Studio GS ... $47
Paintworks Clip Art GS ... $19
Paintworks GS ... $47
Portal ... $27
Shanghai ... $21
Shanghai GS ... $29
Tass Times GS ... $24
Writer's Choice Elite GS ... $47
ARTWORX
Bridge 5.0 ... $19
International Hockey ... $19
Strip Poker ... $19
Data Disk #1 (Female) ... $14
Data Disk #2 (Male) ... $14
BAUDVILLE
Award Maker Plus ... $24
French Paper Pack ... $9.88
Video Vegas ... $19
BRODERBUND
Animate 128K ... $42
Bank St. Writer Plus 128K ... $47
Carmen San Diego USA ... $27
Carmen San Diego World ... $24
Choplifter ... $9.88
Dazzle Draw 128K ... $34
Fantavision ... $30
Karateka ... $21
Loderunner ... $21
On Balance 128K ... $59
Print Shop Enhanced ... $29
Print Shop Companion ... $23
Print Shop GS ... $39
P.S. Graphics Library #1, 2 or 3 ... $14 Ea.
P.S. Graphics Library Sampler Edition GS ... $23
Science Tool Kit ... $39
Toy Shop ... $32
Type! ... $27
Wings of Fury ... $23
CENTRAL POINT
Copy 2 ... $21
DAVIDSON
Algeblaster ... $27
Grammar Gremlins ... $27
Math Blaster ... $27
Spell It ... $27
Word Attack ... $27
DAC
Easy Accounting ... $44
DATA EAST
Commando ... $21
Tag Team Wrestling ... $21
ELECTRONIC ARTS
Age of Adventure ... $9.88
Archon 1 or 2 ... $9.88 Ea.
Arctic Fox ... $26
Bard's Tale GS ... $32
Bard's Tale 2 ... $32
Chessmaster 2000 ... $29
Deluxe Paint II GS ... $65
D. Paint Art #1 GS ... $19
D. Paint Art #2 GS ... $19
Earth Orbit Station ... $23
Instant Music GS ... $65
Lords of Conquest ... $9.88
Marble Madness 128K ... $23
Moebius ... $36
Movie Maker ... $9.88
Music Const. Set ... $9.88
Music Const. Set GS ... $32
One-on-One ... $9.88
Pegasus ... $23
Pinball Const. Set ... $9.88
Russia (The Great War) ... $26
Scrabble ... $26
Seven Cities of Gold ... $9.88
Skyfox ... $9.88
Star Fleet 1 ... $32
Super Boulder Dash ... $9.88
Ultima 4 ... $36
EPYX
Champ. Wrestling ... $24
Create A Calendar ... $19
Destroyer ... $24
Movie Monster ... $14
P.S. Graphics Scrapbook:
#1 Sports ... $14
#2 Off the Wall ... $14
#3 School ... $14
Sub Battle Simulator ... $24
Sub Battle Sim. GS ... $24
Summer Games 2 ... $24
Temple Apshai Trilogy ... $14
Winter Games ... $24
World Games ... $24
FIREBIRD
Elite ... $21
Guild of Thieves ... $24
Starglider ... $27
GAMESTAR
Champ. Baseball 128K ... $24
Ch. Basketball 128K ... $24
Champ. Football 128K ... $24
Top Fuel Eliminator 128K $19
HI-TECH EXPRESSIONS
Award Ware ... $9.88
Card Ware ... $6.88
Heart Ware ... $6.88
Party Ware ... $9.88
INFOCOM
Bureaucracy 128K ... $24
Hitchhiker's Guide ... $19
Hollywood Hijinx ... $24
Leather Goddesses ... $24
Stationfall ... $24
The Lurking Horror ... $24
Zork Trilogy ... $44
MECA
Andrew Tobias: Managing Your Money 2.0 ... $95
MICROLEAGUE
Baseball ... $24
Box Score Stats ... $14
General Mgr./Owner ... $19
1986 Team Disk ... $12
MICROPROSE
F-15 Strike Eagle ... $21
Silent Service ... $21
Silent Service GS ... $25
MINDSCAPE
Bop & Wrestle ... $19
Defender of the Crown GS ... $32
Infiltrator ... $19
Perfect Score SAT ... $44
MONOGRAM
Dollars & Sense 128K ... $69
PERSONAL CHOICE
Term Paper Writer 128K ... $32
SHARE DATA
Family Feud ... $8.88
Jeopardy ... $8.88
Wheel of Fortune ... $8.88
SPRINGBOARD
Certificate Maker ... $29
C.M. Library Vol. 1 ... $21
Newsroom ... $32
N.R. Clip Art #1 or #3 ... $17
N.R. Clip Art #2 ... $23
STYLEWARE
Multiscribe 2.0 128K ... $44
Multiscribe 2.0 GS ... $65
SSI
Battlecruiser ... $35
Eternal Dagger ... $24
Gettysburg ... $35
Phantasie 1, 2 or 3 ... $24 Ea.
Realms of Darkness ... $24
Rings of Zilfin ... $24
Roadwar 2000 ... $24
Shard of Spring ... $24
War in S. Pacific ... $35
Wizard's Crown ... $24
SUBLOGIC
Flight Simulator 2 ... $32
F.S. Scenery Disks ... Call
Jet ... $26
WEEKLY READER
Stickybear ABC's ... $23
Stickybear Math 1 or 2 $23 Ea.
Stickybear Numbers ... $23
Stickybear Printer ... $23
Stickybear Shapes ... $23
Stickybear Spellgrabber $23
Stickybear Typing ... $23
WINDHAM CLASSICS
Alice in Wonderland ... $9.88
Below the Root ... $9.88
Swiss Family Robinson $9.88
Treasure Island ... $9.88
Wizard of Oz ... $9.88

Commodore 64 & 128 Owners: Please read our full page ad in this month's Compute's Gazette!

P.O. BOX 111327—DEPT. CP—BLAWNOX, PA 15238

*Please Read The Following Ordering Terms & Conditions Carefully Before Placing Your Order: Orders with cashiers check or money order shipped immediately on in stock items. Personal & Company checks, allow 3 weeks clearance. No C.O.D.'s! Shipping: Continental U.S.A.—Orders under $100 add $3; free shipping on orders over $100. AK, HI, FPO, APO—add $5 on all orders. Canada & Puerto Rico—add $10 on all orders. Sorry, no other International orders accepted! PA residents add 6% sales tax on the total amount of order including shipping charges! REASONS FOR CALLING CUSTOMER SERVICE—412-361-5291 (1) Status of order or back order (2) If any merchandise purchased within 60 days from S.D. of A. is defective, please call for a return authorization number. Defective merchandise will be replaced with the same merchandise only. Other returns subject to a 15% restocking charge. After 60 days please refer to the manufacturers warranty included with the merchandise & return directly to the manufacturer. Customer service will not accept collect calls or calls on S.D. OF A.'s 800# order lines! Have you seen our on line catalog of 1000 software titles for Commodore, Atari, Apple, IBM and Amiga? It's on Compuserve's Electronic Mall—just type GO SDA and shopping for software will never be the same again! HOURS: Mon.-Fri. 9AM-5:30 PM, Sat. 10AM-4PM Eastern Time. Because this ad had to be written 2-3 mos. before it was published, prices & availability are subject to change! When sending a mail order, please specify make & model# of your computer!

Laser Chess™ Mixup

Last month's "CAPUTE!" column mistakenly stated that the Atari version of *Laser Chess* on the April–June COMPUTE! Disk could not be run from the disk menu. Actually, *Laser Chess* runs just fine when selected from the menu of the Atari disk. It is the Commodore 64 version of *Laser Chess* on the May–July Commodore edition of the COMPUTE! Disk that will not run properly when loaded from the menu. We regret any confusion this may have created.

The problem with the Commodore 64 version does not occur until a playing piece is destroyed by a laser blast. If the game is loaded from the COMPUTE! Disk menu, it will lock up at the end of the explosion animation sequence. This problem does not occur if the program is loaded directly. To run the Commodore 64 version of *Laser Chess* from the COMPUTE! Disk without using the menu, simply use the command LOAD "LASERCHESS.JUN",8 and type RUN.

Smart Alec

Line 1419 in Program 1 from this article for the Commodore 64 in the September issue (p. 81) contains a spurious question mark and is missing the final digit. The line should read as follows:

```
1419:F6 A9 00 99 A7 02 AA A5 01
```

IBM Monte Carlo Enhancements

The IBM version (Program 5, p. 35) of this card-game program from the September issue works properly as printed, but reader Robert St. John was disappointed at the lack of color and the fact that the program recognizes only lowercase letters. Other readers with color monitors may want to try his suggested changes and additions to enhance game play:

```
KN 110 KEY OFF:WIDTH 40:COLOR 1,
    2,2
BP 140 DIM C$(52),COL(52),V$(13)
    ,T$(4),CL(4)
IF 170 READ A:T$(I) = CHR$(A):RE
    AD CL(I):NEXT
JI 200 DATA 3,4,4,4,5,0,6,0
PK 230 COLOR CL(J):PRINT"     ";
    V$(I);T$(J);
KO 250 COLOR 1
HP 300 IF C$(P)="" THEN C$(P)=V$
    (J)+T$(I):COL(P)=CL(I):GO
    TO 330
QO 342 LOCATE 17,5:PRINT SPACE$(
    28)
PD 372 PRINT:PRINT TAB(11)"Press
     Q to quit"
MM 380 A$=INKEY$:IF A$="" THEN 3
    80
OP 382 IF A$="h" OR A$="H" THEN
    C=4:GOTO 420
LP 384 IF A$="m" OR A$="M" THEN
    C=5:GOTO 420
MF 386 IF A$="e" OR A$="E" THEN
    C=6:GOTO 420
JG 388 IF A$="q" OR A$="Q" THEN
    CLS:END
IM 390 GOTO 380
OE 450 COLOR COL(CT):PRINT "
    ";C$(CT);
GO 590 C$(I)=C$(I+1):COL(I)=COL(
    I+1)
FD 630 C$(I)=C$(I+1):COL(I)=COL(
    I+1)
OL 770 IF A$="q" OR A$="Q" THEN
    RUN
```

©

STation

The one STop shop

ORDERS ONLY
CALL TOLL FREE

1-800-CALL TRU

(1-800-225-5878)
ORDER STATUS, INFORMATION, R.A.s
CALL 1-313-595-0788

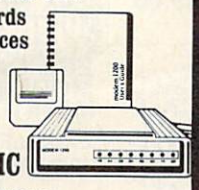
$569

Includes MS-DOS!

BLC PC512/1 100% IBM PC-XT compatibility and 512 K RAM. Includes 5¼" 360 KB floppy disk drive, low profile keyboard, high resolution output (720x350 pixels), centronics parallel and RS232 printer ports, six expansion slots. Includes abbreviated MS-DOS.

1200 BAUD MODEM KIT

Access Bulletin Boards & Information Services



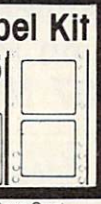
All Just $119

- Avatex 1200 HC
- ST or PC Software
- Compuserve Starter Kit
- RS-232 Cable M-F
- Phone Cord

Super Special Seagate
PC-XT HARD DRIVE KIT
$299 20 MEG
Complete with:
Western Digital Controller and Mounting Hardware

3½" Continuous Label Kit

- 500 3½" Disk Labels **$9.95**
- FREE Label Program
- Automatically Reads Disk and Makes Labels
- Create Database



THE NEW 520ST FM $489 Mono

WITH BUILT IN DRIVE
With 1 MEG Upgrade (Already Installed) Add $99.

Call for pricing on other models.

