ST.GAME

computer gaming for the irreverent

CONTEST: YOU CAN BE A COMPUTER HERO

VOLUME 4 MARCH-APRIL 1984 $2.00

—CONTENTS—

St.Words: Margot Comstock Tommervik, editus maximus; Andrew Christie, editus realus; Carol Ray, editus managum; Matt Yuen, Aristotlean editus; David Durkee, Pythagorean technicus; Todd Zilbert, Forum discussion directus; Cord Cooper, grammatike correctum; Judith Pfeffer, assistant grammatike correctums; Harry McNeil and Steve Thomsen, Phoenicians checkum; Betsy Barnes, goddess of order; Hartley G. Lesser, Sophoclean game judge; Roe R. Adams, Herculean infomainman; Al Tommervik, ancient grease.

St.Art: Trici Venola, Venus in blue jeans; Nancy Baldwin, Muse of aesthetes; Lucas "Lamia" McClure, Ruth "Sprite" Seid, papyrus pasteup; Michael Pender, advertus placius; Weldon O. Lewin, Dan Winkler, daepaintedes pictures; Kurt Wahlner, photi shootum.

St.Bureaucrats: Mary Sue Rennells, Athenian ruler, Marjorie Kaufman, goddess of petty decisions; David Stockman, budget deficits; Cathy Stewart, advertus vendum; Pattie Lesser, goddess of retailers; Michelle Vigneault-Kirschenbaum, circulum subscribia.

St.Ad Sales: East Coast sales, Paul McGinnis Company, 690 Broadway, Massapequa, NY 11758, (212) 490-1021; Midwest and Rocky Mountain sales, Ted Rickard, Kevin Sullivan, Christopher Kaspar, Market/Media Associates, 435 Locust Road, Wilmette, IL 70091, (312) 251-2541; West Coast sales, Mike Antich, St.Game, 7250 Laurel Canyon Boulevard, North Hollywood, CA 91605, (818) 980-5074.

St.Game. Volume 3, Number 3. Copyright © 1984 by Softalk Publishing Inc. All rights reserved, whatever that means. ISSN: 0745-4988, whatever that is. *St.Game* is published around the fifteenth afternoon of every other month by Softalk Publishing Inc., 7250 Laurel Canyon Boulevard, North Hollywood, California; (818) 980-5074.

Composition by the demigods at Photographics, Hollywood, California. Printing by the semigods at Volkmuth Printers, Saint Cloud, Minnesota.

Here we go again: Apple is a trademark of Apple Computer Inc., Cuptertino, California. Atari is a trademark of Atari Inc., Sunnyvale, California. Commodore 64 and VIC-20 are trademarks of Commodore Business Machines, West Chester, Pennsylvania. *Zork* is a trademark of Infocom Inc., Cambridge, Massachusetts. *St.Game*, Softalk, and Infomania are trademarks of Softtalk Publishing Inc. We could go on forever, but you get the idea.

Subscriptions: Write to the snarling mythical beasts and furry creatures at St.Game Circulation, Box 60, North Hollywood, CA 91603. For a sacrificial offering of $12 a year, you get six mind-blowing issues delivered to your door by an actual government civil servant. Be sure to include your name (and what you're the god/goddess of), address, and the make, model, and serial number of your computer. Tell us anything else you'd like us to know.

Moving? For the good of the neighborhood, hopefully. Just send your old mailing label and new address to St.Game Circulation, Box 60, North Hollywood, CA 91603, or call us at (818) 980-5074.

Postmaster: How's it feel to have your own department in this mag? In return, we ask that you send address changes (readers', not yours—unless you are one) to St.Game Circulation, Box 60, North Hollywood, CA 91603. Thanks, babe.

Cover design by Trici Venola

—ADVERTISERS—

Broken Timber Press	53	MTR Enterprises	9
Electronic Arts	25, Cover 4	Sierra On-Line	13, Cover 3
Infocom	6-7	SubLogic Corp.	Cover 2

GAME LIBRARY

Like a bastion of computer culture before the onslaught of micro mediocrity, a shining beacon of gaming goodness cutting through the murky miasma of just so-so software, the Game Library deals only with the right stuff. It lives to serve. All Recommended programs are the best of the current releases and carry the very highest recommendation. The programs in the Stacks were once worthy enough to be so called and have since graciously stepped aside to make room for new blood. In neither of these categories would you be risking wasting your money. New Releases includes any damn fool thing that meets the single criterion of its category, and the items are largely unseen except where obviously otherwise. Please line up on the left; have your I/O open and your RAM ready.

RECOMMENDED

Bruce Lee. Fortier. *Atari.* Hii-yaah! An obvious labor of love, and a fitting tribute to the late, great chop-socky cinema star. He's even wearing his little black gloves. Twenty gorgeous screens; best full-contact karate action available on a home computer.

Datasoft, 19808 Nordhoff Place, Chatsworth, CA 91311. $34.95.

International Soccer. *Commodore 64.* Best of its type. Players cast shadows and always appear in correct proportion relative to their position on the field.

Commodore Software, 1200 Wilson Drive, West Chester, PA 19380. $34.95.

One-on-One. Hammond. *Apple.* Dr. J and Larry Bird jump, dribble, rebound, slam dunk, and occasionally shatter the backboard in this highly impressive basketball simulation.

Electronic Arts, 2755 Campus Drive, San Mateo, CA 94403. $40.

Seven Cities of Gold. Ozark Softscape. *Atari.* Outfit a voyage to the New World, circa 1492. Watch for shallows, ration your food supply, travel overland, get gold, establish missions, interact with five levels of natives. Play in historical mode or use random continent generator. That's right, bunky: It's all done in full-screen animated graphics. Endless joy; a genuine kick in the head.

Electronic Arts, 2755 Campus Drive, San Mateo, CA 94403. $40.

Sorcerer. Meretzky. *Most computers.* Instant Infocom classic. Loads of magic, a 3-D transparent glass cube maze, and a truly wicked ending puzzle. Pirating it will do you no good at all.

Infocom, 55 Wheeler Street, Cambridge, MA 02138. $49.95.

NEW ARRIVALS

Fiction

Boulder Dash. Liepa, Gray. *Atari, Commodore 64, PCjr.* Grab the jewels while undermining the boulders, ever mindful of their position vis-à-vis you. This game displays a masterful grasp of gravity and kinetics; may start its own genre. First Star, 22 East Forty-First Street, New York, NY 10017. $29.95.

C'est La Vie. Eastman. *Apple, Atari, Commodore 64.* The streets are lined with moola, but thieves and the IRS are all over the place. A loan from the neighborhood shark may help, but pay off on time. Adventure International, Box 3435, Longwood, FL 32750. $19.95.

Code Breaker. Barker. *IBM PC.* Play against the computer to break a secret code. Four skill levels. Requires color/graphics adapter. Sylvan Glen Software, Box 31053, Des Peres, MO 63131. $39.95.

Computer Title Bout. *Atari.* A statistical smorgasbord, based on the *Sports Illustrated* board game. Over five hundred boxers on a double-sided disk, rated for defense, endurance, killer instinct, ability to take a punch, and so on. Match up real boxers or roll your own. Microcomputer Games/Avalon Hill, 4517 Harford Road, Baltimore, MD 21214. $30.

The Dallas Quest. Garon. *Commodore 64.* A missing map gives the location of a South American oil field. To get it, you have to go up against J.R. Ewing. A sure-'nuff adventure, on a considerably higher intellectual plane than its source material. Meticulous, impressive hi-res graphics. Datasoft, 19808 Nordhoff Place, Chatsworth, CA 91311. $34.95.

Dawn of Civilization. Schwenk. *Atari.* Family strategy game for one to four players. Build legions, cultivate food, and manage land to ensure the future of mankind. Superware, 2028 Kingshouse Road, Silver Spring, MD 20904. $10.95.

Dreadnoughts. Dowell. *Apple.* Lively action in the North Atlantic, 1939–1941. Virtually every warship of the Allied and Axis navies that was involved in the real thing puts in an appearance, in minute strategic detail. Microcomputer Games/Avalon Hill, 4517 Harford Road, Baltimore, MD 21214. $30.

Encounter. Woakes. *Atari.* Very fluid 3-D *Battlezone*-ish action. You just tootle around a barren landscape shooting at fast-moving abstract objects. What big-screen teevee systems were made for. Synapse, 5221 Central Avenue, Richmond, CA 94804. $34.95.

Free Trader. Bell, Chung. *Apple, Atari.* A long time from now in a galaxy, far, far away . . . you'll still have to take an option on a hot commodities deal and ensure that your products achieve their market share while maintaining an acceptable profit margin. And raiders from the Thoth Empire want you for the lead in *Death of a Salesman*. Microcomputer Games/Avalon Hill, 4517 Harford Road, Baltimore, MD 21214. $30.

Genesis. Furry, Alvarez, Brodie. *Apple, Atari, Commodore 64.* Spiders crawl up the walls of a pit toward your plucky scorpion perched on the edge. *Tempest* in a teapot; lukewarm. Datasoft, 19808 Nordhoff Place, Chatsworth, CA 91311. $29.95.

Gulf Strike. Chung. *Atari.* A what-if brigade-level simulation in the Persian Gulf. Featuring U.S. vs. U.S.S.R vs. Iraq vs. Iran. Microcomputer Games/Avalon Hill, 4517 Harford Road, Baltimore, MD 21214. $25.

Gumby. *Atari, Commodore 64.* The licensing agreement with Art Clokey Productions solemnly stipulates that this program be "for educational purposes only," but those of us who have long known and loved the little clay guy will probably have fun anyway. Datasoft, 19808 Nordhoff Place, Chatsworth, CA 91311. $34.95.

Juno First. Hiscott. *Commodore 64.* The Konami arcade hit lands on the home computer. Another lone spaceship pilot gets jumped by endless waves of hostile aliens. Eight-way joystick action. Datasoft, 19808 Nordhoff Place, Chatsworth, CA 91311. $29.95.

Jupiter Mission 1999. Lamb. *Atari.* Now you can get there two years before HAL and all those monoliths. Promising-looking four-disk graphic adventure with arcade sequences. Microcomputer Games/Avalon Hill, 4517 Harford Road, Baltimore, MD 21214. $50.

Minnesota Fats' Pool Challenge. Bubble Bus. *Commodore 64, IBM PC.* One or two players, six game options. HesWare, 150 North Hill Drive, Brisbane, CA 94005. $29.95.

Music Box. Holland. *Commodore 64.* Gather up the pieces of a magical music box while dodging a wizard on a flying dragon playing a dangerous tune. Get the box together and its magical song will cleanse the universe of evil, it says here. HesWare, 150 North Hill Drive, Brisbane, CA 94005. $29.95.

Quasimodo. Atack, Bigelow. *Atari, Commodore 64.* The famous bell-ringer must get past the palace guards, swing through the bell towers, and scale the castle wall, overcoming obstacles like bats, archers, a wizard, and his own lack of coordination in this multiscreen arcader. Synapse, 5221 Central Avenue, Richmond, CA 94804. $34.95.

Quick Step. Johnson. *Atari.* Catch burgers, cakes, and sundaes before they roll off the screen. Then bounce the check over to your opponent. Not unlike life itself. This is available only via the electronic distribution system of Xante Corporation. Imagic, 981 University Avenue, Los Gatos, CA 95030. $24.95.

Rainbow Quest. Ramella. *CoCo, Commodore 64.* A book/software package. The reader/player helps Molly and Sam survive on the planet Rainbow, enduring a pirate's privations and other hazards via the twenty-five arcade, adventure, and word and logic puzzle programs on the accompanying cassette. WGBooks, Peterborough, NH 03485. $24.97.