BONUS! FREE STARTER KIT

- Monitor Riser
- 3½" Label Kit
- Dust Cover
- 3½" Disk Head Cleaner

$50 VALUE

Panasonic Dot Matrix Printer

KXP-1080i $179.00
KXP-1091i $199.00

- Print speed (cps) KXP-1080i Draft 120
- Print speed (cps) KXP-1091i Draft 160

FREE STARTER KIT

- $68.00 value
- IBM-ST cable
- 200 pack paper
- Heavy duty metal printer stand
- Vinyl dust cover

BULK 3½ DISKS

$.99 SS DD QTY 25 **$1.29** DS DD QTY 25

ATARI ST SOFTWARE

Arctic Fox	27.95
Chessmaster 2000	30.95
Crystal Castles	19.95
Dac Easy Accounting	46.95
Data Manager ST	52.95
DB Man	119.95
DB Master One	39.95
Degas Elite	53.95
Dollars & Sense	66.95
Donald Duck's Playground	19.95
Easy Draw	53.95
First Word	34.95
Flight Simulator II	34.95
GFA Basic	52.95
GFA Basic Compiler	52.95
High Roller	32.95
Joust	18.95
Kings Quest I, II or III	32.95
Mark Williams C	119.95
Music Studio	39.95
Phantasie II	26.95
PrintMaster Plus	26.95
Psion Chess	39.95
Publishing Partner	99.95
S.D.I.	32.95
Skyfox	31.95
Space Quest	32.95
Swiftcalc ST	52.95
Time Bandits	26.95
VIP Professional	166.95
WordWriter ST	52.95

ST HARDWARE

Atari 1040ST	Call
Atari 314 Drive	199.00
Atari 354 Drive	105.00
Atari SC1224 Color Monitor	299.00
Atari SH204 Hard Drive	539.00
Atari SM124 Mono Monitor	119.00
Supra 20 Meg Hard Drive	549.00
Upgrade, EZ-Ram	159.00
Z-Time 1040	49.95
Z-Time 520	39.95

PRINTERS

Okimate 20 w/Plug-N-Print	198.00
Panasonic KX-P1080i	179.00
Panasonic KX-P1091i	199.00
Panasonic KX-P1092i	309.00
Panasonic KX-P1592	399.00
Panasonic KX-P1595	435.00
Panasonic KX-P3151	259.00
Panasonic KX-P3131	409.00

DUST COVERS

1040 ST	9.95
Gemini 10	6.95
IBM PC Keyboard	6.95
IBM PC System	23.95
Panasonic 1080/90/91	6.95
SF 314	5.95
SF 354	5.95
SM 1224	11.95
SM 124	11.95

CABLES, INTERFACES

AB Switch Box Parallel	39.95
AB Switch Box RS-232	39.95
AB-X Crossover Box Par	49.95
IBM/ST Printer Cable 10'	9.95
IBM/ST Printer Cable 6'	6.95
Microstuffer	59.95
RS-232 F/F 6'	9.95
RS-232 M/F 10'	12.95
RS-232 M/F 6'	9.95
RS-232 M/M 6'	9.95
ST Drive Cable 6'	14.95

ACCESSORIES

14" Twist-N-Tilt	14.95
5.25" Disk Notcher	7.95
Amaray Printer Stand	14.95
Mouse House	4.95
Mouse Mat	7.95
Printer Stand 2-Pc.	12.95
Six Outlet Surge Strip	14.95
Disk File 40 - 3.5"	6.99
Disk File 60 - 5.25"	6.99
Library Case-3 (5.25")	.99
Library Case-5 (3.5")	.99
Pocket Pack (3.5")	14.99

DISKETTES

Bulk 3.5" SS/DD	.99
Bulk 3.5" DS/DD	1.29
Bulk 5.25" DS/DD	.39
Maxell 3.5" SS/DD	14.95
Maxell 3.5" DS/DD	22.95
Maxell 5.25" SS/DD	10.95
Maxell 5.25" DS/DD	11.95
Maxell 5.25" DS/HD	23.95
Sony 3.5" SS/DD	14.95
Sony 3.5" DS/DD	22.95
Sony 5.25" SS/DD	8.95
Sony 5.25" DS/DD	9.95
Sony 5.25" DS/HD	22.95

RIBBONS

Epson FX/MX/RX-80	5.99
IBM Proprinter	9.95
Panasonic 1080/90i	9.99
Star/Okidata	2.99
XMM 801/Legend/BMC	7.99
SMM 804	7.99

THOMSON MONITORS

4160 CGA Color	339.00
450A Monochrome - 132 CLM	159.00
4350 EGA Dual Scan	469.00
4375M Ultra Scan	629.00
TTL Monitor	119.00

CONTROLLERS

Atari Economy	4.95
Epyx 500XJ	14.95
Replacement Mouse (ST)	49.95
Tac 2	8.95
TAC-10 IBM & APPLE	27.95
Wico 3-Way	23.95
Wico Boss	12.95

IBM HARDWARE

PC-XT

1200 Baud Internal Modem	119.00
C.G.V. Card	59.95
DS/DD ½ Ht. 360k Drive	89.95
I/O Plus II Card	75.95
Monographics Card	75.95
Printer Card	29.95
RS-232 Card	38.95

IBM SOFTWARE

Archon	10.95
Balance of Power	33.95
F-15 Strike Eagle	23.95
Flight Simulator	33.95
King's Quest I, II or III	33.95
Lotus 1-2-3	333.95
Mark Williams C	333.95
Word Perfect	330.95
Wordstart 2000+	396.95

No surcharge for MasterCard or Visa

All orders are considered verbal contracts and adhere to all laws as set by the Federal Trade Commission. All order cancellations will be charged 5%. Ordering Information: Credit Card - Have number, expiration date, shipping address and DAY phone number ready with order. Personal and company checks: Call order in - you will receive an order number. Send order number and check to the P.O. Box listed. Allow 14 Business Days for check to clear. **Money Orders:** Call order in - you will receive an order number. Send number with money order to P.O. Box listed. **Shipping:** Continental USA - **All Orders add 3%**, minimum $3.00. HI, AK, Canada add 5%, minimum $5.00. All other order areas add 15%, minimum $10.00. No C.O.D.'s over $100.00, add an additional $3.00 shipping charge for C.O.D. All areas not serviced by U.P.S. will be shipped first class insured mail. All shipping charges in excess of the above items will be added. All orders are shipped on a first come first served basis and are usually shipped immediately. Please allow 2-3 weeks delivery time. All goods are new and include factory warranty. **All defectives must have a return authorization number.** Please call (313)595-0788 to obtain an R.A.# or return will not be accepted. Prices and availability subject to change without notice. All correspondence should be sent to P.O. Box 75 - Wayne, MI 48184. *SATISFACTION REFUND - IS LESS SHIPPING & HANDLING - Call for Details.

Not responsible for typographical errors.

MONTGOMERY GRANT

115 Nassau St.Bet. Beekman & Ann Sts.
NY.,NY.10038 (212)732-4500 Open Mon-Fri, 8:30-6:30
/Sat, 9:30-6/ Sun, 9:30-5:30

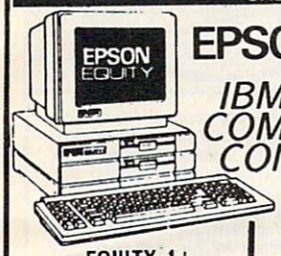
Penn. Station, Amtrack Level
Beneath Madison Sq. Garden,NY.,NY. (212)594-7140
Open Mon-Wed, 8:30-8/Thurs & Fri,8:30-9/Sat & Sun, 10-7

PHONE and MAIL ORDERS CALL TOLL FREE **1(800)345-7058** IN NEW YORK STATE CALL (212)732-4500

ALL COMPUTERS & PERIPHERALS COME COMPLETE WITH AC ADAPTOR, OWNER'S MANUAL AND ALL CONNECTING CABLES.

OPEN SEVEN DAYS A WEEK FOR ORDERS

ALL MERCHANDISE IS BRAND NEW AND FACTORY FRESH. COMPLETE WITH MFR. U.S.A. WARRANTY.



EPSON EQUITY

IBM PC/XT COMPATIBLE COMPUTER PKGS.

EQUITY 1+ PACKAGE
- 640K RAM 360K Floppy
- Drive Serial/Parallel Ports
- Clock Calendar Keyboard
- 12" Monitor

$799.95

EQUITY II PACKAGE
- Equity II Computer 640K
- RAM Power 360K Drive
- 12" High Res. Monitor
- Serial/Parallel Ports

$899.95

EQUITY II PKG. w/1 360K DRIVE & 1 20MB HARD DRIVE...$1199

IBM® COMPUTER PACKAGES

IBM XT HARD DRIVE PACKAGE
- IBM XT Computer - IBM Keyboard - 256K RAM Expandable to 640K - 360K Disk Drive-20MB Hard Drive - Monitor Optional - Package of 10 Diskettes

$1499.95

IBM XT PKG. w/2 FLOPPY DRIVES....$1199

IBM PERSONAL SYSTEM II PACKAGE

MODEL 30
- IBM Computer - Two 720K Drives - 640K RAM - Built-in Graphics - Serial and Parallel Ports

$1379.95

IBM PERSONAL SYSTEM II MODEL 50....$2749

COMPUTER PACKAGES FOR APPLE IIC APPLE IIE APPLE MAC PLUS APPLE IIGS

Apple IIC, IIE, IIGS, or MAC+ Keyboard 5.25" Disk Drive - 12"|Monitor - Package of 10 Diskettes - AC Adaptors - All Hook-up Cables

APPLE IIC PACKAGE	APPLE IIE PACKAGE	APPLE IIGS PACKAGE	MACPLUS PACKAGE
$699	$899	$1195	$1649

IMAGEWRITER II|PRINTER.....$489

PRINTERS

EPSON FX-86	$309.95	HP LASERJET SERIES II	$1695.95	STAR ND-15	$399.95	NEC MULTISPEED	$1325
EPSON FX-286E	$439.95	STAR SG-10C	$179.95	STAR NR-15	$479.95	TOSHIBA 3100	$2650
EPSON LQ-800	$449.95	STAR NX-10	$179.95	STAR NB24-10	$419.95	ZENITH 181	$1639
EPSON LQ-1000	$599.95	STAR NX-10C	$189.95	STAR NB24-15	$629.95	ZENITH 183	$2349
EPSON EX-1000	$489.95	STAR NL-10	$239.95	STAR NB-15	$849.95		
PANASONIC 1080 I	$159.95	STAR NX-15	$309.95	OKIDATA 120	$199.95		
PANASONIC 1091 I	$189.95			OKIMATE 20	$139.95		
PANASONIC 1092	$289.95						

20MB SEAGATE Hard Drive 30MB IN STOCK **329.95**

LAPTOPS

BLUE CHIP

IBM PC/XT COMPATIBLE PCKG.

- Blue Chip Keyboard - 512K Expandable to 640K - 6 Expansion Slots - Optional 20MB Hard Drive - All Hook-up Cables & Adaptors - Package of 10 Diskettes - 12" Monitor

SPECIAL

$529.95

commodore PC10-1 PACKAGE

IBM PC/XT Compatible

- PC10-1 Computer - 512K Expandable to 640K - 360K Disk Drive - Enhanced Keyboard-Serial & Parallel Ports - 12" Monitor - All Hook-up Cables & Adaptors - 10 Diskettes

$549.95

PC10-2 640K Computer w/2 360K Floppy Drives **659.95**

AMIGA 1000 512K SYSTEM

512K Computer with Built-in Disk Drive and 1080 Color Monitor

$949.95

NEW AMIGA 500 NOW IN STOCK
ALL AMIGA PERIPHERALS & MEMORY EXPANSION IN STOCK!
AMIGA 2000.....CALL

IBM PC/XT COMPATIBLE PACKAGE

$699.95

- Keyboard - 640K RAM - Turbospeed 4.7/8 MHz. - Two 360K Drives - MS DOS - Mono/Graphics Card - High Resolution Mono Monitor - FREE Word Processing Software

commodore

COMMODORE 64-C PERSONAL COMPUTER	144.95
COMMODORE 128 PERSONAL COMPUTER	219.95
COMMODORE MPS-1200 Computer Printer	229.95
COMMODORE MPS-803C Computer Printer	129.95
COMMODORE MPS-1000 Computer Printer	199.95
COMMODORE 128 D w/Built-in Disk Drive	449.95

COMPUTER PACKAGES

Get it all together & save!

C64C COMPLETE PACKAGE
- Commodore 64 Computer
- Commodore 1541 Disk Drive
- Commodore Color Printer
- 12" Monitor

$379.95

Color Monitor ADD $130

C64C COLOR PACKAGE
- Commodore 64C Computer
- Commodore 1541C Disk Drive
- Commodore Color Printer
- Color Monitor

With Geos Program!

$529.95

C128 COMPLETE PACKAGE
- Commodore 128 Computer
- Commodore 1541 Disk Drive
- 12" Monitor
- Commodore Color Printer

$479.95

To substitute 1571 for 1541 add **$65**

C128 DELUXE PACKAGE
- Commodore 128 Computer
- Commodore 1571 Disk Drive
- Commodore 1902 Color Monitor
- Commodore 80 Column Printer

$759.95

commodore

COMMODORE 1541C NEW Disk Drive	149.95
COMMODORE 1571 Computer Disk Drive	214.95
COMMODORE 1902 80 Column RGB Color Monitor	229.95
COMMODORE 1802 Color Computer Monitor	169.95
COMMODORE "MAGIC VOICE" Module	49.95
COMMODORE 1700 - 128K Expansion Module	99.95
COMMODORE 1750 - 512K Expansion Module	169.95
COMMODORE 1670 Automodem	119.95
COMMODORE 1764 Expansion Module	Call

MasterCard DISCOVER VISA

WE INVITE CORPORATE & EDUCATIONAL CUSTOMERS

Certified check, Bank check, Mastercard, Visa, Am-Ex, Diner's Club, Carte-Blanche, Discover Card and C.O.D.s accepted. No additional surcharge for credit card orders. Non-certified checks must wait 4-6 weeks clearance. Money orders are non-certified checks. N.Y. residents add applicable sales tax. Prices and availability subject to change without notice-so call before placing your order. Not responsible for typographic errors. Return of defective merchandise must have prior return authorization number, or returns will not be accepted. IBM PC/XT are registered trademarks of International Business Machine Corp. All orders can be shipped Air Express-call for details.

1541/1571 Drive Alignment

1541/1571 Drive Alignment reports the alignment condition of the disk drive as you perform adjustments. On screen help is available while the program is running. Includes features for speed adjustment and stop adjustment. Complete instruction manual on aligning both 1541 and 1571 drives. Even includes instructions on how to load alignment program when nothing else will load! Works on the C64, SX64, C128 in 64 or 128 mode, 1541, 1571 in either 1541 or 1571 mode! Autoboots to all modes. Second disk drive fully supported. Program disk, calibration disk and instruction manual only $34.95!

Super Disk Utilities

Super Disk Utilities is the ultimate utilities disk for the 1571 disk drive and C128 computer. An 80 column monitor is required. SDU will copy whole disks (with 1 or 2 drives), perform many CP/M and MS-DOS utility functions, edit any track or sector, trace files, copy files, rename a disk, rename a file, change disk format (without affecting data), scratch or unscratch files, lock or unlock files, erase a track or a whole disk, create auto-boot, assemble/disassemble drive RAM or ROM, analyze disk format, format in MFM, CPM+, 1541, or 1571 format, reads CPM+ directories, contains a 1571 drive monitor, fully supports a second 1571 drive, performs direct DOS commands and much, much more! Options window displays all choices available at any given time. Many utilities also work on the 1541 drive. No need for numerous utility disks! SDU does it all! Only $39.95!

Free shipping & handling! No surcharge for credit card orders! Immediate shipping! Illinois residents add 8% sales tax. Order from:

Free Spirit Software, Inc.
538 S. Edgewood
La Grange, IL 60525
(312) 352-7323

MasterCard VISA

THE AMAZING VOICE MASTER®

SPEECH DIGITIZER, VOICE RECOGNITION AND MORE . . .

There is nothing else like it. Voice Master gives both speech output and voice recognition with this single hardware product! Your voice controls programs, or home appliances, robots, and more with spoken commands. Verbal response back gives status, verifies, or requests your reply! Speech output and recognition patterns are recorded into memory with your voice. Programming is simple with new commands added to BASIC. Demo programs get you up and running quickly. A music bonus program lets you write and compose musical scores merely by humming a tune. Unlimited applications for fun, education, or commercial use. Design your own programs for profit. Speech and recognition quality the finest available. Only Covox provides this high-tech marvel for unlimited applications at a very affordable price.

The Covox Voice Master comes complete with all hardware and software for only $89.95. 30 day satisfaction guarantee. One year warranty. (Add $4 shipping and handling for USA, $6 Canada, $10 overseas.) Available for Commodore 64/128, Apple II, II+, IIc, IIe, Atari 800, 800XL, 130 XE (IBM PC and compatibles version available soon). Specify when ordering. Visa, MasterCard phone orders accepted.

Call or write today for FREE Covox product catalog and newsletter

COVOX INC.
675-D Conger Street • Eugene, Oregon 97402 • U.S.A.
Area Code (503) 342-1271 • Telex 706017 (Av Alarm UD)

COMPUTER PRICE BLOWOUT!!

FOR MAIL ORDERS ONLY! CALL TOLL FREE 1-800-234-5888

COMPUTER WORLD

ASTORIA
40-17 Broadway-East Of Steinway St. Corner
FOR INFO: 718-726-8118
MON.-FRI. 10:30-7/CLOSED SAT./SUN. 11-6

GREAT NECK
5820 Middle Neck Road
FOR INFO: 516-829-2188
MON.-FRI. 11-7/CLOSED SAT./SUN. 11-5

EPSON
EQUITY I PACKAGE
IBM PC/XT COMPATIBLE
$549

•Equity 1 Computer •256K Ram Power •Dual 360K Disk Drives •12" High Resolution Monitor •Serial and Parallel Ports

BLUE CHIP
IBM PC/XT COMPATIBLE PACKAGE
$535

•Blue Chip Keyboard •512K Expandable To 640K •6 Expansion Slots-Optional 20MB Hard Drive •All Hook-Up Cables & Adaptors •Package Of 10 Diskettes •12" Monitor

IBM® TURBO
IBM PC/XT COMPATIBLE
$749

•640K •2 Drive / 360K •8088 / 2 CPU •4.77 8 MHZ •12" High Resolution Monitor

COMMODORE
PC10 (1 PACKAGE)
IBM PC/XT COMPATIBLE
$565

•PC10 - 1 Computer •512K Expandable To 640K •360K Disk Drive •Enhanced Keyboard •Serial & Parallel Ports •All Hook-Up Cables & Adaptors •10 Diskettes
PC10 - 2 640K COMPUTER W/ 2 360K FLOPPY DRIVES CALL!

AUTHORIZED DEALER FOR MOST MAJOR BRANDS

II GS PACKAGE
•Apple II GS Computer •5.25" Disk Drive •12" Monitor •Package Of 10 Diskettes •AC Adaptors •All Hook-Up Cables **$1049**

COMMODORE

•Commodore 64-C Personal Computer	$135
•Commodore 128 Personal Computer	$218
•Commodore MPS-1200 Computer Printer	$209
•Commodore MPS-803C Computer Printer	$109
•Commodore MPS-1000 Computer Printer	$179
•Panasonic 1080i	$165

ALL COMPUTERS & PERIPHERALS COME COMPLETE W/AC ADAPTOR, OWNER'S MANUAL AND ALL CONNECTING CABLES.

MODEMS IN STOCK!
•IBM
•COMMODORE

Sale

AMIGA
NEW!! 500 AMIGA 512K SYSTEM
WE'LL BEAT ANY PRICE!!!

•512K Computer W/Built-In Disk Drive & 1080 Color Monitor •Including Software

AMIGA 2000 . . . . . CALL
ATARI 520 COLOR . . CALL

EXPANSION MODELS AVAILABLE! Call For Low Prices!!

SOFTWARE

•F-15	$12.95
•Pitstop	$12.95
•Pitfall	$8.95
•Championship Baseball	$19.95

WE CARRY A FULL LINE OF COMMODORE SOFTWARE

COMMODORE 128D
NEW! WE'LL BEAT ANY PRICE!

C 64 COMPLETE PACKAGE
•Commodore 64 Computer •Commodore 1541 Disk Drive •Commodore Color Printer 12" Monitor
$349.00

C 128 COMPLETE PACKAGE
•Commodore 128 Computer •Commodore 1541 Disc Drive •12" Monitor •Commodore Color Printer
COLOR MONITOR ADD $130. TO SUB. 1571 FOR 1541 ADD $55.
$499.00

C 64C COLOR PACKAGE
•Commodore 64C Computer •Commodore 1541 Disk Drive •Commodore Color Printer •Color Monitor With GEOS Program
$449.00

C 128 DELUXE PACKAGE
•Commodore 128 Computer •Commodore 1571 Disk Drive •Commodore 1902 Color Monitor •Commodore 80 Column Printer
$699.00

FULL LINE OF IBM COMPATABLES!! Call For Lowest Prices!!

COMMODORE

•Commodore 1541C New!! Disk Drive	$159
•Commodore 1571 Computer Disk Drive	$205
•Commodore 1902 80 Column RGB Color Monitor	$225

•Expansion Modules Available•

COMMODORE C-1802 COLOR MONITOR $175

ALL MERCHANDISE IS BRAND NEW AND FACTORY FRESH, COMPLETE WITH MFG. U.S.A. WARRANTY.

•SERVICE CENTER FOR COMMODORE•
•ATARI COMPUTERS AT LOW PRICES•

Mastercard, Visa, American Express, Diner's Club, Carte Blanche, Discover Card, Certified And Bank Checks Welcome. C.O.D.'s Accepted Upon Approval. No Additional Surcharge For Credit Card Orders. Non-Certified Checks & Money Orders Must Wait For Clearance. N.Y. Residents Add Applicable Sales Tax. We Are Not Responsible For Typographical Errors. Prices And Availability Are Subject To Change Without Notice. To Return Defective Merchandise Call For Authorization Number. IBM PC/XT Are Registered Trademarks Of International Business Machine Corp. All Orders Can Be Shipped Air Express.

COMPUTE!'s Guide To Typing In Programs

Computers are precise—type the program *exactly* as listed, including necessary punctuation and symbols, except for special characters noted below. We have provided a special listing convention as well as a program to check your typing—"The Automatic Proofreader."

Programs for the IBM, TI-99/4A, and Atari ST models should be typed exactly as listed; no special characters are used. Programs for Commodore, Apple, and Atari 400/800/XL/XE computers may contain some hard-to-read special characters, so we have a listing system that indicates these control characters. You will find these Commodore and Atari characters in curly braces; *do not type the braces.* For example, {CLEAR} or {CLR} instructs you to insert the symbol which clears the screen on the Atari or Commodore machines. A complete list of these symbols is shown in the tables below. For Commodore, Apple, and Atari, a single symbol by itself within curly braces is usually a control key or graphics key. If you see {A}, hold down the CONTROL key and press A. This will produce a reverse video character on the Commodore (in quote mode), a graphics character on the Atari, and an invisible control character on the Apple.

Graphics characters entered with the Commodore logo key are enclosed in a special bracket: [<A>]. In this case, you would hold down the Commodore logo key as you type A. Our Commodore listings are in uppercase, so shifted symbols are underlined. A graphics heart symbol (SHIFT-S) would be listed as S. One exception is {SHIFT-SPACE}. When you see this, hold down SHIFT and press the space bar. If a number precedes a symbol, such as {5 RIGHT}, {6 S}, or [<8 Q>], you would enter five cursor rights, six shifted S's, or eight Commodore-Q's. On the Atari, inverse characters (white on black) should be entered with the inverse video

Atari 400/800/XL/XE

When you see	Type	See	
{CLEAR}	ESC SHIFT <		Clear Screen
{UP}	ESC CTRL -		Cursor Up
{DOWN}	ESC CTRL =		Cursor Down
{LEFT}	ESC CTRL +		Cursor Left
{RIGHT}	ESC CTRL *		Cursor Right
{BACK S}	ESC DELETE		Backspace
{DELETE}	ESC CTRL DELETE		Delete character
{INSERT}	ESC CTRL INSERT		Insert character
{DEL LINE}	ESC SHIFT DELETE		Delete line
{INS LINE}	ESC SHIFT INSERT		Insert line
{TAB}	ESC TAB		TAB key
{CLR TAB}	ESC CTRL TAB		Clear tab
{SET TAB}	ESC SHIFT TAB		Set tab stop
{BELL}	ESC CTRL 2		Ring buzzer
{ESC}	ESC ESC		ESCape key

Commodore PET/CBM/VIC/64/128/16/+4

When You Read:	Press:	See:
{CLR}	SHIFT + CLR/HOME	
{HOME}	CLR/HOME	
{UP}	SHIFT + ↑ CRSR ↓	
{DOWN}	↑ CRSR ↓	
{LEFT}	SHIFT + ← CRSR →	
{RIGHT}	← CRSR →	
{RVS}	CTRL + 9	
{OFF}	CTRL + 0	
{BLK}	CTRL + 1	
{WHT}	CTRL + 2	
{RED}	CTRL + 3	
{CYN}	CTRL + 4	
{PUR}	CTRL + 5	
{GRN}	CTRL + 6	
{BLU}	CTRL + 7	
{YEL}	CTRL + 8	

When You Read:	Press:	See:
[<1>]	COMMODORE + 1	
[<2>]	COMMODORE + 2	
[<3>]	COMMODORE + 3	
[<4>]	COMMODORE + 4	
[<5>]	COMMODORE + 5	
[<6>]	COMMODORE + 6	
[<7>]	COMMODORE + 7	
[<8>]	COMMODORE + 8	
{ F1 }	f1	
{ F2 }	SHIFT + f1	
{ F3 }	f3	
{ F4 }	SHIFT + f3	
{ F5 }	f5	
{ F6 }	SHIFT + f5	
{ F7 }	f7	
{ F8 }	SHIFT + f7	
←	←	

key (Atari logo key on 400/800 models).

Whenever more than two spaces appear in a row, they are listed in a special format. For example, {6 SPACES} means press the space bar six times. Our Commodore listings never leave a single space at the end of a line, instead moving it to the next printed line as {SPACE}.

Amiga program listings contain only one special character, the left arrow (←) symbol. This character marks the end of each program line. Wherever you see a left arrow, press RETURN or move the cursor off the line to enter that line into memory. Don't try to type in the left arrow symbol; it's there only as a marker to indicate where each program line ends.

The Automatic Proofreader

Type in the appropriate program listed below, then save it for future use. The Commodore Proofreader works on the Commodore 128, 64, Plus/4, 16, and VIC-20. Don't omit any lines, even if they contain unfamiliar commands or you think they don't apply to your computer. When you run the program, it installs a machine language program in memory and erases its BASIC portion automatically (so be sure to save several copies before running the program for the first time). If you're using a Commodore 128, Plus/4 or 16, do *not* use any GRAPHIC commands while the Proofreader is active. You should disable the Commodore Proofreader before running any other program. To do this, either turn the computer off and on or enter SYS 64738 (for the 64), SYS 65341 (128), SYS 64802 (VIC-20), or SYS 65526 (Plus/4 or 16). To reenable the Proofreader, reload the program and run it as usual. Unlike the original VIC/64 Proofreader, this version works the same with disk or tape.

On the Atari, run the Proofreader to activate it (the Proofreader remains active in memory as a machine language program); you must then enter NEW to erase the BASIC loader. Pressing SYSTEM RESET deactivates the Atari Proofreader; enter PRINT USR(1536) to reenable it.

The Apple Proofreader erases the BASIC portion of itself after you run it, leaving only the machine language portion in memory. It works with either DOS 3.3 or ProDOS. Disable the Apple Proofreader by pressing CTRL-RESET before running another BASIC program.

The IBM Proofreader is a BASIC program that simulates the IBM BASIC line editor, letting you enter, edit, list, save, and load programs that you type. Type RUN to activate. Be sure to leave Caps Lock on, except when typing lowercase characters.