Regatta. DeMuth, Peterson. *Apple.* Sharpen your sailing skills in four lake races, mindful of buoys and running aground, whilst running, reaching, tacking, and yarning. One or two players. Howard W. Sams, 4300 West Sixty-Second Street, Indianapolis, IN 46268. $29.95.

Rootin' Tootin'. Nesbitt. *Atari, Commodore 64.* Your French horn travels a maze of scales, blasting notes at pursuing instruments. Anybody remember the old Disney cartoon that looked like this? HesWare, 150 North Hill Drive, Brisbane, CA 94005. $34.95.

Space Cowboy. Lamb. *Atari.* Take a stroll down a narrow, sixteen-sector walkway to freedom. Tip yer hat to them ranny owl-hoot robot lasers and pitfalls. Microcomputer Games/Avalon Hill, 4517 Harford Road, Baltimore, MD 21214. $21.

Who-Dun-It? *Commodore 64, VIC 20.* Not to mention how-dun-it? and where-is-it? The lights go out; a woman screams; a necklace is missing. Gather your clues, question twenty-odd suspects, make your arrest. VIC 20 requires 16K expansion. Ramiak, 3133 Nucla Street, Aurora, CO 80013. $19.95.

Nonfiction

BBS Directory. Listing of more than seven hundred North American computer bulletin boards by state, area code, and number, plus info on type of system, password requirement, program downloading. Includes overview of how to use a BBS and available software. BBS Directory, Box 4215, Beach Station, Vero Beach, FL 32964. $5.95.

The First Artificial Intelligence Coloring Book. Cohen. Computer artist Harold Cohen chats with kids about decision making, coloring, and art in general. With thirty-five computer drawings, ready to color. William Kaufmann, 95 First Street, Los Altos CA 94022. $13.95.

Relax. *Apple, Atari, Commodore, IBM.* Let your computer calm you down. Disk, headband, control unit, and audiotape combine techniques of biofeedback and autogenic training in one stress-reduction system. Includes mandala program and balloon game. Available June. Synapse, 5221 Central Avenue, Richmond,CA 94804. $139.95.

Sing a Song of Software. Soltzberg. Computer-illustrated poems about computers, software, and the meaning of it all. Micro-chic moves ahead. William Kaufmann, 95 First Street, Los Altos, CA 94022. $8.95.

Reference

Basic Magic. *Atari.* A series of little pamphlets consisting of programming tricks for stuff like speeding up your joysticks and listing your disk directory without loading DOS. You can't beat the price. Superware, 2028 Kingshouse Road, Silver Spring, MD 20904. Free.

Basic Tricks for the Apple. Wyatt. Thirty-five subroutine shortcuts to more efficient programming so your code can load easier, run faster, jump higher, and live longer. Howard W. Sams, 4300 West Sixty-Second Street, Indianapolis, IN 46268. $8.95.

Game Designer 64. *Commodore 64.* Programming utility lets you design and animate game characters and background screens. Entech, 8224 Sunland Boulevard, Sun Valley, CA 91352. $35.

Graph-Art System 1. *Apple.* Graphic utility includes a coloring program, eraser, inverse pix, tracing sheet, graphic/text mix mode, more. Apollo Software, Box 34057, Houston, TX 77234. $34.

THE STACKS

Check marks denote games recently translated to other computers.

Fiction

Adventure

The Coveted Mirror. Berns, Thomason. *Apple.* Warm-hearted, well-puzzled, pretty hi-res graphics adventure, interspersed with minigames and a couple of simple arcade-type sequences. Talk to everyone in a medieval village to find missing piece of magic mirror, all on borrowed time. Penguin Software, 830 Fourth Avenue, Geneva, IL 60134. $19.95.

Deadline and **Witness.** Blank, Lebling. *Most computers.* The first Interlogic Mystery. *Deadline* has player as detective search for clues, interrogate suspects, show clues and findings to suspects to get revealing reactions. Complete interaction, real-time. *Witness* does the same, but on an introductory level. All text, but who cares? Infocom, 55 Wheeler Street, Cambridge, MA 02138. $49.95 each.

Enchanter I. Blank, Lebling. *Most computers.* The first of a planned trilogy that will succeed the Zorks. Above-ground exploration of castle and its environs in search of lords of darkness. Enchanters use magic logically to solve puzzles, which are great. A delight to play and read. Infocom, 55 Wheeler Street, Cambridge, MA 02138. $49.95.

Exodus: Ultima III. British. *Apple, Atari.* By far the best in the Ultima (so far) trilogy. Role-playing fantasy with good plot, everything integrated, clues everywhere—and there's a lot more to everywhere than meets the eye. Multiple heroes. Many hours of colorful, animated, solid role-playing fantasy in Lord British's inimitable style. Origin Systems, Box 58009, Houston, TX 77058. Atari, 48K. $54.95.

Infidel. Berlyn. *Most computers.* Infocom's Egyptological adventure: Find pyramid, uncover pyramid, find entrance to pyramid . . . then it gets tough. A November release, kicking off the company's Tales of Adventure series. Infocom, 55 Wheeler Street, Cambridge, MA 02138. $49.95.

Masquerade. Johnson. *Apple.* Illustrated adventure with diabolical puzzles, wonderful graphics in minor plot. But the puzzles are all hard but satisfying. Last one's a dog. Shaggy. Phoenix Software, 64 Lake Zurich Drive, Lake Zurich, IL 60047. $34.95.

Planetfall. Meretzky. *Most computers.* A lighthearted and strongly involving sci-fi text adventure with characters who care. Was Steve Meretzky an adventurer or a writer first? It's hard to tell—that's how rich the text is. Fascinating exploring, good puzzles, well-written text. Infocom, 55 Wheeler Street, Cambridge, MA 02138. $49.95.

Zork I, II, III. Blank, Lebling. *Most computers.* Classic trilogy of incredibly graphic text adventures by the masters of the genre. Pure logic, intelligent rapport, wit. A complete Underground Empire, at your service. *I* and *II* are archetypical; *III* is unique, with highly challenging puzzles. Infocom, 55 Wheeler Street, Cambridge, MA 02138. $39.95 each.

Arcade

Astro Chase. Herrera. *Atari, Commodore 64.* Flashy style and attention to detail in this new variation on an old theme. The game is standard operating procedure (save Earth, or else), but alien variety, really tremendous graphics, and rich sound push this one past the point of being "just another compuier game." Parker Brothers, 50 Dunham Road, Beverly, MA 01915. $36.

Bandits. Ngo, Bertoni. *Apple, VIC-20.* Just about the only differences between the VIC cart and the 48K original are fewer, simpler colors and a limited repeat-shot function. Also, incredibly, the animation is actually smoother at the highest levels. Sirius, 10364 Rockingham Drive, Sacramento, CA 95827. $39.95.

Beach-Head. Carver. *Commodore 64.* A one- or two-player, joystick-controlled, six-sequence WW II naval engagement, the equivalent of six different (yes, different!) games. Action, sound, design, all tops. Truly fine arcading. Access Software, 925 East 900 South, Salt Lake City, UT 84105. $34.95.

Choplifter. Gorlin. *Apple, Atari, Commodore 64, VIC-20.* The most awarded computer game in history has virtually no variety in repeat play, but it's a grabber nonetheless. Broderbund, 17 Paul Drive, San Rafael, CA 94903. $34.95.

Crossfire. Sullivan. *Apple, Commodore 64, IBM, VIC-20.* There's just one level, but only a masochist would want more; no game is the same. Smoothly programmed classic challenges some of today's bests. Sierra On-Line, Sierra On-Line Building, Coarsegold, CA 93614. $29.95.

Dino Eggs. Schroeder. *Adam, Apple, Atari, IBM.* The ultimate in prehistoric wildlife preservation. A dense, ineffable arcading experience that reveals its pleasures gradually and allows the development of myriad individual strategies. Micro Fun, 2699 Skokie Valley Road, Highland Park, IL 60035. $40.

Drol. Beng. *Apple, Atari, Commodore 64.* Very cute, but not too cute; very hard, but it just makes you try harder. At the moment, it is unquestionably the most graphically delightful, impressively animated arcade action around. Broderbund, 17 Paul Drive, San Rafael, CA 94903. $34.95.

Frogger. Harris, Benton. *Atari, Commodore 64.* What would any computer be without this classic conversion of the arcade hit? Disk or cartridge, it doesn't matter; both versions do full justice to the original. Sierra On-Line, Sierra On-Line Building, Coarsegold, CA 93614. $34.95. Parker Brothers, 50 Dunham Road, Beverly, MA 01915. $36.

Hard Hat Mack. Abbott, Alexander. *Apple, Commodore 64.* Despite the competition, *Mack* holds its own. Fast and smooth animation coupled with a stiff challenge make up for its three-screen limit. One original concept in the area of nonviolence: You can't kill those who hound you; you can only try to avoid them. Electronic Arts, 2755 Campus Drive, San Mateo, CA 94403. $35.

Lode Runner. Smith. *Apple, Atari, Commodore 64, IBM.* Here's a game that brings a new meaning to the word *variety*. One hundred fifty levels. If there are any reasons to be addicted to a game, Doug Smith has found them all and included them in this baby. Play the game; cheat if you want; construct your own levels. The Apple never looked so good. Broderbund, 17 Paul Drive, San Rafael, CA 94903. $34.95.

Miner 2049er. Hogue. *Most computers.* The game that even software pirates decided to buy. Ignore the plot; it doesn't make too much sense anyway. Boot it up (or plug it in) and have a ball. Three levels even on the lowest-end machines, ten on the high ones. Bill Hogue broke new ground when this one debuted; it's subsequently become the most translated game in microdom. Big Five, Box 9078-185, Van Nuys, CA 91409. $49.95. Micro Fun, 2699 Skokie Valley Road, Highland Park, IL 60035. $39.95.

Night Mission Pinball. Artwick. *Apple, Commodore 64, IBM.* User-adjustable classic; ten play modes and forty parameters, for your customizing pleasure. Twenty-page manual. SubLogic, 713 Edgebrook Drive, Champaign, IL 61820. Disk or cassette. $29.95.

Shamus: Case 2. Mataga. *Atari.* The movie industry could learn something from this game—a sequel that surpasses the original. Shamus is back in the same survival game, facing the Shadow and aquatic mutants. Several dozen screens means you'll have something to do on rainy days. Synapse, 5221 Central Avenue, Richmond, CA 94804. $34.95.

✓ **Spare Change.** Zeller, Zeller. *Apple, Atari.* Slapstick comedy in an arcade setting. Not Broderbund's most timeless work, but thoroughly enjoyable. Whimsical theme and multiple cartoon intermissions. Broderbund, 17 Paul Drive, San Rafael, CA 94903. $34.95.

Zaxxon. Fortier, Garcia. *Apple, Atari, Commodore 64, TRS-80.* Here's one of the best computer adaptations of a coin-op game ever. Extraordinary special effects for a computer game; never has such an impossible effect been simulated so well. Datasoft, 19808 Nordhoff Place, Chatsworth, CA 91311. Synapse (C-64), 5221 Central Avenue, Richmond, CA 94804. $39.95.

Fantasy

Knight of Diamonds. Greenberg, Woodhead. *Apple.* Second scenario in the *Wizardry* series has more puzzles to solve. Requires strong characters developed in *Proving Grounds.* Sir-tech, 6 Main Street, Ogdensburg, NY 13669. $39.95.

Legacy of Llylgamyn. Greenberg, Woodhead. *Apple.* Topnotch episode of a topnotch game features super atmosphere, neat puzzles, and an eerie transformation of *Wizardry* characters to their own progeny. Sir-tech, 6 Main Street, Ogdensburg, NY 13669. $39.95.

Wizardry: Proving Grounds of the Mad Overlord. Greenberg, Woodhead. *Apple, IBM.* Original of the *Wizardry* series, *Proving Grounds* is essential for creating characters for all other scenarios. Most involving of all role-playing games. *Wizardry* characters are very personal and traverse the dungeon in groups of six, interacting and supporting each other under player's guidance. Sir-tech, 6 Main Street, Ogdensburg, NY 13669. $49.95.

Strategy

Archon. Reiche, Freeman, Westfall. *Atari, Commodore 64.* Arcade games too fast? Strategy games too slow? This Atari masterpiece combines the best of both worlds. As someone who found chess to be boring beyond belief, author Paul Reiche did something about it, along with fantasy game pros Jon Freeman and Anne Westfall. If you want to take an opponent's square, you fight for it—in hi-res. Skilled spell-casters will have the upper hand in this thoroughly enjoyable battle between light and dark forces. Electronic Arts, 2755 Campus Drive, San Mateo, CA 94403. $40.

Geopolitique 1990. Ketchledge. *Apple.* As the chief executive, you call the shots. In the diplomatic first part of this two-phase game, the United States and Soviet Union struggle for economic and political world dominance. In phase two, you play "what if" in a conventional war situation. Seven scenarios ranging from today's global situation to the hypothetical 1990s are yours for the choosing. Strategic Simulations, 883 Stierlin Road, A-200, Mountain View, CA 94043. $39.95.

Gnosis VII. *Apple.* Unique logic game. No graphics, doesn't need 'em. Determine the secret names of seven gods who move between seven towns. Player must make a living, worship, learn skills, earn sanctuary by figuring out which god rules what and which towns that god is currently influencing. Mind-stretching. Program is totally open, with line guide to key routines printed in manual. Magnetic Harvest, Box 255, Hopkins, SC 29061. $19.95.

I.Q. Baseball. *Apple.* Beautifully executed, amusing baseball trivia game. Major and minor league levels—minor is still about the majors but fine for duffers; major league's only for superbuffs. Answering correctly moves players around the diamond—singles, doubles, and triples, depending on difficulty of question. Rain delays and a relief pitcher, too. Davka, 845 North Michigan Avenue, Chicago, IL 60611. $24.95. Supplementary disks by team (American and National leagues and World Series). $14.95.

North Atlantic '86. Grigsby. *Apple.* Be either the Soviet aggressor or the commander of NATO forces in this hypothetical battle. The Soviets have taken control of Germany and Norway and now plan to swallow Great Britain and the North Atlantic. *Possible, realistic,* and *scary* are the words to describe the scenario. The Soviets' attempt to take Iceland and NATO's efforts to keep England supplied make for some interesting parallels between this confrontation and World War II. Strategic Simulations, 883 Stierlin Road, A-200, Mountain View, CA 94043. $59.95.

Pensate. Besnard. *Apple.* Computer-age abstract strategy game is fast moving and fun for those to whom thinking is good entertainment. Penguin Software, 830 Fourth Avenue, Geneva, IL 60134. $19.95.

Sargon III. Spracklen, Spracklen. *Apple.* Manages to play chess both fast and well. Far and away the most powerful chess program for the microcomputer. Hayden, 600 Suffolk Street, Lowell, MA 01853. $49.95.

TAC. Bosson. *Apple, Atari, Commodore 64, IBM.* It stands for Tactical Armor Command. Detail is what sets this strategy game apart from the rest. This is no board game on screen; tanks and turrets actually move and turn! Distinguished by its quick execution of commands (no more fixing sandwiches while the computer deploys forces) and variables so numerous that the game could be played only on a computer. Microcomputer Games/Avalon Hill, 4517 Harford Road, Baltimore, MD 21214. $40.

Nonfiction

Music Construction Set. Harvey. *Apple.* In the same vein as its pinball predecessor, this gem puts two-voice music through the Apple's speaker, or up to six voices through Mockingboard-equipped machines. No knowledge of music is required to write, edit, or transpose music. Also lets you use music from within your own programs. Bravo. Electronic Arts, 2755 Campus Drive, San Mateo, CA 94403. $35.

NEW NAME, SAME GAME

Yes, it's still us.

We're not under new management. We didn't file Chapter 11 and reorganize. We didn't have to get out of the country in a hurry and leave a lot of angry subscribers holding the bag . . . no, no, no.

We just have a new name.

For you old-timers, this is *Softline* you're reading.

For you new people, just ignore all this. *St.Game* is the name of the magazine. We like it. We think it swings. We think it goes a bit farther toward saying what we're all about—a computer game magazine published by Softalk Publishing Inc.

As many of you are indeed seeing us for the first time, that probably doesn't explain it all. For you, then, this is a computer game magazine that takes computer games seriously (but not much else). We don't do VCS stuff. We don't tell you how to dress for the arcades. We don't use three picture-soaked pages to review one 8K cartridge. We just get down to the business of high-grade computer fun—how to have it, how to get better at it, how to make your own, and how much you can expect to have in the future. We've been doing that for a while; now we're moving into high gear and making lots of new friends, and we figured we should have a new name for the occasion.

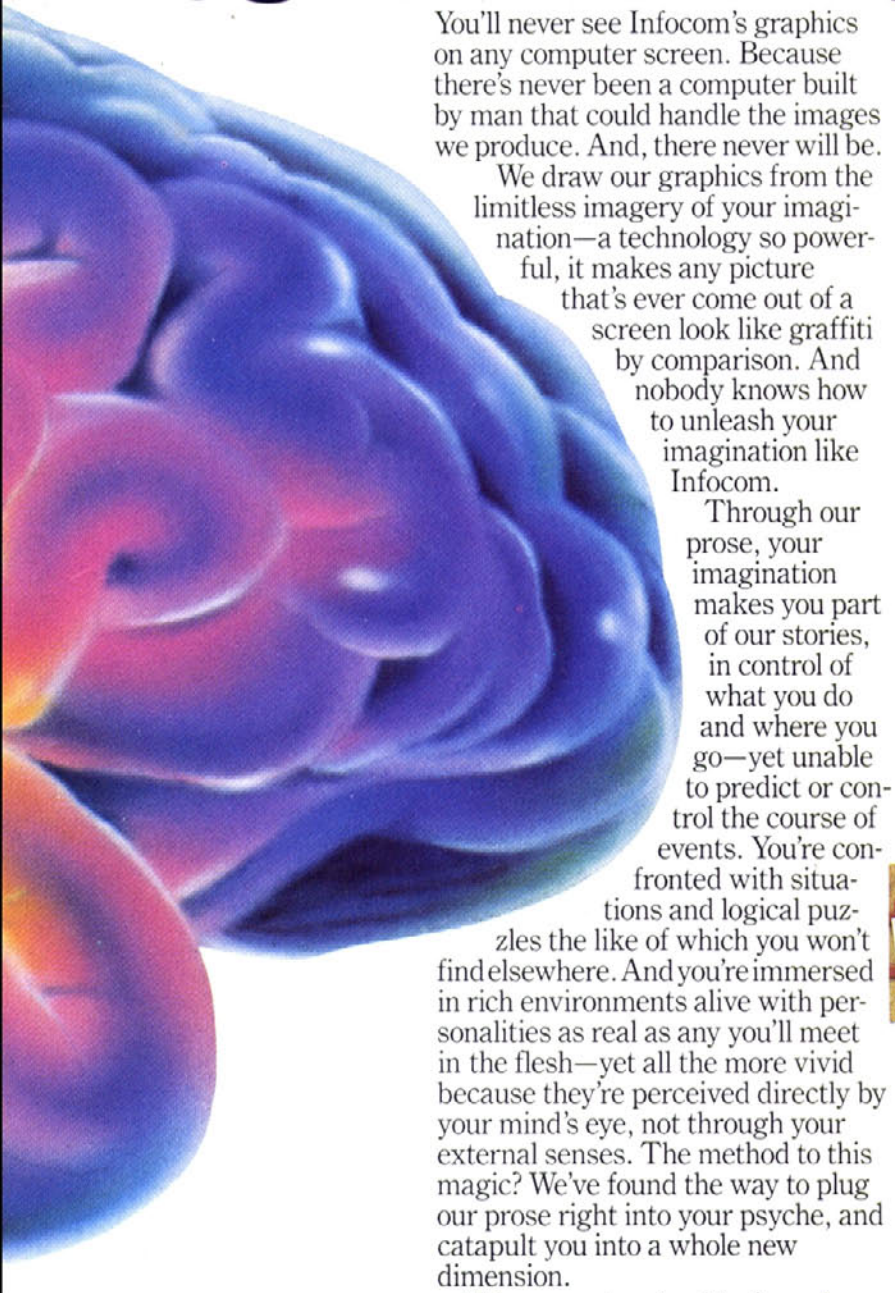
It was a very emotional decision. We still haven't changed the stationery. Softline was a great old name and it will live in our hearts forever. It is not truly gone, because it's still here. And this is still it. Only different.

Bless you.

Photo: Kurt Wahlner

WE UNLEASH TH
POWERFUL GRAP

E WORLD'S MOST HICS TECHNOLOGY.



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

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

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

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

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

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

(For more information on Infocom games contact: Infocom, Inc., P.O. Box 855, Garden City, NY 11530.)

INFOCOM™

The next dimension.

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

A GAME'S LIFE

Secrets of Selling Software

by Philip Chien

You've written the most fantastic game for your computer and you want to sell a million copies. What should you do to make sure that everyone who sees your program will want to buy it?

First, make a flashy demo for your program—something to catch the eye. Color graphics and animated displays are always appreciated. Design your disk so that when it is booted it will go straight into demo mode, without the user even hitting a keystroke. Make sure that the title page seen upon boot-up is a flashy display containing the name of the program, the copyright notice, your name and/or company name and/or hi-res company logo, and how to get out of the demo (for example, "Hit escape to start game"). Don't overdo the title page, though; it shouldn't be an advertisement. Let the demo and program do that. The demo should work on every system possible, even systems that don't have enough memory or accessories (like paddles or joysticks) to use the full program.

If it's within your budget, send a free dealer copy to each authorized dealer. Dealers love free copies! Then, about two weeks later, call each dealer and see what they like and don't like (for future revisions), and solicit sales. Offer good dealer margins; offer better margins for large orders. When you send dealer demos, make them in as normal an operating system as possible. Some dealers may want to put their copy of your program on a hard disk drive or give copies to all of their salespeople. This won't be easy if you've put the program in a modified version of the operating system and made it uncopyable. Don't worry about copies of your demo getting out; this only makes more advertising for you.

When advertising, whether through direct mail or magazine ads, remember: attract attention! Make sure your ad is concise, lists all the features of your program, describes the system requirements (memory, media, accessories, languages, and so on), and tells *which computers the program works on*! (One recent ad in a major computer magazine failed to mention that the program would only work on a TRS-80.) Just because you're advertising in an Apple-only magazine doesn't mean that readers will know your product is for the Apple (among other things, you may wish to run the ad in other, non-Apple magazines). Don't just say that the program is "Apple IIe-compatible." Does IIe-compatible mean that the program won't work with a 64K Apple II? Does it mean that the eighty-column version won't work on an Apple II with a Videoterm card, or on a IIe with a Videoterm card (because it only works with the Apple IIe eighty-column card)? Will it work on a 48K Apple II?

And don't forget the most important thing to put in the ad: the retail price and ordering information. Don't include a dealer price in your ads or anywhere on the product itself. That is information that should only be revealed to a legitimate dealer.