Once the Proofreader is active, try typing in a line. As soon as you press RETURN, either a hexadecimal number (on the Apple) or a pair of letters (on the Commodore, Atari, or IBM) appears. The number or pair of letters is called a *checksum*.

Compare the value displayed on the screen by the Proofreader with the checksum printed in the program listing in the magazine. The checksum is given to the left of each line number. Just type in the program a line at a time (without the printed checksum), press RETURN or Enter, and compare the checksums. If they match, go on to the next line. If not, check your typing; you've made a mistake. Because of the checksum method used, do not type abbreviations, such as ? for PRINT. On the Atari and Apple Proofreaders, spaces are not counted as part of the checksum, so be sure you type the right number of spaces between quote marks. The Atari Proofreader does not check to see that you've typed the characters in the right order, so if characters are transposed, the checksum still matches the listing. The Commodore Proofreader catches transposition errors and ignores spaces unless they're enclosed in quotation marks. The IBM Proofreader detects errors in spacing and transposition.

IBM Proofreader Commands

Since the IBM Proofreader replaces the computer's normal BASIC line editor, it has to include many of the direct-mode IBM BASIC commands. The syntax is identical to IBM BASIC. Commands simulated are LIST, LLIST, NEW, FILES, SAVE, and LOAD. When listing your program, press any key (except Ctrl-Break) to stop the listing. If you enter NEW, the Proofreader prompts you to press Y to be especially sure you mean yes.

Two new commands are BASIC and CHECK. BASIC exits the Proofreader back to IBM BASIC, leaving the Proofreader in memory. CHECK works just like LIST, but shows the checksums along with the listing. After you have typed in a program, save it to disk. Then exit the Proofreader with the BASIC command, and load the program as usual (this replaces the Proofreader in memory). You can now run the program, but you may want to resave it to disk. This will shorten it on disk and make it load faster, but it can no longer be edited with the Proofreader. If you want to convert an existing BASIC program to Proofreader format, save it to disk with SAVE "*filename*",A.

Program 1: Atari Proofreader

By Charles Brannon

```
100 GRAPHICS 0
110 FOR I=1536 TO 1700:REA
    D A:POKE I,A:CK=CK+A:N
    EXT I
120 IF CK<>19072 THEN ? "E
    rror in DATA Statement
    s.  Check Typing.":END

130 A=USR(1536)
140 ? :? "Automatic Proofr
    eader Now Activated."
150 END
160 DATA 104,160,0,185,26,
    3,201,69,240,7
170 DATA 200,200,192,34,20
    8,243,96,200,169,74
180 DATA 153,26,3,200,169,
    6,153,26,3,162
190 DATA 0,189,0,228,157,7
    4,6,232,224,16
200 DATA 208,245,169,93,14
    1,78,6,169,6,141
210 DATA 79,6,24,173,4,228
    ,105,1,141,95
220 DATA 6,173,5,228,105,0
    ,141,96,6,169
230 DATA 0,133,203,96,247,
    238,125,241,93,6
240 DATA 244,241,115,241,1
    24,241,76,205,238
250 DATA 0,0,0,0,0,32,62,2
    46,8,201
260 DATA 155,240,13,201,32
    ,240,7,72,24,101
270 DATA 203,133,203,104,4
    0,96,72,152,72,138
280 DATA 72,160,0,169,128,
    145,88,200,192,40
290 DATA 208,249,165,203,7
    4,74,74,74,24,105
300 DATA 161,160,3,145,88,
    165,203,41,15,24
310 DATA 105,161,200,145,8
    8,169,0,133,203,104
320 DATA 170,104,168,104,4
    0,96
```

Program 2: IBM Proofreader

By Charles Brannon

```
10 'Automatic Proofreader Vers
   ion 3.0 (Lines 205,206 adde
   d/190 deleted/470,490 chang
   ed from V2.0)
100 DIM L$(500),LNUM(500):COLO
    R 0,7,7:KEY OFF:CLS:MAX=0:
    LNUM(0)=65536!
110 ON ERROR GOTO 120:KEY 15,C
    HR$(4)+CHR$(70):ON KEY(15)
     GOSUB 640:KEY (15) ON:GOT
    O 130
120 RESUME 130
130 DEF SEG=&H40:W=PEEK(&H4A)
140 ON ERROR GOTO 650:PRINT:PR
    INT"Proofreader Ready."
150 LINE INPUT L$:Y=CSRLIN-INT
    (LEN(L$)/W)-1:LOCATE Y,1
160 DEF SEG=0:POKE 1050,30:POK
    E 1052,34:POKE 1054,0:POKE
     1055,79:POKE 1056,13:POKE
     1057,28:LINE INPUT L$:DEF
     SEG:IF L$="" THEN 150
170 IF LEFT$(L$,1)=" " THEN L$
    =MID$(L$,2):GOTO 170
```

```
180 IF VAL(LEFT$(L$,2))=0 AND
    MID$(L$,3,1)=" " THEN L$=M
    ID$(L$,4)
200 IF ASC(L$)>57 THEN 260 'no
    line number, therefore co
    mmand
205 BL=INSTR(L$," "):IF BL=0 T
    HEN BL$=L$:GOTO 206 ELSE B
    L$=LEFT$(L$,BL-1)
206 LNUM=VAL(BL$):TEXT$=MID$(L
    $,LEN(STR$(LNUM))+1)
210 IF TEXT$="" THEN GOSUB 540
    :IF LNUM=LNUM(P) THEN GOSU
    B 560:GOTO 150 ELSE 150
220 CKSUM=0:FOR I=1 TO LEN(L$)
    :CKSUM=(CKSUM+ASC(MID$(L$,
    I))*I) AND 255:NEXT:LOCATE
     Y,1:PRINT CHR$(65+CKSUM/1
    6)+CHR$(65+(CKSUM AND 15))
    +" "+L$
230 GOSUB 540:IF LNUM(P)=LNUM
    THEN L$(P)=TEXT$:GOTO 150
    'replace line
240 GOSUB 580:GOTO 150 'insert
     the line
260 TEXT$="":FOR I=1 TO LEN(L$
    ):A=ASC(MID$(L$,I)):TEXT$=
    TEXT$+CHR$(A+32*(A>96 AND
    A<123)):NEXT
270 DELIMITER=INSTR(TEXT$," ")
    :COMMAND$=TEXT$:ARG$="":IF
     DELIMITER THEN COMMAND$=L
    EFT$(TEXT$,DELIMITER-1):AR
    G$=MID$(TEXT$,DELIMITER+1)
     ELSE DELIMITER=INSTR(TEXT
    $,CHR$(34)):IF DELIMITER T
    HEN COMMAND$=LEFT$(TEXT$,D
    ELIMITER-1):ARG$=MID$(TEXT
    $,DELIMITER)
280 IF COMMAND$<>"LIST" THEN 4
    10
290 OPEN "scrn:" FOR OUTPUT AS
     #1
300 IF ARG$="" THEN FIRST=0:P=
    MAX-1:GOTO 340
310 DELIMITER=INSTR(ARG$,"-"):
    IF DELIMITER=0 THEN LNUM=V
    AL(ARG$):GOSUB 540:FIRST=P
    :GOTO 340
320 FIRST=VAL(LEFT$(ARG$,DELIM
    ITER)):LAST=VAL(MID$(ARG$,
    DELIMITER+1))
330 LNUM=FIRST:GOSUB 540:FIRST
    =P:LNUM=LAST:GOSUB 540:IF
    P=0 THEN P=MAX-1
340 FOR X=FIRST TO P:N$=MID$(S
    TR$(LNUM(X)),2)+" "
350 IF CKFLAG=0 THEN A$="":GOT
    O 370
360 CKSUM=0:A$=N$+L$(X):FOR I=
    1 TO LEN(A$):CKSUM=(CKSUM+
    ASC(MID$(A$,I))*I) AND 255
    :NEXT:A$=CHR$(65+CKSUM/16)
    +CHR$(65+(CKSUM AND 15))+"
     "
370 PRINT #1,A$+N$+L$(X)
380 IF INKEY$<>"" THEN X=P
390 NEXT :CLOSE #1:CKFLAG=0
400 GOTO 130
410 IF COMMAND$="LLIST" THEN O
    PEN "lpt1:" FOR OUTPUT AS
    #1:GOTO 300
420 IF COMMAND$="CHECK" THEN C
    KFLAG=1:GOTO 290
430 IF COMMAND$<>"SAVE" THEN 4
    50
440 GOSUB 600:OPEN ARG$ FOR OU
    TPUT AS #1:ARG$="":GOTO 30
    0
450 IF COMMAND$<>"LOAD" THEN 4
    90
460 GOSUB 600:OPEN ARG$ FOR IN
    PUT AS #1:MAX=0:P=0
470 WHILE NOT EOF(1):LINE INPU
    T #1,L$:BL=INSTR(L$," "):B
    L$=LEFT$(L$,BL-1):LNUM(P)=
    VAL(BL$):L$(P)=MID$(L$,LEN
    (STR$(VAL(BL$)))+1):P=P+1:
    WEND
480 MAX=P:CLOSE #1:GOTO 130
490 IF COMMAND$="NEW" THEN INP
    UT "Erase program - Are yo
    u sure";L$:IF LEFT$(L$,1)=
    "y" OR LEFT$(L$,1)="Y" THE
    N MAX=0:LNUM(0)=65536!:GOT
    O 130:ELSE 130
500 IF COMMAND$="BASIC" THEN C
    OLOR 7,0,0:ON ERROR GOTO 0
    :CLS:END
510 IF COMMAND$<>"FILES" THEN
    520
515 IF ARG$="" THEN ARG$="A:"
    ELSE SEL=1:GOSUB 600
517 FILES ARG$:GOTO 130
520 PRINT"Syntax error":GOTO 1
    30
540 P=0:WHILE LNUM>LNUM(P) AND
     P<MAX:P=P+1:WEND:RETURN
560 MAX=MAX-1:FOR X=P TO MAX:L
    NUM(X)=LNUM(X+1):L$(X)=L$(
    X+1):NEXT:RETURN
580 MAX=MAX+1:FOR X=MAX TO P+1
     STEP -1:LNUM(X)=LNUM(X-1)
    :L$(X)=L$(X-1):NEXT:L$(P)=
    TEXT$:LNUM(P)=LNUM:RETURN
600 IF LEFT$(ARG$,1)<>CHR$(34)
     THEN 520 ELSE ARG$=MID$(A
    RG$,2)
610 IF RIGHT$(ARG$,1)=CHR$(34)
     THEN ARG$=LEFT$(ARG$,LEN(
    ARG$)-1)
620 IF SEL=0 AND INSTR(ARG$,".
    ")=0 THEN ARG$=ARG$+".BAS"
630 SEL=0:RETURN
640 CLOSE #1:CKFLAG=0:PRINT"St
    opped.":RETURN 150
650 PRINT "Error #";ERR:RESUME
     150
```

Program 3: Commodore Proofreader

By Philip Nelson, Assistant Editor

```
10 VEC=PEEK(772)+256*PEEK(773)
   :LO=43:HI=44
20 PRINT "AUTOMATIC PROOFREADE
   R FOR ";:IF VEC=42364 THEN
   {SPACE}PRINT "C-64"
30 IF VEC=50556 THEN PRINT "VI
   C-20"
40 IF VEC=35158 THEN GRAPHIC C
   LR:PRINT "PLUS/4 & 16"
50 IF VEC=17165 THEN LO=45:HI=
   46:GRAPHIC CLR:PRINT"128"
60 SA=(PEEK(LO)+256*PEEK(HI))+
   6:ADR=SA
70 FOR J=0 TO 166:READ BYT:POK
   E ADR,BYT:ADR=ADR+1:CHK=CHK
   +BYT:NEXT
80 IF CHK<>20570 THEN PRINT "*
    ERROR* CHECK TYPING IN DATA
    STATEMENTS":END
90 FOR J=1 TO 5:READ RF,LF,HF:
   RS=SA+RF:HB=INT(RS/256):LB=
   RS-(256*HB)
100 CHK=CHK+RF+LF+HF:POKE SA+L
    F,LB:POKE SA+HF,HB:NEXT
110 IF CHK<>22054 THEN PRINT "
    *ERROR* RELOAD PROGRAM AND
    {SPACE}CHECK FINAL LINE":EN
    D
120 POKE SA+149,PEEK(772):POKE
     SA+150,PEEK(773)
130 IF VEC=17165 THEN POKE SA+
    14,22:POKE SA+18,23:POKESA+
    29,224:POKESA+139,224
140 PRINT CHR$(147);CHR$(17);"
    PROOFREADER ACTIVE":SYS SA
150 POKE HI,PEEK(HI)+1:POKE (P
    EEK(LO)+256*PEEK(HI))-1,0:N
    EW
160 DATA 120,169,73,141,4,3,16
    9,3,141,5,3
170 DATA 88,96,165,20,133,167,
    165,21,133,168,169
180 DATA 0,141,0,255,162,31,18
    1,199,157,227,3
190 DATA 202,16,248,169,19,32,
    210,255,169,18,32
200 DATA 210,255,160,0,132,180
    ,132,176,136,230,180
210 DATA 200,185,0,2,240,46,20
    1,34,208,8,72
220 DATA 165,176,73,255,133,17
    6,104,72,201,32,208
230 DATA 7,165,176,208,3,104,2
    08,226,104,166,180
240 DATA 24,165,167,121,0,2,13
    3,167,165,168,105
250 DATA 0,133,168,202,208,239
    ,240,202,165,167,69
260 DATA 168,72,41,15,168,185,
    211,3,32,210,255
270 DATA 104,74,74,74,74,168,1
    85,211,3,32,210
280 DATA 255,162,31,189,227,3,
    149,199,202,16,248
290 DATA 169,146,32,210,255,76
    ,86,137,65,66,67
300 DATA 68,69,70,71,72,74,75,
    77,80,81,82,83,88
310 DATA 13,2,7,167,31,32,151,
    116,117,151,128,129,167,136
    ,137
```

Program 4: Apple Proofreader

By Tim Victor, Editorial Programmer

```
10 C = 0: FOR I = 768 TO 768 +
   68: READ A:C = C + A: POKE I
   ,A: NEXT
20 IF C < > 7258 THEN PRINT "ER
   ROR IN PROOFREADER DATA STAT
   EMENTS": END
30 IF PEEK (190 * 256) < > 76 T
   HEN POKE 56,0: POKE 57,3: CA
   LL 1002: GOTO 50
40 PRINT CHR$ (4);"IN#A$300"
50 POKE 34,0: HOME : POKE 34,1:
    VTAB 2: PRINT "PROOFREADER
   INSTALLED"
60 NEW
100 DATA 216,32,27,253,201,141
110 DATA 208,60,138,72,169,0
120 DATA 72,189,255,1,201,160
130 DATA 240,8,104,10,125,255
140 DATA 1,105,0,72,202,208
150 DATA 238,104,170,41,15,9
160 DATA 48,201,58,144,2,233
170 DATA 57,141,1,4,138,74
180 DATA 74,74,74,41,15,9
190 DATA 48,201,58,144,2,233
200 DATA 57,141,0,4,104,170
210 DATA 169,141,96
```

MLX Machine Language Entry Program For Commodore 64

Ottis Cowper, Technical Editor

"MLX" is a labor-saving utility that allows almost fail-safe entry of Commodore 64 machine language programs.

Type in and save some copies of MLX—you'll want to use it to enter future machine langauge (ML) programs from COMPUTE!. When you're ready to enter an ML program, load and run MLX. It asks you for a starting address and an ending address. These addresses appear in the article accompanying the MLX-format program listing you're typing.

If you're unfamiliar with machine language, the addresses (and all other values you enter in MLX) may appear strange. Instead of the usual decimal numbers you're accustomed to, these numbers are in *hexadecimal*—a base 16 numbering system commonly used by ML programmers. Hexadecimal—hex for short—includes the numerals 0–9 and the letters A–F. But don't worry—even if you know nothing about ML or hex, you should have no trouble using MLX.

After you enter the starting and ending addresses, you'll be offered the option of clearing the workspace. Choose this option if you're starting to enter a new listing. If you're continuing a listing that's partially typed from a previous session, don't choose this option.

A functions menu will appear. The first option in the menu is ENTER DATA. If you're just starting to type in a program, pick this. Press the E key, and type the first number in the first line of the program listing. If you've already typed in part of a program, type the line number where you left off typing at the end of the previous session (be sure to load the partially completed program before you resume entry). In any case, make sure the address you enter corresponds to the address of a line in the listing you are entering. Otherwise, you'll be unable to enter the data correctly. If you pressed E by mistake, you can return to the command menu by pressing RETURN alone when asked for the address. (You can get back to the menu from most options by pressing RETURN with no other input.)

Entering A Listing

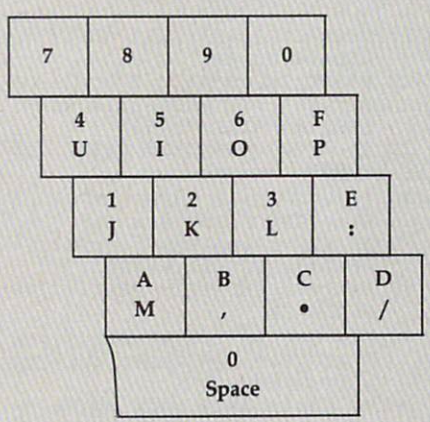
Once you're in Enter mode, MLX prints the address for each program line for you. You then type in all nine numbers on that line, beginning with the first two-digit number after the colon (:). Each line represents eight data bytes and a checksum. Although an MLX-format listing appears similar to the "hex dump" listings from a machine language monitor program, the extra checksum number on the end allows MLX to check your typing.

When you enter a line, MLX recalculates the checksum from the eight bytes and the address and compares this value to the number from the ninth column. If the values match, you'll hear a bell tone, the data will be added to the workspace area, and the prompt for the next line of data will appear. But if MLX detects a typing error, you'll hear a low buzz and see an error message. The line will then be redisplayed for editing.

Invalid Characters Banned

Only a few keys are active while you're entering data, so you may have to unlearn some habits. You *do not* type spaces between the columns; MLX automatically inserts these for you. You *do not* press RETURN after typing the last number in a line; MLX automatically enters and checks the line after you type the last digit.

Only the numerals 0–9 and the letters A–F can be typed in. If you press any other key (with some exceptions noted below), you'll hear a warning buzz. To simplify typing, a numeric keypad is now incorporated in the listing. The keypad is active only while entering data. Addresses must be entered with the normal letter and number keys. The figure below shows the keypad configuration:



MLX checks for transposed characters. If you're supposed to type in A0 and instead enter 0A, MLX will catch your mistake. There is one error that can slip past MLX: Because of the checksum formula used, MLX won't notice if you accidentally type FF in place of 00, and vice versa. And there's a very slim chance that you could garble a line and still end up with a combination of characters that adds up to the proper checksum. However, these mistakes should not occur if you take reasonable care while entering data.

Editing Features

To correct typing mistakes before finishing a line, use the INST/DEL key to delete the character to the left of the cursor. (The cursor-left key also deletes.) If you mess up a line really badly, press CLR/HOME to start the line over. The RETURN key is also active, but only before any data is typed on a line. Pressing RETURN at this point returns you to the command menu. After you type a character of data, MLX disables RETURN until the cursor returns to the start of a line. Remember, you can press CLR/HOME to quickly get to a line number prompt.

More editing features are available when correcting lines in which MLX has detected an error. To make corrections in a line that MLX has redisplayed for editing, compare the line on the screen with the one printed in the listing, then move the cursor to the mistake and type the correct key. The cursor left and right keys provide the normal cursor controls. (The INST/DEL key now works as an alternative cursor-left key.) You cannot move left beyond the first character in the line. If you try to move beyond the rightmost character, you'll reenter the line. During editing, RETURN is active; pressing it tells MLX to recheck the line. You can press the CLR/HOME key to clear the entire line if you want to start from scratch, or if you want to get to a line number prompt to use RETURN to get back to the menu.

Display Data

The second menu choice, DISPLAY DATA, examines memory and shows the contents in the same format as the program listing (including the checksum). When you press D, MLX asks you for a starting address. Be sure that the starting address you give corresponds to a line number in the listing. Otherwise, the checksum display will be meaningless. MLX displays program lines until it reaches the end of the program, at which point the menu is redisplayed. You can pause the display by pressing the space bar. (MLX finishes printing the current line before halting.) Press space again to

restart the display. To break out of the display and get back to the menu before the ending address is reached, press RETURN.

Other Menu Options

Two more menu selections let you save programs and load them back into the computer. These are SAVE FILE and LOAD FILE; their operation is quite straightforward. When you press S or L, MLX asks you for the filename. You'll then be asked to press either D or T to select disk or tape.

You'll notice the disk drive starting and stopping several times during a load or save. Don't panic; this is normal behavior. MLX opens and reads from or writes to the file instead of using the usual LOAD and SAVE commands. Disk users should also note that the drive prefix 0: is automatically added to the filename (line 750), so this should *not* be included when entering the name. This also precludes the use of @ for Save-with-Replace, so remember to give each version you save a different name.

Remember that MLX saves the entire workspace area from the starting address to the ending address, so the save or load may take longer than you might expect if you've entered only a small amount of data from a long listing. When saving a partially completed listing, make sure to note the address where you stopped typing so you'll know where to resume entry when you reload.

MLX reports the standard disk or tape error messages if any problems are detected during the save or load. (Tape users should bear in mind that Commodore computers are never able to detect errors during a save to tape.) MLX also has three special load error messages: INCORRECT STARTING ADDRESS, which means the file you're trying to load does not have the starting address you specified when you ran MLX; LOAD ENDED AT *address*, which means the file you're trying to load ends before the ending address you specified when you started MLX; and TRUNCATED AT ENDING ADDRESS, which means the file you're trying to load extends beyond the ending address you specified when you started MLX. If you see one of these messages and feel certain that you've loaded the right file, exit and rerun MLX, being careful to enter the correct starting and ending addresses.

The QUIT menu option has the obvious effect—it stops MLX and enters BASIC. The RUN/STOP key is disabled, so the Q option lets you exit the program without turning off the computer. (Of course, RUN/STOP–RESTORE also gets you out.) You'll be asked for verification; press Y to exit to BASIC, or any other key to return to the menu. After quitting, you can type RUN again and reenter MLX without losing your data, as long as you don't use the clear workspace option.

The Finished Product

When you've finished typing all the data for an ML program and saved your work, you're ready to see the results. The instructions for loading and using the finished product vary from program to program. Some ML programs are designed to be loaded and run like BASIC programs, so all you need to type is LOAD *"filename"*,8 for disk or LOAD *"filename"* for tape, and then RUN. Such programs will usually have a starting address of 0801 for the 64. Other programs must be reloaded to specific addresses with a command such as LOAD *"filename"*,8,1 for disk or LOAD *"filename"*,1,1 for tape, then started with a SYS to a particular memory address. On the Commodore 64, the most common starting address for such programs is 49152, which corresponds to MLX address C000. In either case, you should always refer to the article which accompanies the ML listing for information on loading and running the program.

An Ounce Of Prevention

By the time you finish typing in the data for a long ML program, you may have several hours invested in the project. Don't take chances—use our "Automatic Proofreader" to type the new MLX, and then test your copy *thoroughly* before first using it to enter any significant amount of data. Make sure all the menu options work as they should. Enter fragments of the program starting at several different addresses, then use the Display option to verify that the data has been entered correctly. And be sure to test the Save and Load options several times to ensure that you can recall your work from disk or tape. Don't let a simple typing error in the new MLX cost you several nights of hard work.