As far as design is concerned, whether to save money and use a dot-matrix printer or to spend money for a graphics artist to design an ad is something you must let your budget decide. A slick, four-color full-page ad will definitely attract more attention and thereby get more sales, but it can also bankrupt you before your first items make it to the dealer's shelves. Don't forget to include your business address and phone number, and make sure that the person answering the phone knows something about the program and your pricing policies. If possible, have someone else available who knows the technical aspects of the program.

It's a good idea to prepare two separate letters for people asking for information on the program—one for retail customers, and one for dealers (including dealer prices and discounts). Include technical specifications and figures stating the limits of the program (such as requires 48K, color monitor, paddles, and/or joystick). Customers want to know whether your program will do what they want it to do.

Use high-quality stationery for your mailings, and if you use your computer's word processor, use a good letter-quality printer. Most people realize they are getting a form letter, but a fifth carbon will turn a person off no matter how good your program is. It's worth the money for a good offset printing job. If your ad in a magazine has a "bingo number," you can also use it for mailings. Look into presorted first-class mail rates. With the presorted mail rate and elimination of the envelope (folding the information sheet and putting the address on the outside), almost any company can afford advertising mailings. If your program is designed for the "adult only" market, use an envelope (plain brown wrappers come to mind); you do not want to aggravate the "Post Awful" inspectors. You may even wish to put a disclaimer on the envelope stating that the enclosed materials are designed for the adult audience only and may offend certain people.

Include a warranty card with your product. It is an excellent way to set up a mailing list for future products and to determine proof of purchase. (Do not, under any circumstances, give out the names and addresses on the mailing list. It is tacky, illegal in many states, and a good way to lose respect.) Give a reasonable warranty policy, in writing, specifying length of warranty, what to do if the media is dead on arrival, what to do if the media is dead in warranty, what to do if it goes dead out of warranty, and what to do if the user finds a bug in the program. Set up a return policy with reasonable replacement and upgrade charges. Set up an upgrade policy; that is, what the end user should do to get the upgrade when a new version comes out.

If you must copy-protect your media, make sure you offer backup media at a reasonable price (perhaps free with the package or free upon return of the warranty card). This cannot be overstressed. Many users will not buy copy-protected software without backups. There is no reason to charge more than five dollars for a backup disk. If you charge a large amount (such as $30 or more), the end user will probably resent you enough to make his own backup or obtain a "broken copy" for a backup. Dealers will also be resentful that they are not getting their "cut" when you make your backup sale. Make the copy protection as transparent to the user as possible. (Your adventure generator should make normal user-accessible files that users can read from their own programs, not just yours.) If possible, put in the capability for a user to back up a disk without copying it—for example, a master that creates three slave disks which all work but will not make any more slaves.

Make süre that your program will work on all versions of the computer's operating system, drives, languages, and so forth. Make it possible for the end user to copy the program over to a hard disk drive or, upon request, provide a copy that can be copied over. People who have spent more than $1,000 for hard disk drives are not really interested in giving out copies of your program. If you're really worried, make the program such that, if copied down from the hard drive to a floppy, it no longer works or will work in demo mode only without the program itself working (an excellent way to get prospective users interested in your program). Tell the dealers they are free to give out copies of the demo disk to their customers.

As part of your "dealer support program" (convincing dealers why they should buy yoùr products, as opposed to your competitors'), provide your dealer with a "Lazarus program." When run, this program asks the dealer to insert a dead (either DOA or accidentally erased) user disk. It then determines whether or not the disk is a legitimate copy by checking various sectors for formatting information. If it is a legit copy, it recopies over any dead sectors. Naturally, this program won't work if the disk has been introduced to the "peanut butter syndrome" or has been totally erased. In those cases the user is instructed to return the disk to your headquarters. Having a Lazarus program gives you three advantages:

1) Customers have less time to wait to replace a dead disk.

2) You have fewer headaches replacing dead disks.

3) Dealers get fewer complaints from customers, and better sales support; this indirectly helps sales and encourages people to buy your programs rather than those of company X, which may not have a Lazarus program.

Remember rule number one of copy protection: A copy protection method should work at least as well as the normal operating system!

Only copy-protect what *must* be protected. There is no good reason for copy-protecting an entire disk. If people can copy one track, they can copy the entire disk. If you do protect the entire disk, you will have problems with foreign disk drives, new operating systems, and the like. If your program is too large for one disk, make only the boot disk copy-protected; encourage the user to make backups of the data disks. People tend to refuse to support a company that charges more than a nominal fee (say ten dollars) to replace an uncopyable disk.

Rule number two: Accept the fact that sooner or later someone will find a way around your protection system. Don't panic. Not everyone who owns a microcomputer is out to get you by making copies of your program. A copy protection system should only be used to prevent the casual user from giving out copies to friends. Protection that causes the program to work improperly on end user systems will cause you no end of headaches, and possibly a lawsuit. If someone wants to break into your system badly enough, there are always ways.

Rule number three: When a new disk drive or operating system comes out, make sure your program works with it, and if it doesn't, *fix it*!

Rule number four: The copy protection system used should be efficient. It should produce the media quickly, inexpensively, and reliably. A system that requires three hours and fifty dollars of labor probably won't work too well for games in the mass market.

Rule number five: Test your system thoroughly; remember that not everyone using your system will know as much as you do.

Most users, especially business users (yes, Virginia, business users do play games! A gentleman who works for a large New York banking firm is known to have a set of paddles and a copy of *Alien Typhoon* hidden in his bottom drawer), don't have too much computer knowledge. Many haven't even turned on a computer before. Your manual should do everything but tell the person how to plug in the computer (and in some cases even that). If your program requires data disks, the program should prompt the user and initialize or format the disk. Before formatting a blank disk, try to have your program read the disk to see if it already has data on it (and if it does, print some prompt like "Warning, this disk already has data on it," print a directory, and ask if the user wants to continue). Many users can't even initialize their own disks (sad but true). A copy utility should also be included for the data disks. If possible, make your data disk such that, if a user tries to look at the directory, the directory will indicate that this is being used by your program and that a user shouldn't even try to save programs, or data, onto it. If you can, fill the index of available disk sectors such that even if a user does try to save a program, the operating system will respond with a "DISK FULL ERROR". Assume only that your user can read the manual, but try not to talk down to the reader. An experienced user shouldn't fall asleep reading the manual. You may even wish to produce it in two parts—a beginner's tutorial and a technical information section for advanced users.

When your program is being loaded, have it check itself out, making sure everything has been loaded properly. If possible, include two copies on the disk so that if the primary copy doesn't work, the backup is loaded automatically. If neither copy works, the system should print a message on the screen telling the user what to do.

When you make your production master (the disk that will be copied and sold), do so by copying your object files from your master source disk over to the production disk. Do *not* just make a copy of your master disk and delete your source files (files can easily be undeleted by a competent user). When producing end user disks, the best method usually is to use a 128K RAM card and two disk drives. With a proper generator program, you just load the program once and dump copies out to each drive in turn. Copies can be made as fast as fifteen seconds per copy using this method. Make the generator program as dummy-proof as possible; then you can hire a three-fifty-an-hour employee to make the copies, rather than tie up a programmer. Before making copies, make sure that there is no proprietary data on the production disk. Check with all manufacturers of software that you use within your package for distribution rights. Make sure that you don't put anything into your program that someone else wrote unless you get their written permission.

Send free copies of the program for review as far in advance as possible. Remember that it takes several weeks or even months for an article to get printed. Don't send out an untested original version. You don't want a review stating that you release programs before they are properly tested.

Don't rush an unfinished program out onto the market; similarly, don't advertise a date for release that you can't make in time.

Don't put the "other company" down in your ads. Most people get turned off by this. Wait for the magazines to do a comparison, and let them compare!

Price your program properly. See what similar programs are selling for and price accordingly. If your 3-D graphics package goes for $55 and theirs sells at $39.95, you're probably going to lose sales. Conversely, if theirs goes for $55 and you price yours at $49.95 and they price theirs at $45 and you price yours at $39.95 . . . well, you get the idea.

Support your programs. Have a good manual with the program, and an address (preferably not your home address) and phone number that customers can use if they need help. Yes, folks, there are people out there who want you to tell them how to get by the large green snake that is blocking their way, even if it does mean ruining the game.

When you come out with future programs, make them compatible with older programs but able to run separately also. For example: First you come out with a text editor, then an adventure program generator. If your adventure generator comes with a rudimentary editor but can use your more advanced editor, you'll sell more editors. Make sure that all of your programs use the same keystrokes for similar functions. All you have to do is plan in advance, and each program will help to sell the other ones.

Be original. Broderbund's *Choplifter* was a hot seller because it was the only one of its kind. It was a unique, original game not found in any arcade; fast, action-packed, and most important—fun!

If your program is a utility, it is best to include a sample program full of remarks as well as a tutorial. Suggest extra books or manuals that users can read to understand how to use your system better. If someone wants to write a book about how to use your program, don't discourage them; help them! As long as the book doesn't include your entire manual, and does contain information to help someone learn how your program works, it will give you fewer headaches and help you sell more programs.

Offer the manuals for sale separately from the complete package. Many users, especially educational institutions, will buy the manual separately from the system, read and evaluate it, and then buy the system itself. Decide on a policy for what to do if a person wants to buy a manual, buys the system, and then wants a refund on the manual. And stick by that decision!

Make your program as compatible as possible with all brands of accessories. If you come out with a program to print graphs on a graphics printer, support as many as possible, including options such as color, sheet feeders, and plotters, and using numerous interfacing combinations (parallel eight-bit, serial half duplex, serial TTL, serial 20 mA loop, brainwaves, and all the rest).

Accept the fact that many people are paranoid about modifying their system and making hardware changes. You won't sell as many word processors if you require an XYZ keyboard as you will if you allow an XYZ brand keyboard, an ABC brand keyboard, or a plain vanilla keyboard.

For example: The following Apple graphics utility doesn't exist. Maybe someone would like to write it.

The disk will boot on either a sixteen-sector or thirteen-sector system. It will automatically determine which version it is in and load the correct version of DOS. It will look to see if an eighty-column terminal is attached (either an external terminal or any of a dozen or so different eighty-column cards available), and if so it will use it. A system configuration file (menu-driven, of course) will include this and other information. The menu will ask whether an upper/lower-case character generator is installed, what type of printer is in use, its special features, and the control sequences to get those features. Several popular printers are already predefined and can be loaded. Most plotters are also supported. All major color printers are supported. (When one company was asked why their graphics utility supported more than twenty different black-and-white printers but supported color printers in black-and-white mode only, they said that there wasn't a market for color output.) The user is asked whether input should be from joystick, keyboard, graphics tablet, or a user-written driver. If the user has a 16K RAM card in the system, it will be used for extra memory storage. Several different configuration files can be stored for a user who has several different systems or different printers. The program can be transferred over to a hard disk drive. The master disk will, upon demand, produce three backup disks. Each time it creates a slave, it writes out a pointer on the master. After three slaves have been created, the portion of the master that creates the slaves is destroyed. If a person copies any of the slaves or the master, or tries to download the hard drive copy to a floppy, a demo copy will be created. The demo will be identical to the original but will only store 2K of data—not enough to make it worth stealing, but enough to show the capabilities of the program. The data is stored on normal Apple DOS – compatible disks that a user program can read. (This automatically makes the program compatible with several popular word processors.) Files can also be read/written to disks in Pascal, CP/M, and SOS.

All Apple Computer and most other brands of printer interface cards are supported. If a user wishes, a custom-written assembly language printer driver can be attached, although in most circumstances it isn't necessary. Minimum system requirements are 48K RAM and at least one disk drive. The same disks will work on an Apple II, Apple IIe, Apple III (emulation mode), or Apple-lookalike, although the latter is not supported. (If you support the imitation Apple, you will have problems with schlocky companies, schlocky construction, weird keyboards, and the rest of it. Better leave such users on their own, or let them *goto* the manufacturer.) Optional but highly recommended hardware enhancements for the program include extra memory, printer and interface, hard disk drives, RAM disk drives, eighty-column terminals, a modem, or an external user program.