MLX For Commodore 64

```
SS 10 REM VERSION 1.1: LINES 8
      30,950 MODIFIED, LINES 4
      85-487 ADDED
EK 100 POKE 56,50:CLR:DIM IN$,
      I,J,A,B,A$,B$,A(7),N$
DM 110 C4=48:C6=16:C7=7:Z2=2:Z
      4=254:Z5=255:Z6=256:Z7=
      127
CJ 120 FA=PEEK(45)+Z6*PEEK(46)
      :BS=PEEK(55)+Z6*PEEK(56
      ):H$="0123456789ABCDEF"
SB 130 R$=CHR$(13):L$="{LEFT}"
      :S$=" ":D$=CHR$(20):Z$=
      CHR$(0):T$="{13 RIGHT}"
CQ 140 SD=54272:FOR I=SD TO SD
      +23:POKE I,0:NEXT:POKE
      {SPACE}SD+24,15:POKE 78
      8,52
FC 150 PRINT"{CLR}"CHR$(142)CH
      R$(8):POKE 53280,15:POK
      E 53281,15
EJ 160 PRINT T$" {RED}{RVS}
      {2 SPACES}[8 @]
      {2 SPACES]"SPC(28)"
      {2 SPACES}{OFF}{BLU} ML
      X II {RED}{RVS}
      {2 SPACES}"SPC(28)"
      {12 SPACES}{BLU}"
FR 170 PRINT"{3 DOWN}
      {3 SPACES}COMPUTE!'S MA
      CHINE LANGUAGE EDITOR
      {3 DOWN}"
JB 180 PRINT"{BLK}STARTING ADD
      RESS[43";:GOSUB300:SA=A
      D:GOSUB1040:IF F THEN18
      0
GF 190 PRINT"{BLK}{2 SPACES}EN
      DING ADDRESS[43";:GOSUB
      300:EA=AD:GOSUB1030:IF
      {SPACE}F THEN190
KR 200 INPUT"{3 DOWN}{BLK}CLEA
      R WORKSPACE [Y/N][43";A
      $:IF LEFT$(A$,1)<>"Y"TH
      EN220
PG 210 PRINT"{2 DOWN}{BLU}WORK
      ING...";:FORI=BS TO BS+
      EA-SA+7:POKE I,0:NEXT:P
      RINT"DONE"
DR 220 PRINTTAB(10)"{2 DOWN}
      {BLK}{RVS} MLX COMMAND
      {SPACE}MENU {DOWN}[43":
      PRINT T$"{RVS}E{OFF}NTE
      R DATA"
BD 230 PRINT T$"{RVS}D{OFF}ISP
      LAY DATA":PRINT T$"
      {RVS}L{OFF}OAD FILE"
JS 240 PRINT T$"{RVS}S{OFF}AVE
       FILE":PRINT T$"{RVS}Q
      {OFF}UIT{2 DOWN}{BLK}"
JH 250 GET A$:IF A$=N$ THEN250
HK 260 A=0:FOR I=1 TO 5:IF A$=
      MID$("EDLSQ",I,1)THEN A
      =I:I=5
FD 270 NEXT:ON A GOTO420,610,6
      90,700,280:GOSUB1060:GO
      TO250
EJ 280 PRINT"{RVS} QUIT ":INPU
      T"{DOWN}[43ARE YOU SURE
       [Y/N]";A$:IF LEFT$(A$,
      1)<>"Y"THEN220
EM 290 POKE SD+24,0:END
JX 300 INS=N$:AD=0:INPUTIN$:IF
      LEN(IN$)<>4THENRETURN
KF 310 B$=IN$:GOSUB320:AD=A:B$
      =MID$(IN$,3):GOSUB320:A
      D=AD*256+A:RETURN
PP 320 A=0:FOR J=1 TO 2:A$=MID
      $(B$,J,1):B=ASC(A$)-C4+
      (A$>"@")*C7:A=A*C6+B
JA 330 IF B<0 OR B>15 THEN AD=
      0:A=-1:J=2
GX 340 NEXT:RETURN
CH 350 B=INT(A/C6):PRINT MID$(
      H$,B+1,1);:B=A-B*C6:PRI
      NT MID$(H$,B+1,1);:RETU
      RN
RR 360 A=INT(AD/Z6):GOSUB350:A
      =AD-A*Z6:GOSUB350:PRINT
      ":";
BE 370 CK=INT(AD/Z6):CK=AD-Z4*
      CK+Z5*(CK>Z7):GOTO390
PX 380 CK=CK*Z2+Z5*(CK>Z7)+A
JC 390 CK=CK+Z5*(CK>Z5):RETURN
QS 400 PRINT"{DOWN}STARTING AT
      [43";:GOSUB300:IF IN$<>
      N$ THEN GOSUB1030:IF F
      {SPACE}THEN400
EX 410 RETURN
HD 420 PRINT"{RVS} ENTER DATA
      {SPACE}":GOSUB400:IF IN
      $=N$ THEN220
JK 430 OPEN3,3:PRINT
SK 440 POKE198,0:GOSUB360:IF F
```

```
        THEN PRINT IN$:PRINT"
        {UP}{5 RIGHT}";
GC 450  FOR I=0 TO 24 STEP 3:B$
        =S$:FOR J=1 TO 2:IF F T
        HEN B$=MID$(IN$,I+J,1)
HA 460  PRINT"{RVS}"B$L$;:IF I<
        24THEN PRINT"{OFF}";
HD 470  GET A$:IF A$=N$ THEN470
FK 480  IF(A$>"/"ANDA$<":")OR(A
        $>"@"ANDA$<"G")THEN540
GS 485  A=-(A$="M")-2*(A$=",")-
        3*(A$=".")-4*(A$="/")-5
        *(A$="J")-6*(A$="K")
FX 486  A=A-7*(A$="L")-8*(A$=":
        ")-9*(A$="U")-10*(A$="I
        ")-11*(A$="O")-12*(A$="
        P")
CM 487  A=A-13*(A$=S$):IF A THE
        N A$=MID$("ABCD123E456F
        0",A,1):GOTO 540
MP 490  IF A$=R$ AND((I=0)AND(J
        =1)OR F)THEN PRINT B$;:
        J=2:NEXT:I=24:GOTO550
KC 500  IF A$="{HOME}" THEN PRI
        NT B$:J=2:NEXT:I=24:NEX
        T:F=0:GOTO440
MX 510  IF(A$="{RIGHT}")ANDF TH
        ENPRINT B$L$;:GOTO540
GK 520  IF A$<>L$ AND A$<>D$ OR
        ((I=0)AND(J=1))THEN GOS
        UB1060:GOTO470
HG 530  A$=L$+S$+L$:PRINT B$L$;
        :J=2-J:IF J THEN PRINT
        {SPACE}L$;:I=I-3
QS 540  PRINT A$;:NEXT J:PRINT
        {SPACE}S$;
PM 550  NEXT I:PRINT:PRINT"{UP}
        {5 RIGHT}";:INPUT#3,IN$
        :IF IN$=N$ THEN CLOSE3:
        GOTO220
QC 560  FOR I=1 TO 25 STEP3:B$=
        MID$(IN$,I):GOSUB320:IF
         I<25 THEN GOSUB380:A(I
        /3)=A
PK 570  NEXT:IF A<>CK THEN GOSU
        B1060:PRINT"{BLK}{RVS}
        {SPACE}ERROR: REENTER L
        INE [4]":F=1:GOTO440
HJ 580  GOSUB1080:B=BS+AD-SA:FO
        R I=0 TO 7:POKE B+I,A(I
        ):NEXT
QQ 590  AD=AD+8:IF AD>EA THEN C
        LOSE3:PRINT"{DOWN}{BLU}
        ** END OF ENTRY **{BLK}
        {2 DOWN}":GOTO700
GQ 600  F=0:GOTO440
QA 610  PRINT"{CLR}{DOWN}{RVS}
        {SPACE}DISPLAY DATA ":G
        OSUB400:IF IN$=N$ THEN2
        20
RJ 620  PRINT"{DOWN}{BLU}PRESS:
         {RVS}SPACE{OFF} TO PAU
        SE, {RVS}RETURN{OFF} TO
         BREAK[4]{DOWN}"
KS 630  GOSUB360:B=BS+AD-SA:FOR
        I=BTO B+7:A=PEEK(I):GOS
        UB350:GOSUB380:PRINT S$
        ;
CC 640  NEXT:PRINT"{RVS}";:A=CK
        :GOSUB350:PRINT
KH 650  F=1:AD=AD+8:IF AD>EA TH
        ENPRINT"{DOWN}{BLU}** E
        ND OF DATA **":GOTO220
KC 660  GET A$:IF A$=R$ THEN GO
        SUB1080:GOTO220
EQ 670  IF A$=S$ THEN F=F+1:GOS
        UB1080
AD 680  ONFGOTO630,660,630
CM 690  PRINT"{DOWN}{RVS} LOAD
        {SPACE}DATA ":OP=1:GOTO
        710
PC 700  PRINT"{DOWN}{RVS} SAVE
        {SPACE}FILE ":OP=0
RX 710  IN$=N$:INPUT"{DOWN}FILE
        NAME[4]";IN$:IF IN$=N$
        {SPACE}THEN220
PR 720  F=0:PRINT"{DOWN}{BLK}
        {RVS}T{OFF}APE OR {RVS}
        D{OFF}ISK: [4]";
FP 730  GET A$:IF A$="T"THEN PR
        INT"T{DOWN}":GOTO880
HQ 740  IF A$<>"D"THEN730
HH 750  PRINT"D{DOWN}":OPEN15,8
        ,15,"I0:":B=EA-SA:IN$="
        0:"+IN$:IF OP THEN810
SQ 760  OPEN 1,8,8,IN$+",P,W":G
        OSUB860:IF A THEN220
FJ 770  AH=INT(SA/256):AL=SA-(A
        H*256):PRINT#1,CHR$(AL)
        ;CHR$(AH);
PE 780  FOR I=0 TO B:PRINT#1,CH
        R$(PEEK(BS+I));:IF ST T
        HEN800
FC 790  NEXT:CLOSE1:CLOSE15:GOT
        O940
GS 800  GOSUB1060:PRINT"{DOWN}
        {BLK}ERROR DURING SAVE:
        [4]":GOSUB860:GOTO220
MA 810  OPEN 1,8,8,IN$+",P,R":G
        OSUB860:IF A THEN220
GE 820  GET#1,A$,B$:AD=ASC(A$+Z
        $)+256*ASC(B$+Z$):IF AD
        <>SA THEN F=1:GOTO850
RX 830  FOR I=0 TO B:GET#1,A$:P
        OKE BS+I,ASC(A$+Z$):IF(
        I<>B)AND ST THEN F=2:AD
        =I:I=B
FA 840  NEXT:IF ST<>64 THEN F=3
FQ 850  CLOSE1:CLOSE15:ON ABS(F
        >0)+1 GOTO960,970
SA 860  INPUT#15,A,A$:IF A THEN
         CLOSE1:CLOSE15:GOSUB10
        60:PRINT"{RVS}ERROR: "A
        $
GQ 870  RETURN
EJ 880  POKE183,PEEK(FA+2):POKE
        187,PEEK(FA+3):POKE188,
        PEEK(FA+4):IFOP=0THEN92
        0
HJ 890  SYS 63466:IF(PEEK(783)A
        ND1)THEN GOSUB1060:PRIN
        T"{DOWN}{RVS} FILE NOT
        {SPACE}FOUND ":GOTO690
CS 900  AD=PEEK(829)+256*PEEK(8
        30):IF AD<>SA THEN F=1:
        GOTO970
SC 910  A=PEEK(831)+256*PEEK(83
        2)-1:F=F-2*(A<EA)-3*(A>
        EA):AD=A-AD:GOTO930
KM 920  A=SA:B=EA+1:GOSUB1010:P
        OKE780,3:SYS 63338
JF 930  A=BS:B=BS+(EA-SA)+1:GOS
        UB1010:ON OP GOTO950:SY
        S 63591
AE 940  GOSUB1080:PRINT"{BLU}**
         SAVE COMPLETED **":GOT
        O220
XP 950  POKE147,0:SYS 63562:IF
        {SPACE}ST>0 THEN970
FR 960  GOSUB1080:PRINT"{BLU}**
         LOAD COMPLETED **":GOT
        O220
DP 970  GOSUB1060:PRINT"{BLK}
        {RVS}ERROR DURING LOAD:
        {DOWN}[4]":ON F GOSUB98
        0,990,1000:GOTO220
PP 980  PRINT"INCORRECT STARTIN
        G ADDRESS (";:GOSUB360:
        PRINT")":RETURN
GR 990  PRINT"LOAD ENDED AT ";:
        AD=SA+AD:GOSUB360:PRINT
         D$:RETURN
FD 1000 PRINT"TRUNCATED AT END
        ING ADDRESS":RETURN
RX 1010 AH=INT(A/256):AL=A-(AH
        *256):POKE193,AL:POKE1
        94,AH
FF 1020 AH=INT(B/256):AL=B-(AH
        *256):POKE174,AL:POKE1
        75,AH:RETURN
FX 1030 IF AD<SA OR AD>EA THEN
        1050
HA 1040 IF(AD>511 AND AD<40960
        )OR(AD>49151 AND AD<53
        248)THEN GOSUB1080:F=0
        :RETURN
HC 1050 GOSUB1060:PRINT"{RVS}
        {SPACE}INVALID ADDRESS
         {DOWN}{BLK}":F=1:RETU
        RN
AR 1060 POKE SD+5,31:POKE SD+6
        ,208:POKE SD,240:POKE
        {SPACE}SD+1,4:POKE SD+
        4,33
DX 1070 FOR S=1 TO 100:NEXT:GO
        TO1090
PF 1080 POKE SD+5,8:POKE SD+6,
        240:POKE SD,0:POKE SD+
        1,90:POKE SD+4,17
AC 1090 FOR S=1 TO 100:NEXT:PO
        KE SD+4,0:POKE SD,0:PO
        KE SD+1,0:RETURN
```

©

MLX Machine Language Entry Program For Apple

Tim Victor, Editorial Programmer

To make it easier to enter machine language programs into your computer without typos, COMPUTE! is introducing its "MLX" entry program for the Apple II series. It's our best MLX yet. It runs on the II, II+, IIe, and IIc, and with either DOS 3.3 or ProDOS.

A machine language (ML) program is usually listed as a long series of numbers. It's hard to keep your place and even harder to avoid making mistakes as you type in the listing, since an incorrect line looks almost identical to a correct one. To make error-free entry easier, COMPUTE! generally lists ML programs for Commodore and Atari computers in a format designed to be typed in with a utility called "MLX." The MLX program uses a checksum system to catch typing errors almost as soon as they happen.

Apple MLX checks your typing on a line-by-line basis. It won't let you enter invalid characters or let you continue if there's a mistake in a line. It won't even let you enter a line or digit out of sequence. Best of all, you don't have to know anything about machine language to enter ML programs with MLX. Apple MLX makes typing ML programs almost foolproof.

Using Apple MLX

Type in and save some copies of Apple MLX on disk (you'll want to use MLX to enter future ML programs in COMPUTE!). It doesn't matter whether you type it in on a disk formatted for DOS 3.3 or ProDOS. Programs entered with Apple MLX, however, must be saved to a disk formatted with the same operating system as Apple MLX itself.

If you have an Apple IIe or IIc, make sure that the key marked CAPS LOCK is in the down position. Type RUN. You'll be asked for the starting and ending addresses of the ML program. These values vary for each program, so they're given at the beginning of the ML program listing and in the program's accompanying article. Find them and type them in.

The next thing you'll see is a menu asking you to select a function. The first is (E)NTER DATA. If you're just starting to type in a program, pick this. Press the E key, and the program asks for the address where you want to begin entering data. Type the first number in the first line of the program listing if you're just starting, or the line number where you left off if you've already typed in part of a program. Hit the RETURN key and begin entering the data.

Once you're in Enter mode, Apple MLX prints the address for each program line for you. You then type in all nine numbers on that line, beginning with the first two-digit number after the colon (:). Each line represents eight bytes and a checksum. When you enter a line and hit RETURN, Apple MLX recalculates the checksum from the eight bytes and the address. If you enter more or less than nine numbers, or the checksum doesn't exactly match, Apple MLX erases the line you just entered and prompts you again for the same line.

Invalid Characters Banned

Apple MLX is fairly flexible about how you type in the numbers. You can put extra spaces between numbers or leave the spaces out entirely, compressing a line into 18 keypresses. Be careful not to put a space between two digits in the middle of a number. Apple MLX will read two single-digit numbers instead of one two-digit number (F 6 means F and 6, not F6).

You can't enter an invalid character with Apple MLX. Only the numerals 0–9 and the letters A–F can be typed in. If you press any other key (with some exceptions noted below), nothing happens. This safeguards against entering extraneous characters. Even better, Apple MLX checks for transposed characters. If you're supposed to type in A0 and instead enter 0A, Apple MLX will catch your mistake.

Apple MLX also checks to make sure you're typing in the right line. The address (the number to the left of the colon) is part of the checksum recalculation. If you accidentally skip a line and try to enter incorrect values, Apple MLX won't let you continue. Just make sure you enter the correct starting address; if you don't, you won't be able to enter any of the following lines. Apple MLX will stop you.

Editing Features

Apple MLX also includes some editing features. The left- and right-arrow keys allow you to back up and go forward on the line that you are entering, so you can retype data. Pressing the CONTROL (CTRL) and D keys at the same time (*delete*) removes the character under the cursor, shortening the line by one character. Pressing CTRL-I (*insert*) puts a space under the cursor and shifts the rest of the line to the right, making the line one character longer. If the cursor is at the right end of the line, neither CTRL-D nor CTRL-I has any effect.

When you've entered the entire listing (up to the ending address that you specified earlier), Apple MLX automatically leaves Enter mode and redisplays the functions menu. If you want to leave Enter mode before then, press the RETURN key when Apple MLX prompts you with a new line address. (For instance, you may want to leave Enter mode to enter a program listing in more than one sitting; see below.)

Display Data

The second menu choice, (D)ISPLAY DATA, examines memory and shows the contents in the same format as the program listing. You can use it to check your work or to see how far you've gotten. When you press D, Apple MLX asks you for a starting address. Type in the address of the first line you want to see and hit RETURN. Apple MLX displays program lines until you press any key or until it reaches the end of the program.

Save And Load

Two more menu selections let you save programs on disk and load them back into the computer. These are (S)AVE FILE and (L)OAD FILE. When you press S or L, Apple MLX asks you for the filename. The first time you save an ML program, the name you assign will be the program's filename on the disk. If you press L and specify a filename that doesn't exist on the disk, you'll see a disk error message.

If you're not sure why a disk error has occurred, check the drive. Make sure there's a formatted disk in the drive and that it was formatted by the same operating system you're using for Apple MLX (ProDOS or DOS 3.3). If you're trying to save a file and see an error message, the disk might be full. Either save the file on another disk or quit Apple MLX (by pressing the Q key), delete an old file or two, then run Apple MLX again. Your typing should still be safe in memory.

cursor, shortening the line by one character. Pressing CTRL-I (*insert*) puts a space under the cursor and shifts the rest of the line to the right, making the line one character longer. If the cursor is at the right end of the line, neither CTRL-D nor CTRL-I has any effect.

When you've entered the entire listing (up to the ending address that you specified earlier), Apple MLX automatically leaves Enter mode and redisplays the functions menu. If you want to leave Enter mode before then, press the RETURN key when Apple MLX prompts you with a new line address. (For instance, you may want to leave Enter mode to enter a program listing in more than one sitting; see below.)

Display Data

The second menu choice, (D)ISPLAY DATA, examines memory and shows the contents in the same format as the program listing. You can use it to check your work or to see how far you've gotten. When you press D, Apple MLX asks you for a starting address. Type in the address of the first line you want to see and hit RETURN. Apple MLX displays program lines until you press any key or until it reaches the end of the program.

Save And Load

Two more menu selections let you save programs on disk and load them back into the computer. These are (S)AVE FILE and (L)OAD FILE. When you press S or L, Apple MLX asks you for the filename. The first time you save an ML program, the name you assign will be the program's filename on the disk. If you press L and specify a filename that doesn't exist on the disk, you'll see a disk error message.

If you're not sure why a disk error has occurred, check the drive. Make sure there's a formatted disk in the drive and that it was formatted by the same operating system you're using for Apple MLX (ProDOS or DOS 3.3). If you're trying to save a file and see an error message, the disk might be full. Either save the file on another disk or quit Apple MLX (by pressing the Q key), delete an old file or two, then run Apple MLX again. Your typing should still be safe in memory.

Apple MLX: Machine Language Entry Program

For instructions on entering this program, please refer to "COMPUTE!'s Guide to Typing In Programs" elsewhere in this issue.

```
80 100 N = 9: HOME : NORMAL : PR
       INT "APPLE MLX": POKE 34,
       2: ONERR GOTO 610
CC 110 VTAB 1: HTAB 20: PRINT "S
       TART ADDRESS";: GOSUB 530
       : IF A = 0 THEN PRINT CHR
       $ (7): GOTO 110
8C 120 S = A
EJ 130 VTAB 2: HTAB 20: PRINT "E
       ND ADDRESS  ";: GOSUB 530
       : IF S > = A OR A = 0 THE
       N PRINT CHR$ (7): GOTO 13
       0
20 140 E = A
B5 150 PRINT : PRINT "CHOOSE:(E)
       NTER DATA";: HTAB 22: PRI
       NT "(D)ISPLAY DATA": HTAB
        8: PRINT "(L)OAD FILE  (
       S)AVE FILE  (Q)UIT": PRIN
       T
AE 160 GET A$: FOR I = 1 TO 5: I
       F A$ < > MID$ ("EDLSQ",I,
       1) THEN NEXT : GOTO 160
93 170 ON I GOTO 270,220,180,200
       : POKE 34,0: END
AF 180 INPUT "FILENAME: ";A$: IF
        A$ < > "" THEN PRINT CHR
       $ (4);"BLOAD";A$;",A";S
A1 190 GOTO 150
38 200 INPUT "FILENAME: ";A$: IF
        A$ < > "" THEN PRINT CHR
       $ (4);"BSAVE";A$;",A";S;"
       ,L";(E - S) + 1
92 210 GOTO 150
C2 220 GOSUB 590: IF B = 0 THEN
       150
9E 230 FOR B = B TO E STEP 8:L =
        4:A = B: GOSUB 580: PRIN
       T A$;": ";:L = 2
85 240 FOR F = 0 TO 7:V(F + 1) =
        PEEK (B + F): NEXT : GOS
       UB 560:V(9) = C
F2 250 FOR F = 1 TO N:A = V(F):
       GOSUB 580: PRINT A$" ";:
       NEXT : PRINT : IF PEEK (4
       9152) < 128 THEN NEXT
94 260 POKE 49168,0: GOTO 150
CC 270 GOSUB 590: IF B = 0 THEN
       150
48 280 FOR B = B TO E STEP 8
A6 290 HTAB 1:A = B:L = 4: GOSUB
        580: PRINT A$;": ";: CAL
       L 64668:A$ = "":P = 0: GO
       SUB 330: IF L = 0 THEN 15
       0
F9 300 GOSUB 470: IF F < > N THE
       N PRINT CHR$ (7);: GOTO 2
       90
27 310 IF N = 9 THEN GOSUB 560:
       IF C < > V(9) THEN PRINT
       CHR$ (7);: GOTO 290
72 320 FOR F = 1 TO 8: POKE B +
       F - 1,V(F): NEXT : PRINT
       : NEXT : GOTO 150
8E 330 IF LEN (A$) = 33 THEN A$
       = O$:P = O: PRINT CHR$ (7
       );
22 340 L = LEN (A$):O$ = A$:O =
       P:L$ = "": IF P > 0 THEN
       L$ = LEFT$ (A$,P)
E0 350 R$ = "": IF P < L - 1 THE
       N R$ = RIGHT$ (A$,L - P -
        1)
55 360 HTAB 7: PRINT L$;: FLASH
       : IF P < L THEN PRINT MID
       $ (A$,P + 1,1);: NORMAL :
        PRINT R$;
7B 370 PRINT " ";: NORMAL
E6 380 K = PEEK (49152): IF K <
       128 THEN 380
C1 390 POKE 49168,0:K = K - 128
5B 400 IF K = 13 THEN HTAB 7: PR
       INT A$;" ";: RETURN
8A 410 IF K = 32 OR K > 47 AND K
        < 58 OR K > 64 AND K < 7
       1 THEN A$ = L$ + CHR$ (K)
        + R$:P = P + 1
C1 420 IF K = 4 THEN A$ = L$ + R
       $
5F 430 IF K = 9 THEN A$ = L$ + "
        " + MID$ (A$,P + 1,1) +
       R$
0A 440 IF K = 8 THEN P = P - (P
       > 0)
93 450 IF K = 21 THEN P = P + (P
       < L)
9D 460 GOTO 330
37 470 F = 1:D = 0: FOR P = 1 TO
        LEN (A$):C$ = MID$ (A$,P
       ,1): IF F > N AND C$ < >
       " " THEN RETURN
B8 480 IF C$ < > " " THEN GOSUB
       520:V(F) = J + 16 * (D =
       1) * V(F):D = D + 1
5F 490 IF D > 0 AND C$ = " " OR
       D = 2 THEN D = 0:F = F +
       1
08 500 NEXT : IF D = 0 THEN F =
       F - 1
17 510 RETURN
B5 520 J = ASC (C$):J = J - 48 -
        7 * (J > 64): RETURN
AB 530 A = 0: INPUT A$:A$ = LEFT
       $ (A$,4): IF LEN (A$) = 0
        THEN RETURN
6F 540 FOR P = 1 TO LEN (A$):C$
       = MID$ (A$,P,1): IF C$ <
       "0" OR C$ > "9" AND C$ <
       "A" OR C$ > "Z" THEN A =
       0: RETURN
2D 550 GOSUB 520:A = A * 16 + J:
        NEXT : RETURN
28 560 C = INT (B / 256):C = B -
        254 * C - 255 * (C > 127
       ):C = C - 255 * (C > 255)
20 570 FOR F = 1 TO 8:C = C * 2
       - 255 * (C > 127) + V(F):
       C = C - 255 * (C > 255):
       NEXT : RETURN
DA 580 I = FRE (0):A$ = "": FOR
       I = 1 TO L:T = INT (A / 1
       6):A$ = MID$ ("0123456789
       ABCDEF",A - 16 * T + 1,1)
        + A$:A = T: NEXT : RETUR
       N
1F 590 PRINT "FROM ADDRESS ";: G
       OSUB 530: IF S > A OR E <
        A OR A = 0 THEN B = 0: R
       ETURN
0D 600 B = S + 8 * INT ((A - S)
       / 8): RETURN
B6 610 PRINT "DISK ERROR": GOTO
       150
```

©

Classified

SOFTWARE

COMMODORE: TRY BEFORE YOU BUY. Best selling games, utilities, educational, + classics and new releases. 100's of titles. Visa/MC. Free brochure. RENT-A-DISC, Frederick Bldg. #345, Hunt'n, WV 25701 (304) 529-3232

FREE APPLE SOFTWARE
Over 1,000 Public Domain Programs on 50 diskettes. $5 each plus $1 for shipping per order. Send $1 for catalog, refundable with order.
C&H ENTERPRISES
PO Box 29243, Memphis, TN 38127

DISCOUNT SOFTWARE for most computers. FREE CATALOG. Sale: 5.25" DSDD Disks 25 for $13.95 ppd. WMJ DATA SYSTEMS-C, 4 Butterfly Dr., Hauppauge, NY 11788

Cheap Software/IBM,PCjr, Compatibles. 5.25 or 3.5 disk. For catalog of diskettes available for education, games, business, etc., Write: Morning Star Industry, Box 3095, Ann Arbor, MI 48106

FREE SOFTWARE for C64, C128, IBM & CPM send SASE for info (specify computer) to: PUBLIC DOMAIN USERS GROUP PO Box 1442-A1, Orange Park, FL 32067

MILLION $ "LOTTO BUSTER" PROGRAM
SCIENTIFIC. HITS JACKPOTS. NO RN/G $49.95 Ap II, Atari, MS/DOS, C64. OH res add 6% s.tax. Also: FANTASTIC DAILY NUMBER FORECASTER! GETS Straight Hits Weekly! $42.95. Both: $79.95. Large SASE gets info. Z-Way, POB 9017-C, Canton OH 44711 (216) 875-2512

More than 200 great ML routines for 64 and 128, ready to add to your own programs, in COMPUTE! Books' MACHINE LANGUAGE ROUTINES FOR THE COMMODORE 64/128. Explanations, uses, commented source code. 585 pages, $18.95. Check your local bookstore or call (800) 346-6767

FREE LIST: CARTRIDGES/SOFTWARE FOR: COLECO, ATARI 2600, 5200, 7800, NINTENDO, ADAM, XL, XE, Intellivision: Modem: 414-265-5149, 24 hrs, Villa Video, Box 17085, Milw, WI 53217

TI-99/4A Software/Hardware bargains. Hard-to-find items. Huge selection. Fast service. Free catalog. D.E.C., Box 690, Hicksville, NY 11801

CALC RESULT 64 - SPREADSHEET SOFTWARE
C/R 64 EASY - Single page Cartridge $25
C/R 64 ADVANCED - 3D - Cart/Disk 32 pages w/Help, Graphics, Windows only $45
Text: Applied C/R w/32 Examples $10
Call The Order Line 201-445-5998 MC/VISA

LEARN THE CONSTITUTION! FUN Bicentennial educational game for IBM: $35 (req's BASIC), or for C64: $25. Basic Fundamentals, 3362 S 2300 E, SLC, UT 84109

FREE PUBLIC DOMAIN SOFTWARE - Request free catalog or send $2 for sample disk & catalog (refundable). APPLE, C64-128 (specify) CALOKE IND., Box 18477, KC, MO 64133

FREE! PUBLIC DOMAIN SOFTWARE FLYER! IBM & Compatibles - (MS-DOS) Save $$$ @ $3.50 per disk! Write to: AP-JP Inc., Box 1155, W. Babylon, NY 11704

WE WILL PAY YOU $20 IN COLD HARD CASH if your computer can't earn $15,000 with our business plan. Send $1 (s/h) to: 1111 Logan Ave. Unit C, Box 106, Winnipeg, MB R3E 1R1 Canada. Specify computer model!