The warranty period is ninety days. If the program fails within the warranty period, return the disk to the company and a replacement will be sent. Out of warranty, there is a ten-dollar replacement charge. Warranty is proven by a copy of the sales slip and a warranty card on file. If the original disk is dead on arrival, return it to the dealer for replacement. (The dealer will receive a replacement copy when the dead disk is received.) The dealer will also be given a copyable demo and is encouraged to distribute that copy. Upon buying at least ten copies of the program for sale and signing a nonrelease contract in which he/she agrees to make no changes or modifications, the dealer will be provided with a Lazarus program for free. Manuals will be sold separately for $15. They will include a demo copy. The cost of the manual is not refundable upon purchase of the package, and the demo will not be upgraded to a full copy. The suggested retail price is $100. Dealer cost is $60 for five or more copies at a time, with demo copies for $10. If an order is paid in advance, shipment is free; otherwise, C.O.D. Terms are available for qualified customers. The dealer price is $40 for orders of greater than fifty pieces at a time.

Please note that the above description does not say how the program should work. That is the decision of the programmer. Every feature noted above is physically possible and compatible with every other feature (including keeping the program copy-protected and hard disk drive–compatible), but *not one* of the thousands of programs available for the Apple II has even half of the above features.

Not all of the above recommendations will apply in every case (in fact, it would be hard to make a totally original program that is "better" than an existing program, as there would be nothing comparable), but they could help to improve the sales and usability of any program.

Over the last six years, Philip Chien has had occasion to be an end user, a retail dealer, a programmer, and technical manager for United Software of America, and is currently president of Earth News, an OEM firm.

NO TIPPING

Adventurers are invited to respond to the pleas of the stumped listed herein and to solicit aid in return. In addition to this reader-supplied public service, any tip submitted in response to a Typified that the editors judge to be extraordinarily trenchant, to the point, and terrific will be rewarded with a $100 bounty. Include your idea of the straightforward solution along with each masterpiece of literary or cryptographic subtlety. This is not optional.

Try to limit yourself to a maximum of five questions and five tips apiece, or mix and match for a total of no more than ten. We ask this favor of you, gentle reader, because the volume of mail is now such that we couldn't print all requests/assists if No Tipping were twice its current size. Therefore, we strive instead to give something like equal representation to all who write. (A single tip that takes up a full page or more is likely to be neither prized nor printed.)

Remember to state the name of the game and the question to which you are responding.

Certain questions regarding certain adventures have a high rate of recurrence and will be recirculated in the Typifieds from time to time. By all means, keep answering them. The classic adventures are like busy turnstiles. They are always new to new adventurers, and certain areas of high frustration will always produce the same requests. If your tip didn't make it to print or to riches when the question was first asked, hang on to it, rework it, and improve it. The question may come up again.

Final note: Poems win bounties when they have some bearing, in their entirety, on the clue being given—not because they make pretty rhymes.

Address all tips/questions to St. Gametip, Box 60, North Hollywood, CA 91603.

Postmark deadline: April 9, 1984.

March bounty winners:

Adventure—How do I open the north door of the giant room?

And the winner is:

The sunrise puddle
Will solve your muddle.
—Keith Amman, Evanston, IL

Cranston Manor—How do you get rid of the suit of armor?

And the winner is:

You're trying to scare the armor. Think of elephants.
—Charles Follmer, Silsbee, TX

Ultima II—How do you get the items needed for traveling in the rocket/raise character attribute points?

And the winner is:

```
10 GOTO HOTEL
20 IF CLERK THEN GET G: = GOLD - (G*100): ATTRIBUTES =
   ATTRIBUTES + INT (G*RND(1))
30 IF ST> = 30 THEN PLATE = 1: POWER = 1
40 IF PLATE OR POWER THEN GOSUB ROCKET: GOTO SPACE
```

—Theodore Chen, Potomac, MD

TYPIFIEDS

Adventure

q. How do I get across the reservoir?—Shawn Smith, Greendale, WI

Castles of Darkness

q. I have the bow, arrows, and asbestos but can't get past the dragon.—Theodore Chen, Potomac, MD

The Coveted Mirror

q. Where do I find a guide?—Mike Brown, New Carlisle, OH

Cranston Manor

a. *To deactivate the computer:* Think opposite the question, that's the clue; think of liquids and what they do.—James Marcolesco, Villa Park, CA

Deadline

q. Can the second murder be prevented?—Sarah Murdoch, Toronto, Ontario

q. Where is the fragment?—Greg Threadgill, Cantonment, FL

Enchanter

a. *Getting the Kulcad spell from the machine room:* Roger Carlson wants haiku, Roger Carlson can have haiku: Multicolored friend/trudges slowly up the stairs;/escaping answers.—Judith Lucero Turchin, San Francisco, CA

q. How do you get back out of the machine room? What about the nasty door/translucent rooms/silver spoon?—Riva Bickel, Lake Worth, FL

q. Should I use the Kulcad on the guarded door or the stairs? I can kill the dragon and the guy with the axe, but no Krill; I think I need Guncho, but if I get that, then I can't get to Krill!—T.K. Almroth III, Huntington Beach, CA

Escape from Rungistan

q. How do I get out of the pot in the jungle the native keeps putting me in?—Juan Fernando Serra, Miami, FL

Legacy of Llylgamyn

q. How do you get to the island, or otherwise get to levels 4, 5, and 6?—John Hagen, Saint Louis, MO

Mask of the Sun

a. *What to do in the talking statue scene:* A stone jaguar knows the word of truth.—John Kight, Dallas, TX

a. *Getting by the snake:* Illumination means elimination.—Theodore Chen, Potomac, MD

Mystery House

q. How is the trunk in the storeroom opened?—Sharon Weiner, Chicago, IL

Planetfall

q. I've made it all the way through, but I've never gotten past Blather, walked on the catwalks, or explored the reactor access stairs and transportation supply, both dark and grue-some. Can one?—Judith Lucero Turchin, San Francisco, CA

q. How do you dispose of the amoeba/get safely into and out of the radiation room?—Riva Bickel, Lake Work, FL

SAGA #3: Mission Impossible

a. *Getting through the large metal doors:* Don't be too shy to get your picture taken by the apparatus above the table. . . . Find the colored key, unlock the colored button, and soon you will be on your way to showing your authority.—Pat Nash, Enid, OK

a. *Getting the yellow key without blowing the place up:* The saboteur is good for other things besides lying around.—Steve Moody, Houston, TX

SAGA #4: Voodoo Castle

a. *What is meant by "moving bag":* What do oceans and long good-byes have in common?—Matthew Rockloff, Belmont, CA

SAGA #5: The Count

a. *Making it through the night with the stake:* Where the Count hasn't been in a long time is where he isn't going to go now.—Riva Bickel, Lake Worth, FL

SAGA #6: Strange Odyssey

q. How do I get out of the room with crushing gravity?—Jr. Anderson, Glenview, IL

SAGA #7: Mystery Fun House

a. *How to get past the calliope:* Do you need glasses? Look at a mirror! The answer may be the pits, but with some bounce you'll get through, even with a bad turn.—Robbie Bain, Morristown, NJ

SAGA #8: Pyramid of Doom

a. *Killing the cobra:* If you look at the matter first, you will not need to.—Clint Wheeler, Dallas, TX

q. Where do you stash the treasures? What is in the mirror-lined room?—Robbie Bain, Morristown, NJ

Serpent's Star

a. *How to get past the door with six dents in it after the maze*: Gee! Explored Maze, Steele?—Theodore Chen, Potomac, MD

Starcross

q. How do I get into the bubble at the top of the tree?—David Epner, Barberton, OH

Ultima II

a. *After finding the quicksword and receiving blessing from Antos*: The space ahead is sure a lot, yet when you see stars reverse the Beast and follow his path.—George Booras, Miami, FL

q. Where exactly on planet X is Antos?—Charles Follmer, Silsbee, TX

Ultima III

a. *Where to find exotic arms*: One wizard in Dawn must die.—Bill Seligman, Brooklyn, NY

q. Does anyone know in what dungeon one finds the mark of the snake?—David W. Nelson, Lindstrom, MN

q. Where are the four hidden shrines? How do I get them?—Bill Seligman, Brooklyn, NY

Ulysses and the Golden Fleece

q. Where is the potion on the island of storms?—Randy Moen, Gary, MN

Witness

q. I know the clock was rigged, but can't seem to find any other clues . . . I solved *Deadline*. Could it be I'm missing a big clue?—Nancy E. Fong, San Francisco, CA

Wizard and the Princess

q. How do you get past the rickety old bridge?—Barry Spencer, Brighton, MI

Wizardry

a. *Raising character levels without cheating*: First-level solitaire gains the most, and there is always Murphy's ghost. Fight, and be bold; forget the gold.—Kevin Scoles, Minot, ND

q. How do you get past the large sliding wall with the image of the bear? What are the bronze and silver keys for? Where is Rent-A-Stairs Corp?—Mike Parks, Amery, WI

ZORKTALK

First off, let's award one gold-plated kudos to an anonymous phone caller to these offices for his assistance in teching us all an object lesson. The conversation went something like this:

(ring)
"Hello, I have a problem with Zork."
"What kind of problem?"
"Well, I went to save it to disk."
"Yeah?"
"And it saved."
"So?"
"Oh: I left the program disk in the drive."
"And?"
"And so now when I try to run the program, it just spits out call statements."
"Well, on our copy of Zork, it tells you to take the program disk and put the save disk in before you save."
"Oh."
"Do you have the original or a copy?"
"A copy."
"You should have gotten the original."
"Oh."
"Bye."
"Bye."
(click)

Best tip you'll get all week.

March bounty winner:
Zork III—How to get out of the royal puzzle without losing the book. And the winner is (finally!):

The way you came in
Is the exit to choose;
The walls that move provide the clues.
You just have to find
The Right ladder to use.
—Doug Jacobson, Nassau Bay, TX

GENERAL PLEAS AND ASSISTANCE

I

Help wanted. A gentleman known only as Clark (Duluth, MN) is trying to load a lot of stuff on the boat, but he keeps dropping the candles.

Help granted. How to get the slag? "You don't want it," opines Richard Todd (Cocoa, FL). "Think of where it's at, and use some." Some sound songbird advice from Mike Levy (Dallas, TX): "The songbird you seek cannot be caught, but holds a treasure which you have sought. A bauble which shines from end to end can be gained with the songbird's clockwork friend." A tip from Harry Page and company (Sunnyvale, CA): "Write to Infocom for map and hints. They sell for $7.95 and were a great help to us."

II

Help wanted: Mike Levy (Dallas, TX) has freed the demon, given him all the treasures, and gotten the wand, and now he's standing there waving it, trying to move the menhir. Chris Lee (Foster City, CA) was waiting for Keith Amman (Evanston, IL) to ask, but now he just has to know: What is the answer to the riddle in the riddle room? How can Doug Jacobson (Nassau Bay, TX) get by the three-headed dog? And the poetic Bruce Kleinman (Mill Valley, CA) wonders, "How in the hell do I get up the well?"

Help granted: Currently, in the ever-popular Ways To Get Past the Dragon department, one of our favorites is that of Sarah Murdoch (Toronto, Ontario): "You can lead a dragon to water (or something like water) but you can't make him think." Murdoch also notes that James B. O'Neill, in wanting to know what to do with the place mat and how to get into the dreary room, has almost answered his own question: "All you have to do now is push the key." Speaking of keys, if you want to get one from the unicorn, Stephen Jensen (Provo, UT) intones, "The princess doesn't have much to say/but I wouldn't let her get away." To get the portrait out of the bank, Richard Todd (Cocoa, FL) says, "You have to pay for it."

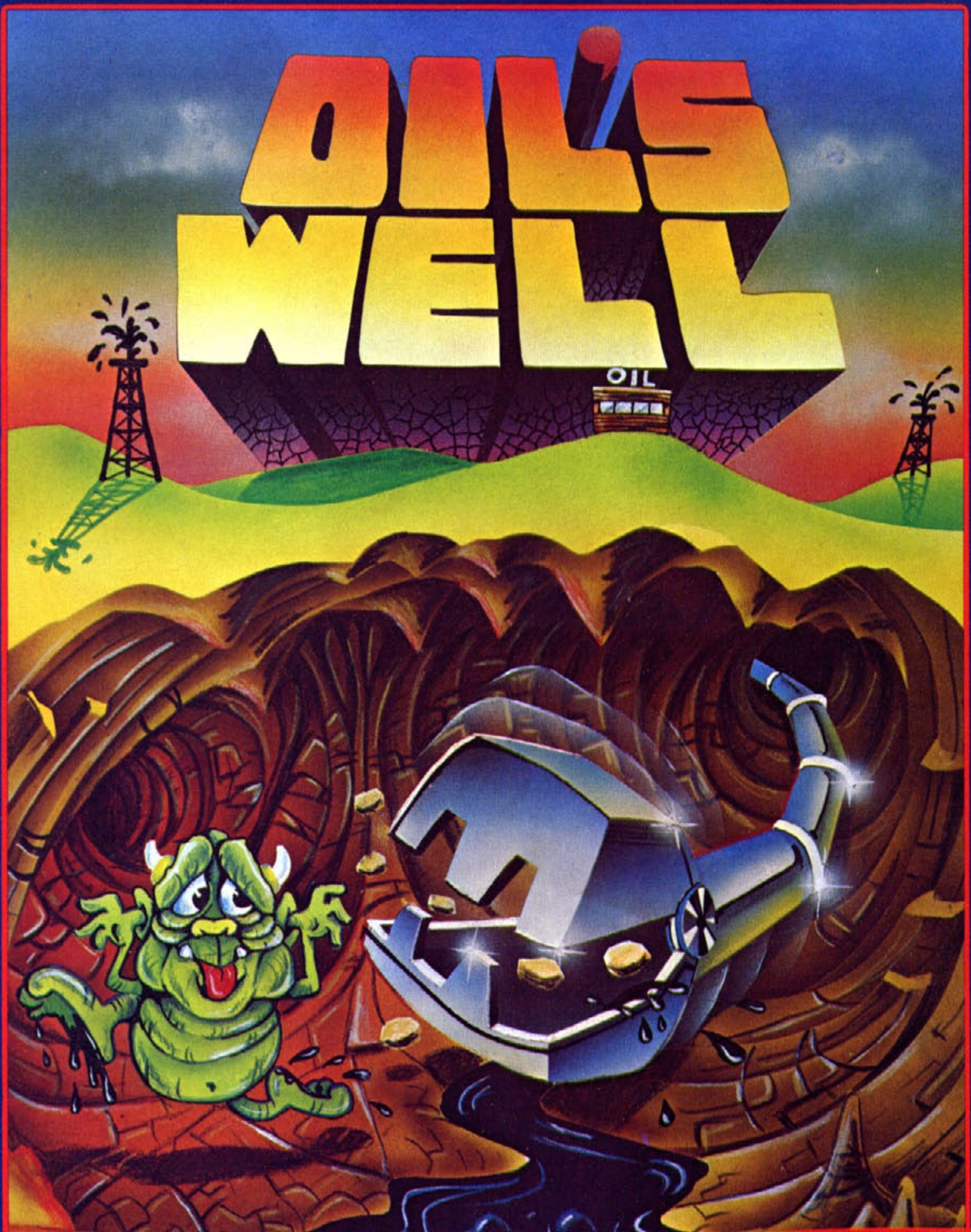
III

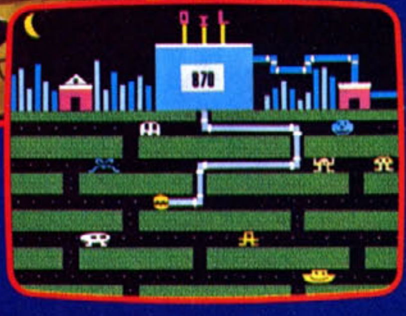
Help wanted: Robbie Bain (Morristown, NJ) speaks for many when he says: "What do you do with the golden amulet, strange key, golden machine, grue repellant, and iron door, not to mention the royal puzzle and heavy vial?" (Partial replies are okay.) Sarah Murdoch (Toronto, Ontario) is stuck in the mirror room. What is the wooden staff used for, Nancy Fong (San Francisco, CA) would like to know. She's gotten to the dungeon master and is stuck.

Help granted: To get a light source to the south shore, Sarah Murdoch (Toronto, Ontario) suggests you circumvent the problem. "You don't need a light source if you travel early. Also, the bugs are ferocious, so take a little something." To get the chest open, Bob Dickson (Foster City, CA) recommends: "Upon first sighting the chest, take it and be patient. The answer over your head? Why not wait and see?"

★ ★ ★ ★ A FOUR STAR WINNER*

*Computers & Electronics Magazine





"This is a high-speed, high-skill game, and certainly one of the better that we've seen recently."

Computers & Electronics Magazine

Sierra ON-LINE

THE ULTIMATE FULL-COLOR NO-RES ADVENTURE#1

ADVENTURES IN ADVENTURING

by Ken Rose

The big Bumblebee Airline jet rocked and rolled in the overcast April sky as it awaited clearance to land at O'Hare Airport. O'Hare being Chicago's answer to the Los Angeles freeways, it was anyone's guess as to when we would finally set down.

Normally I wouldn't mind, as the limousine from the Text Adventure Writers Guild would wait until the flight's arrival to take me to the Lakeshore Drive home of the Guild. But on this particular afternoon, somehow or other I had wound up on the same L.A.-to-Chicago flight with one Latimer Eutheria, a fellow member of the Guild. Eutheria was known to one and all as "Rodent," due to his amazing resemblance to either a mouse or a guinea pig. In fact, there are approximately 2,400 living species of Rodentia and, depending on the light, Latimer could conceivably resemble any one of them.

In any case, he had come wandering back from his first-class seat to the peon section to stretch his legs and had noticed me sitting there rereading Fritz Leiber's Fafhrd and Gray Mouser series.

"Gracious! My, my . . . bless me. If it isn't Ken Rose. I must, simply *must* talk to you . . . simply must."

"Latimer," I replied, "how nice to see you. Look, it's awfully crowded back here—why don't we wait until we get back to Chicago?"

"Simply can't, simply can't, Kenneth—it's very important, you see, very important. And there's all sorts of space up front."

"Latimer, the airline frowns on moving around without appropriate recompense, which I would rather not expend."

Latimer looked at me quizzically.

"Oh, no problem at all. I own twenty or thirty percent of Bumblebee and they are very accommodating. Please, be my guest."

And so, there we were. First class is all right, but first class next to Eutheria makes for a long three-and-a-half-hour ride.

"Rose, I want to thank you for mentioning me and all that in your last article, but I have some serious criticism of the latest effort you have produced."

It is traditional at the Text Adventure Writers Guild to pass one's work around for debugging and comments before releasing it. One tries one's best to see that Eutheria either doesn't get a copy (which he often doesn't, as he travels extensively to visit his Rock Hare holdings) or, if he does, to avoid him if at all possible.

I began pouring down the first-class Scotch as fast as they supplied it, attempting to drown out Eutheria's comments.

It didn't do any good.

"You'll recall that when I told you that there should be a way to write an adventure to run on any machine, you took my comments to heart. In fact, you wrote a splendid example that fit neatly on my Timex."

He continued, "However, this latest effort, which expands on the earlier, is much too complex. Even though my Timex is expanded to 16K, it wouldn't fit.

"I finally borrowed Muffy's C-64 and got it to run there, but I had to change the parser. See what I mean?"

```
LINES 100 TO 190 STAY THE SAME

195 IF MID$(A$,A,1) = " "THEN X = A - 1
200 IF MID$(A$,A,1) = " "THEN A = 0
210 IF A =0 THEN GOTO 250
220 NEXT A
230 V1$ =A$
240 IF V1$ =A$ THEN GOTO 280
250 V1$ =LEFT$(A$,X)
260 IF RIGHT$(A$,LEN(A$) - X) = " " THEN N1$ =""
265 IF RIGHT$(A$,LEN(A$) - X) =THEN GOTO 280
270 N1$ =RIGHT$(A$,LEN(A$) - (X +!))
280 X=0
```

I nodded. "That'll work with most extended Basics, such as Applesoft and IBM and Commodore. If you can't do that, you may need to set up separate strings for V1$ and N1$, as I outlined last issue.

"But Latimer," I groaned, "What's the point?"

"The object is to make the program easy, not complicated. And you have managed to take a grand design and make it difficult," Eutheria squeaked.

"No," I said, "the object is to take Basic programming commands and build a full-fledged adventure. And that's what the 'Full-Color No-Res Adventure' does.

"The program is 23,743 bytes long and should fit on any computer with 24K of usable memory. It can be made considerably shorter by cutting out REM lines and shortening room definitions. And, of course, you can put multiple statements on the same line. To make the game universal, I put each statement on a separate line. You can save memory by combining lines, but watch line numbers.

"The structure of the game is very straightforward."

I handed him a piece of paper out of my briefcase showing the following structure:

LINES	0000–0080	Introduction
	0100–0320	Parser
	0400–1170	General Verb Handling Routines
	1200–1780	Moving Around Routines
	1800–1890	Listing of Objects Routines
	1900–4000	Room Descriptions
	4100–4350	Look, Describe Routines
	4400–4580	Getting Routines
	4600–4740	Dropping Routines
	4800–4880	Inventory Routines
	4900–4980	Quitting Routine
	5000–5170	Saving Game (For Apple)
	5200–5290	Restoring Game (For Apple)
	5300–5440	Assigning Object Numbers
	5500–5710	Object Listing for Rooms
	5800–7440	Start Up Routines
	7500–7770	General Hint Sections
	7900–8020	Turtle Routines
	8100–8380	Puzzle Handling
	8400–8660	Switches and Some Buttons
	8700–8810	Climbing
	8900–8810	Opening
	9100–9390	Inserting
	9400–9850	Turning
	9900–9970	Magic

Eutheria glanced at it momentarily and handed it back.

"No, no, you have precisely missed the point."

At that moment, the jet took an unexpected and particularly nasty bounce, and Eutheria poured his tea in my lap. He didn't notice.

"You see," he said, "after I got it to fit, I couldn't solve it. It was much too complex. It didn't have any colors in it at all, and I couldn't do anything. In fact, I was tempted to (he looked around) cheat.

"After all, what is the object besides carrying around a turtle? I found six rooms that were in last issue's example, found one more, got killed once, and that's all I could do. Is that right for all that typing?"

"Of course not," I said. "It's a very large, deliberately complex adventure to show what can be done with very few programming tools. I did choose to muddy up the waters so you couldn't solve it easily by following the program listing, but you should be able to do that. You should have peeked, or as you put it, cheated a wee bit."

The color drained from Eutheria's face.

His squeak, while remaining the same in volume, rose an octave: "How dare you! Cheat? In a text adventure?! Leave me! Go away! You . . . you tempter . . . you . . . you cheat maker!"