IBM PUBLIC DOMAIN SOFTWARE $3 PER DISK
Send stamp for cat. Hundreds of disks to choose from. Excellent service. Two disk sizes now available: 5¼" @ $3.00 and 3½" @ $5.00. Send for your list. Now accepting MasterCard and Visa...
JDX/C, P.O. BOX 1561, CORONA, CA 91718

$$ L O T T O U S A $$
Pick the winning numbers in all states that have LOTTO games.Easy to run menu. Lottery games for daily picks,LOTTO 40, 48 etc.Diskette only C-64 or IBM.Allow 3-4 weeks for delivery.Send $19.95 + $3.50 S/H.New York residents add tax. To: QUEENS SOFTWARE PO Box 7279 Grand Central Station N.Y.,N.Y. 10163-6028

$$ WIN $$
with our
SPORTS
Handicapping
Software

THOROUGHBRED/HARNESS Handicapping..$29.95, enhanced..$49.95. GREYHOUND Handicapping..$29.95, enhanced..$49.95. Up to 13 entries analyzed: class, speed, times, odds, track, post, stretch, & finish positions. Pro Football System..$39.95. Pro Basketball System.. $49.95. Specify disk/tape: Apple II+ce, Atari, IBM PC, COM 64/128, TI, TRS-80 Mod 3/4, Mod 100/200. Color. Add $2 p&h. MC/VISA accepted. Free info. SOFTWARE EXCHANGE, P.O. Box 5382-CP, W. Bloomfield, MI 48033. (313) 626-7208. Orders: 1-800-527-9467.

TI-99/4A QUALITY SOFTWARE for Business, Home and Entertainment ** Bonus Software Offer! ** Send for FREE catalog to MICRO-BIZ HAWAII, BOX 1108, PEARL CITY, HI 96782

FOOTBALL COMPU-SCHED™: 1987 NFL SCHEDULE/Team logs. 1986 standings/team logs/playoff results. Enter scores - create new standings/team logs. $19.95 + $1 s/h (CA res add tax) College confs available. IBM-PC or compatible. Other software: Football Stats-alyzer, Sportscard Inventory. Free brochures. Cosoft Micro Systems, Dept. CO, 26458 Mocine Ave., Hayward, CA 94544

CANADIANS: MAIL ORDER SOFTWARE FOR YOU. Titles by Abacus, Activision, Epyx and more. Low prices, wide selection, free catalog. CANSOFT, Dept. G, Box 3464, Courtenay BC V9N 5N5

HARDWARE

PC CLONE BUILDER'S MANUAL with parts list. Detailed instr for Assembly, Testing & Troubleshooting. Covers all switches, jumpers & Hard Disk installation: $19.95. Digital Solutions, 26 E 14th St. #505C, Indpls, IN 46202

MISCELLANEOUS

SAFEWARE INSURES COMPUTERS against fire, theft, & power surges for as little as $39. Call Safeware, The Insurance Agency Inc. at 800/848-3469, Columbus, Ohio.

PERSONAL COMPUTER OWNERS CAN EARN $1000 to $5000 monthly selling simple services part time. Free list of 100 best services. Write: A.I.M.H.U., PO Box 60369, San Diego, CA 92106-8369

OVER 100 SPECIAL COMPUTER BARGAINS!
Send $5.95 ch/mo for 12-issue subscription of The Bit Parade. Or send $1 cash for sample. Mel-Mark Software Int., Inc., POB 2014-P, Medford Lakes, NJ 08055 (609) 654-9698

COMPUTE! Classified is a low-cost way to tell over 350,000 microcomputer owners about your product or service.

Rates: $25 per line, minimum of four lines. Any or all of the first line set in capital letters at no charge. Add $15 per line for boldface words, or $50 for the entire ad set in boldface (any number of lines.) Inquire about display rates.
Terms: Prepayment is required. Check, money order, American Express, Visa, or MasterCard is accepted. Make checks payable to COMPUTE! Publications.
Form: Ads are subject to publisher's approval and must be either typed or legibly printed. One line equals 40 letters and spaces between words. Please underline words to be set in boldface.
General Information: Advertisers using post office box numbers in their ads must supply permanent address and telephone numbers. Ad will appear in next available issue after receipt.
Closing: 10th of the third month preceding cover date (e.g., June issue closes March 10th). Send order and remittance to: Harry Blair, Classified Manager, COMPUTE!, P.O. Box 5406, Greensboro, NC 27403. To place an ad by phone, call Harry Blair at (919) 275-9809.
Notice: COMPUTE! Publications cannot be responsible for offers or claims of advertisers, but will attempt to screen out misleading or questionable copy.

Advertisers Index

COMPUTE!

A WORTHWHILE INVESTMENT

You may have easily spent thousands on your computer. Whatever you spent, it was a wise and worthwhile investment! Now make your computer even more valuable by subscribing to Compute!, the leading magazine of home, educational and recreational computing.

☐ **1 Year/$24** ☐ **2 Years/$45** JOC14

Name _____

Address _____

City _____ State _____ Zip _____

☐Payment Enclosed ☐ Bill Me Later

For faster service call: 1-800-727-6937

For Foreign & Canadian Subscribers, please add $6 (U.S.) per year postage.

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY MAIL

FIRST CLASS PERMIT NO. 7478 DES MOINES, IA

POSTAGE WILL BE PAID BY ADDRESSEE

COMPUTE!

P.O. Box 10955
Des Moines, IA 50347-0955

COMPUTE!'s
FREE Reader Information Service

Use these cards to request FREE information about the products advertised in this issue. Clearly print or type your full name and address. Only one card should be used per person. Circle the numbers that correspond to the key number appearing in the advertisers index.

Send in the card and the advertisers will receive your inquiry. Although every effort is made to insure that only advertisers wishing to provide product information have reader service numbers, COMPUTE! cannot be responsible if advertisers do not provide literature to readers.

Please use these cards *only* for subscribing or for requesting product information. Editorial and customer service inquiries should be addressed to: COMPUTE!, P.O. Box 5406, Greensboro, NC 27403. Check the expiration date on the card to insure proper handling.

Use these cards and this address only for COMPUTE!'s Reader Information Service. Do not send with payment in any form.

COMPUTE!

101	102	103	104	105	106	107	108	109	110	111	112	113	114	115	116	117
118	119	120	121	122	123	124	125	126	127	128	129	130	131	132	133	134
135	136	137	138	139	140	141	142	143	144	145	146	147	148	149	150	151
152	153	154	155	156	157	158	159	160	161	162	163	164	165	166	167	168
169	170	171	172	173	174	175	176	177	178	179	180	181	182	183	184	185
186	187	188	189	190	191	192	193	194	195	196	197	198	199	200	201	202
203	204	205	206	207	208	209	210	211	212	213	214	215	216	217	218	219
220	221	222	223	224	225	226	227	228	229	230	231	232	233	234	235	236
237	238	239	240	241	242	243	244	245	246	247	248	249	250	251	252	253

Circle 101 for a one year new U.S. subscription to COMPUTE!: you will be billed for $24.

Please let us know. Do you own: plan to buy:

- ☐ 270 Apple _____ ☐ 271
- ☐ 272 Atari _____ ☐ 273
- ☐ 274 Commodore _____ ☐ 275
- ☐ 276 IBM _____ ☐ 277
- ☐ 278 TI-99/4A _____ ☐ 279
- ☐ 280 Other _____ (specify model) ☐ 281

Please print or type name and address.
Limit one card per person.

Name _____

Address _____

City _____

State/Province _____ Zip _____

Country _____

Phone _____

Please Include ZIP Code

Expiration Date 11/30/87 GA1087

SUBSCRIBE TO COMPUTE!

For Fastest Service, Call Our **Toll-Free** US Order Line
800-727-6937

☐ $24.00 One Year US Subscription
☐ $45.00 Two Year US Subscription

Name _____

Address _____

City _____ State _____ Zip _____

☐ Payment Enclosed ☐ Bill me
Charge my: ☐ VISA ☐ MasterCard ☐ American Express
Account No. _____ Expires ____/____

Your subscription will begin with the next available issue. Please allow 4-6 weeks for delivery of first issue. Subscription prices subject to change at any time. Outside the U.S.A. please add $6 for each subscription.

JOC48

Place
Stamp
Here

COMPUTE! Reader Service
P.O. Box 2141
Radnor, PA 19089

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

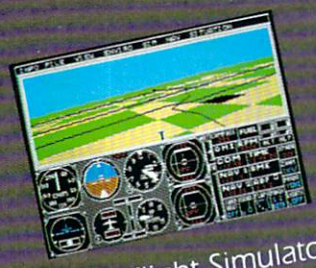
BUSINESS REPLY MAIL
FIRST CLASS PERMIT NO. 7478 DES MOINES, IA

POSTAGE WILL BE PAID BY ADDRESSEE

COMPUTE!

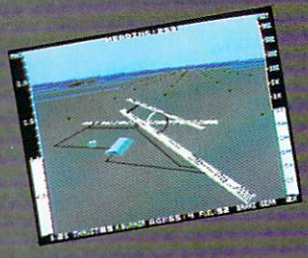
P.O. Box 10955
Des Moines, IA 50347-0955

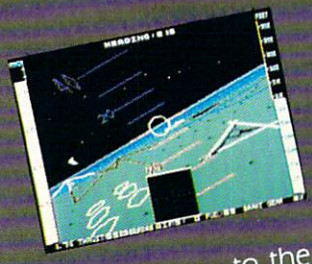
THE FLIGHT CHOICE!



Atari ST screens shown.
Other computer versions may vary.

From the sophisticated realism of Flight Simulator...





IBM PC screens

...to the thrills and excitement of Jet...

Atari ST screens

...to new Scenery Disk adventures... SubLOGIC. Make The Flight Choice.

See Your Dealer. For additional product ordering information or the name of the dealer nearest you, call (800) 637-4983.

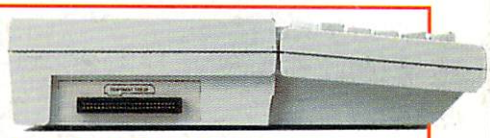
subLOGIC Corporation
713 Edgebrook Drive
Champaign IL 61820
(217) 359-8482 Telex: 206995
ORDER LINE: (800) 637-4983
(except in Illinois)

The standard equipment on a Laser 128 is optional on most computers

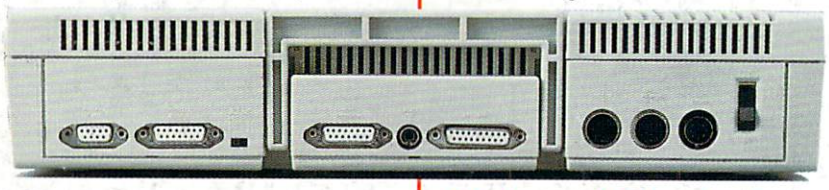
Take a look at the Laser 128 Apple®-compatible computer, and beginning with its built-in disk drive and 128K RAM of memory, you start to feel that this machine was designed with you in mind. The Laser 128 is ready-to-run – plug it in, insert your program and go. Everything is built-in for you.

Choose from the largest software library in the world. The Laser 128 runs Apple IIe and IIc software, which makes it nice for you and your family, because children can use the same programs at home that they learn on in school. The Laser 128 is a computer for the whole family.

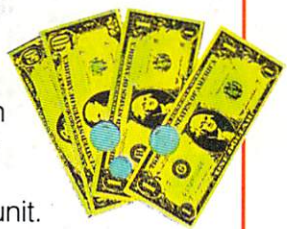
The Laser 128 has built-in interfaces for all your peripherals, including serial and parallel printer interfaces, modem and mouse interfaces, a game port and you can hookup a color or monochrome monitor, or even a TV to it. Only the Laser has this much built-in as standard equipment.



It's expandable, too. The Laser 128 includes an expansion slot to let the computer grow with your increasing demands.



The Laser 128 is priced to keep money in your pocket. The Laser is about half the price of an Apple, and is even less than a comparably equipped Commodore unit. You get a lot of computer for a little money.



Look for the attractive Laser packaging at a store near you. For more information on the Laser 128 and the name of your nearest dealer, contact Video Technology Computers, Inc., 400 Anthony Trail, Northbrook, IL 60062, or call (312) 272-6760.

LASER 128®

Manufactured by
VIDEO TECHNOLOGY COMPUTERS, INC.

MAKING COMPUTERS AFFORDABLE