In fact, I was a bit harsh. But insulting him was the only way I could find to exit gracefully. I made a brief apology and returned to Fafhrd and the Gray Mouser.

(Oh, incidentally, dear readers, many of you have inquired about a multiple statement parser, one that allows you to say "Go North. Get Chicken. Throw Chicken Off Cliff" on the same line. That's what we'll look at next time.)

Illustration: Weldon O. Lewis

Graphics: Nancy James

Part 2: Not Just for Display

by Matt Loveless

The Atari screen display is controlled by two dedicated chips: the ANTIC and the TIA. While the TIA maintains a display by sending color/luminance data to the television, the ANTIC's job is to create an interface by retrieving from memory data that determines how the screen will look and translating it into signals that a television will understand. The ANTIC chip allows us to custom-design the display with an overwhelming amount of variation, so fire up the brain cells and read on.

If you look closely at a television display, you'll notice that it is not a smooth and unbroken image, but instead comprises many tightly packed dots which, when examined together, form a complete picture. One of these small dots is called a color clock. Color clocks are grouped into horizontal scan lines that stretch across the screen. There are 262 scan lines from the top of the screen to the bottom, but only about 200 scan lines are visible on a standard television. Since about 60 lines lie out of view of the user, most programs only utilize 192 scan lines. When designing the screen, this fact must be taken into account. Horizontally, the situation is similar. There are 228 color clocks in a single scan line, but only about 160 can be seen. Fortunately, the user is only required to adjust the image vertically; the Atari defaults to 160 color clocks horizontally.

Antic à la Mode. The ANTIC chip derives the fundamental screen image from two sources: the display list and the display memory (both of which can reside virtually anywhere in memory). In general, the display list is composed of a set of instructions that describe to ANTIC the actual architecture of the screen—an outline, if you will—and the display memory is the actual data that will be shown. In other words, the display list is no more than a list that the ANTIC chip references in order to discover how to interpret and display the information in the display memory.

Vertically, the screen display is broken up into various groupings called mode lines. Mode lines are a series of scan lines that are united for one particular display function. For example, one row of a graphics 0 character screen is actually only one mode line, even though it is eight scan lines high. Those eight scan lines are treated as one homogenous unit devoted to displaying graphics 0 type text. Hence, a Basic mode 0 line is represented in the display list by one instruction.

Mode lines come in three types: blank lines, bit-mapped mode lines, and character mode lines. Blank line instructions, as the name implies, instruct the ANTIC to display from one to eight scan lines that retain the background color but contain no data whatsoever.

Bit-mapped modes are sometimes referred to as point-plotting modes, because the data in the display memory addresses individual points within the chosen mode line. These points, which vary in size according to the mode chosen, are commonly referred to as pixels. Note that a pixel is not identical to a color clock; a pixel is the smallest-sized dot that can be displayed in a particular mode. For example, a Basic mode three pixel would be represented by four color clocks horizontally and eight scan lines vertically. There are eight different bit-mapped modes that vary in the number of usable colors and the size of the individual pixels. The third type, character modes, are the most powerful and versatile type of mode line available on the Atari. There are six character display modes accessible from the display list.

Character modes such as Basic modes zero through two differ from bit-mapped modes in that instead of simply displaying a pattern of pixels (squares) according to information in the display memory, they display the whole character. In other words, the display memory does not hold the data that describes what the character looks like; rather, it only tells ANTIC which character to display. The display memory might contain the data to display a row of A's, but would not contain the data describing the actual shape of an A. The actual data for the character is held in the character set, which is redefinable. We will discuss character sets in a future article.

In addition, there are several options that were designed into the various mode line instructions: display list interrupts, load memory scan, vertical scroll enable, and horizontal scroll enable. These further enhance the power of the display list by allowing certain options to be used selectively on desired mode lines. The only option of any importance to us at this point is the load memory scan. The load memory scan or LMS of a mode line instruction tells ANTIC that the next two bytes point to the memory address that marks the beginning of the display memory.

Finally, there is the jump instruction, which appears in two forms. The first form is the straight jump or JMP. The JMP instruction is analogous to the Basic *goto* statement. It instructs ANTIC to continue the display list at a different location in memory. The new address should follow the JMP instruction (low, high). The second type of JMP instruction, which is more common, is the JVB (jump to vertical blank). Essentially, the JVB tells ANTIC that it has reached the end of the display list and feeds ANTIC the address of the beginning of the display list. The JVB instruction is also followed by an address (low, high). All display lists should end with a JVB.

The first step in creating a custom display list is to design, on paper, exactly how you would like the screen to appear. In order to do this, you must reference table 1 to decide what you want and how many scan lines it will take. You should keep a running total of the number of scan lines used so that you do not exceed the maximum. If you use too many scan lines, the screen will begin to roll as if your vertical hold were maladjusted. Every display list should begin with three blank-eight-line instructions ($70). This brings the display down twenty-four lines into the viewable portion of the screen. After that, you have 192 scan lines to play with without the worry of losing data off the bottom.

The three blank-eight-line instructions should be followed by the first mode line with the LMS option invoked. As we mentioned, the LMS is used to tell ANTIC where the display memory resides. The LMS must be followed by the address of the display memory. The low byte of the address comes first, followed by the high byte (for additional explanation, see the program). Then come any additional mode lines you desire. Note, however, that it is not necessary to use all 192 available lines. In fact, the fewer mode lines used, the faster your Atari will run.

Now let's step through the creation of a display list. This sample display list will be rather simple. It will consist of one of everything. The first step is to put down the three blank-eight-line instructions. By looking at the table, we see the instruction to blank eight lines is a $70, so the first three entries in our display list will be $70s. Second, we want ANTIC mode two, which is the first mode line, with the LMS option invoked. We take the two and add $40 to it for the LMS, which gives us $42. Next, we need to put the low and high bytes of the display memory into the display list. Since we don't have any data, let's just point it somewhere in memory where we know there is a lot of arbitrary data. $E400 is a good choice. The low byte will be $00 and the high byte will be $E4. Next, we need modes three, four, and so on until mode $0F, the last mode. Notice that we only used 89 of the 192 usable scan lines. Finally, we need the JVB (jump to vertical blank) instruction, which tells ANTIC to stop displaying and shows where the beginning of the display list can be found. Since in the example program the display list is on page 6 ($600, decimal 1536), the low byte will be $00 and the high byte $06. Table 2 shows the completed display list.

The next step is to incorporate the new display list into a program. After creating the display list, you must first decide where to place the display list in memory. Since display lists are stationary and seldom consist of more than 200 bytes, they are usually assembled directly into the program. There is one constraint concerning the location of the display list, however; it must not cross a 1K boundary—that is, any two-byte address ending in $00. If it is absolutely essential for a display list to cross one of these boundaries, you may do so with a JMP instruction, but this is not recommended.

Deciding where to place the display memory is the next task. This is a bit trickier. First of all, a mixture of mode lines might require a strange arrangement of memory, because the amount of memory needed per mode line varies greatly. Second, the display memory must not cross a 4K boundary (an address divisible by $1000). But if the situation does arise, you may use a second LMS instruction to bridge a 4K boundary in order to remedy the problem.

Once the memory situation has been resolved, only one step remains: informing ANTIC as to the whereabouts of your display list. The first step is to place ANTIC on hold while you modify its display list pointers. This is done by storing a zero in the register called SDMCTL at location $22F. Now store the low byte of your display list into SDLSTL ($230) and store the high byte into SDLSTH ($231). Finally, turn ANTIC back on by storing a $22 in $22F. Now our new display list should be in action.

The example program uses the sample display list from table 2. The run address is $620, and the source code is in MAC/65 format. To run with Atari's ASM/ED, make two line changes:

```
480 LDA #DLIST&$FF
500 LDA #DLIST/256
```

The comments should make the program self-explanatory. In the next issue, we'll take a look at using the display memory with custom display lists. Until then, happy assembling.

ANTIC MODE # (HEX)	BASIC MODE #	TYPE	# OF COLORS	COLOR CLOCKS	SCAN LINES	PIXELS/ LINE	BYTES/ LINE
2	0	CHAR	2½	4	8	40	40
3	—	CHAR	2½	4	10	40	40
4	—	CHAR	5	4	8	40	40
5	—	CHAR	5	4	16	40	40
6	1	CHAR	5	8	8	20	20
7	2	CHAR	5	8	16	20	20
8	3	BIT	4	4	8	40	10
9	4	BIT	2	2	4	80	10
A	5	BIT	4	2	4	80	20
B	6	BIT	2	1	2	160	20
C	—	BIT	2	1	1	160	20
D	7	BIT	4	1	2	160	40
E	—	BIT	4	1	1	160	40
F	8	BIT	2½	½	1	320	40

Note: The ANTIC mode numbers and the Basic mode numbers are not identical.

Options:	ADD TO ANTIC MODE #
DLI	$80
LMS	$40
VERT. SCROLL	$20
HORIZ. SCROLL	$10

SPECIAL	BLANK 1 LINE - $00	BLANK 7 - $60
	BLANK 2 LINE - $10	BLANK 8 - $70
	BLANK 3 LINE - $20	
	BLANK 4 LINE - $30	JMP - $01
	BLANK 5 LINE - $40	JVB - $41
	BLANK 6 LINE - $50	

Table 1.

```
0100 ; ------------------------------------------------------------
0110 ; DISPLAY LIST EXAMPLE
0120 ;   MAR/APR ST. GAME
0130 ;     By Matt Loveless
0140 ; ------------------------------------------------------------
0150 ;
0160     .OPT OBJ    ;generate object code
0170 ;
0180 ;<<<< SYSTEM EQUATES >>>>
0190 ;
0200 SDMCTL - $022F    ;DMA control ($00 - screen off; $22 - normal) [OS]
0210 SDLSTL - $0230    ;display list pointer (low byte) [OS]
0220 SDLSTH - $0231    ;display list pointer (high byte)[OS]
0230 ;
0240 ;<<<< PROGRAM EQUATES >>>>
0250 ;
0260 DMAON - $22       ;value to store in SDMCTL for norm. display
0270 DISMEM - $E400    ;arbitrary sample display memory address (see text)
0280 ;
0290 ;
0300 ;<<< DISPLAY LIST >>>
0310 ;
0320       * - $0600        ;start at beginning of page six for convenience
0330 ;
0340 DLIST .BYTE $70,$70,$70 ;three blank-eight-line instructions
0350       .BYTE $42       ;mode two with LMS option
0360       .WORD DISMEM ;follow LMS with display memory address
0370       .BYTE $03,$04,$05,$06,$07,$08,$09,$0A,$0B,$0C,$0D,$0E,$0F
           ;one of everything
0380       .BYTE $41       ;JVB (jump to vertical blank) signify end
0390       .WORD DLIST ;point ANTIC back to the top
0400 ;
0410 ;
0420 ;<<<< MAIN PROGRAM >>>>
0430 ;
0440       * - $0620        ;start program a little bit from display list
0450 ;
0460 START LDA #$00      ;turn the screen off
0470       STA SDMCTL
0480       LDA # <DLIST  ;set display list pointer (low byte)
0490       STA SDLSTL
0500       LDA # >DLIST  ;(high byte)
0510       STA SDLSTH
0520       LDA #DMAON    ;turn screen back on
0530       STA SDMCTL
0540       RTS           ;return
```

SAMPLE DISPLAY LIST

DISPLAY LIST ADDR - $0600
DISPLAY MEMBER ADDR - $E400

VALUE	SCAN LINES NEEDED	FUNCTION
$70	8	BLANK 8 LINES
$70	8	BLANK 8 LINES
$70	8	BLANK 8 LINES
$42	8	ANTIC MODE 2 WITH LMS
$00	—	DISPLAY MEMORY ADDR (LOW BYTE)
$E4	—	(HIGH BYTE)
$03	10	ANTIC MODE 3
$04	8	ANTIC MODE 4
$05	16	ANTIC MODE 5
$06	8	ANTIC MODE 6
$07	16	ANTIC MODE 7
$08	8	ANTIC MODE 8
$09	4	ANTIC MODE 9
$0A	4	ANTIC MODE $A
$0B	2	ANTIC MODE $B
$0C	1	ANTIC MODE $C
$0D	2	ANTIC MODE $D
$0E	1	ANTIC MODE $E
$0F	1	ANTIC MODE $F
$41	—	JVB
$00	—	DISPLAY LIST ADDR (LOW)
$06	—	(HIGH)

Table 2.

True Confessions. We could do that. We could use Lisa. But it would be *wrong*.

And it was. We cheated. You see, it's easier to develop software on the Apple than on the Commodore, mainly because the Apple drives are much faster and more reliable. And, as your genial host has his Apple and Commodore connected together via a parallel port, it's easy to load stuff back and forth. Thus, those Commodore assembly listings have so far been created with an Apple assembler.

Starting with our next issue, Commodore listings will be in the proper MAE assembler format.

Red Pencil Time. And now, on to business. How many of you did your homework? What!? Only three? Out of how many millions? Unbelievable. And it's for your own good. Well, forget it then. No article this month, until everyone else does their homework.

Excuse me a moment.

> WRITER: What's up, oh noble ED?
> EDITOR: THOU HAST AN ARTICLE TO DOETH. SO DOETH IT!! OR FACETH THE PENALTY!!
> WRITER: (gasp) You mean— ?
> EDITOR: YES! THOU SHALT BE CAST DOWN INTO COMPUTER WRITER'S HELL, NE'ER MORE TO BE SEEN IN THE HALLOWED HALLS OF *ST.GAME*.
> WRITER: (panicked) No, no! I'll do it!

For the benefit of those three who did work on last issue's problems, and for the one hundred thirty-five thousand who were going to but never got around to it, let us unfold the veil and examine the untold mysteries of time, space, and game programming.

Check, Please. Take a look at listing 1; this is code for the XMOVE routine discussed last issue. A short explanation is in order.

```
XMOVE    inc  CYCLE
         lda  CYCLE
         and  #$07
         sta  CYCLE
         lda  XOLD
         cmp  XNEW
         beq  NOXMOVE
         jsr  XMAGFNC
         sta  DELTA
         ldx  #$02
XLP1     ldy  #$00
XLP2     lda  DIX,Y
         cmp  LINEX,X
         bcs  XSKP
         jsr  MOVXLINE
XSKP     lda  POWER2 + 1,Y
         sec
         sbc  #$01
         and  CYCLE
         bne  XOUT
         iny
         cpy  #$04
         bne  XLP2
XOUT     dex
         bpl  XPLP1
NOXMOVE  rts
;
;
POWER2   byt  $01,$02,$04,$08
         byt  $10,$20,$40,$80
;
;
MOVXLINE jsr  CLRXLINE
         clc
         lda  LINEX,X
         adc  DELTA
         sta  LINEX,X
         jsr  CHECKX
         jmp  PLTXLINE
```

Listing 1.

First, the

```
INC CYCLE
LDA CYCLE
AND #$07
STA CYCLE
```

sequence does the

```
CYCLE** = CYCLE** + 1
IF CYCLE > 7 THEN CYCLE = 0
```

shtick. See? We're only interested in the low-order three bits (0–7), and 8 == 0 if the high-order bits don't count (the == means "equivalent to"). The next five instructions should be obvious. Study your 6502 books and read the Tarot if you're still confused.

For the loops: The NUM loop will go backward, to save a compare, and the DEPTH loop will go forward because it can't go backward (for a very good reason). The X register is the NUM loop, and the Y register is the DEPTH loop. Since we are keeping loop counters in registers, we must remember to save the register contents when we go to plot a line or something. The lines

```
LDA DIS,Y
CMP LINEX,X
```

are backward from the way we stated them last time because we can save a branch that way; *do plot on LINEX(NUM) > DIS(DEPTH)* is the same as *don't plot on DIS(DEPTH) > = LINEX(NUM)*, which is a BCS (the other would need a BEQ and a BCC).

Now we're at the LDA POWER2 + 1,Y sequence. What we're after here is to compute CYCLE mod 2∧(DEPTH + 1). So we do LDA POWER2 + 1,Y to get 2∧(DEPTH + 1), then subtract 1 and AND with CYCLE to do the MOD. Then we branch if these bits we're checking (that's what we're after) are not all zero. If they are all zero, then we can keep going by incrementing, comparing, and branching. If not, we skip to do the next line. The reason we do a MOD 2∧(DEPTH + 1), instead of MOD 2∧(DEPTH), is that we're checking for the possibility of moving the next lines; we've already moved this line (if necessary).

MOVXLINE should be obvious in function (ignore the JSR CHECKX for now), and that brings us to the end of XMOVE. So, what about YMOVE? Well, it's pretty much the same, as witness listing 2, except that we have no CYCLE and thus no DEPTH loop to worry about. So, what about the hard stuff?

```
YMOVE    lda  YOLD
         cmp  YNEW
         beq  NOYMOVE
         jsr  YMAGFNC
         sta  DELTA
         ldx  #$05
YLP      jsr  MOVYLINE
         dex
         bpl  YLP
NOYMOVE  rts
;
;
MOVYLINE jsr  CLRYLINE
         clc
         lda  LINEY,X
         adc  DELTA
         sta  LINEY,X
         jsr  CHECKY
         jmp  PLTYLINE
```

Listing 2.

GOTO page 26

Robot: Trici Venola

Photo: Tim Durr

BRIAN FITZGERALD

ANATOMY OF A GAME

PART 4

WHAT FOOLS BE THESE MORTALS?

The Biggest Names in Software Want To See Your Stuff

Once there was a race of happy, clever, and nimble-fingered people called Gamers who lived in a large valley. It was called Happy Valley, because it rang with happy shouts and glad cries all the day and night. The Gamers were happy because they got to play their favorite games all the time.

In the north valley, there was much zapping and beeping, and all the Gamers who lived there always had their joysticks with them.

In the south valley, there were a lot of caves and old castles with dungeons that the Gamers were constantly exploring, hoping to accumulate charisma, strength, wisdom, and other good things.

The Gamers in the west valley would go adventuring whenever anyone shouted their rallying cry, "Hi Res!" They liked to push buttons, put things inside other things, take things, examine them, drop them, and apply their clever brains to figuring out what to do next.

The Gamers of the east valley liked to do much the same thing, but less graphically.

And in the central valley, the Gamers would sit motionless for days on end, contemplating their grids and thinking up strategies that would gain them victory in possible future conflicts or that would have done so in past conflicts.

The gods of gaming looked down on all this from their high mountain, and they smiled, for it was good for business. They were glad the Gamers so enjoyed the bountiful fruits of computing that the gods had provided. They were much impressed by the cleverness and stamina of the Gamers, and thought that perhaps there were some among them who could qualify for demigodhood.

They decided to hold a contest among all the Gamers to discover who among them might possess the powers of creation that would mark them as good game-god material, and to this end they posted a decree in each of the five parts of Happy Valley. This is what it said:

Gamers! This is your chance to design, write, and program an original game. You can compete in any of five categories. Should your program win in its category, you shall receive $500. All category winners will be eligible for publishing contracts with software publishers. The grand-prize winner, chosen from the five category winners, will receive special goodies and favors above and beyond those mentioned.

Photo Comp: Trici Venola

Flash! Judging company Datasoft has contributed an extra $2000 for the grandprize winner! Other prizes will be announced as the contest progresses!

The Judges

Arcade

Doug Carlston, Broderbund
Sue Currier, SoftSync
Elliot Dahan, Creative Software
Trip Hawkins, Electronic Arts
Fernando Herrera, First Star
Jerry Jewell, Sirius
Patrick Ketchum, Datasoft
Mark Pelczarski, Penguin
Ihor Wolosenko, Synapse

Adventure (graphic)

Scott Adams, Adventure International
Mark Pelczarski, Penguin
Ron Unrath, Phoenix
Ken Williams, Roberta Williams, Sierra On-Line
Programming Staff, Ultrasoft

Adventure (text)

Michael Berlyn, Marc Blank, Infocom
Doug Carlston, Broderbund
Ken Rose, Omega

Fantasy Role-Playing

Richard Garriott, Origin Systems
Trip Hawkins, Electronic Arts
Michael Katz, Epyx
Robert Sirotek, Sir-tech

Strategy, War Games

Joel Billings, Strategic Simulations
Jackson Dott, Avalon Hill

Miscellaneous

Joel Billings, Strategic Simulations
Sue Currier, SoftSync
Elliot Dahan, Creative Software
Jerry Jewell, Sirius
Trip Hawkins, Electronic Arts
Mark Pelczarski, Penguin
Ken Williams, Sierra On-Line

The Rules:

1. Design and program an original game for your computer in one of these five categories: Adventure, Arcade, Role-Playing Fantasy, Strategy, or Miscellaneous.

2. The program must be totally original; games based on or resembling already existing programs will not be accepted.

3. The program must be a finished product, with title page and all parts working.

4. Programs with bugs that interfere with playing will be disqualified. Make sure your game is thoroughly debugged.

5. Programs must be submitted on a 5¼-inch floppy disk (or disks, if necessary), using the normal operating system for your computer. It may be programmed in any language that will run on your computer, with no modifications other than those widely available.

6. Entries in the Arcade and Fantasy categories must include hi-res graphics if they are available on your computer. Adventures may contain graphics or be text-only. Entries in the miscellaneous category must include miscellany.

7. Programs will be judged by a board consisting of software publishers and programmers from the top companies in the world of computer gaming. Judging will be based on originality of concept, game development, playability, and execution of program.

8. Anyone who has had any kind of program published is not eligible to enter.

9. Deadline is February 1, 1985. Programs must arrive on disk with documentation at *St.Game* no later than this date. Send your entry with your name and address, your computer brand and the configuration of your system to:

St.Game-a-Thon
Box 60
North Hollywood, CA 91603

WE HAVE SPOKEN!

Jock Root

Dungeon Fun

Level 4

A Place to Rest Your Weary Head

Illustration: Trici Venola

Joe is about to graduate—or, as we dungeon people say, move up a level. The program where he lives has been growing nicely these last few months, but it's a long way from full-grown. With the material in this issue, our dungeon machine program is going to graduate into a game system (more than one program) and is going to learn to behave like a young adult.

In computer programming, it's considered kid stuff or hacker's work when you have to end a program by turning it off; "professional" programs are supposed to be more graceful than that. However, when you're first developing a program, you're more concerned with what comes in between than you are with the beginning or the ending.

We're past that point now; we have quite a lot of middle in our program. The dungeon machine can simulate a complex of rooms, with secret compartments in the walls and other "special effects," and it can provide Joe, our wandering hero, with several nasty opponents. It's time to start thinking about refinements—like giving Joe a way to get out of there before he gets killed.

That means, of course, that we will have to provide a safe place for Joe to escape to and rest in. And that means we need to create another imaginary universe (What? You mean one wasn't enough?), since the activities that go on in this resting place will be completely different from what happens in the dungeon area. For example, this is where Joe gets stored on disk, so he can keep any treasures he found (in the earlier version, he lost everything at the end of each trip, and he *hated* that!). In a future version, this will also be the place where he buys new equipment before a trip, and where he finds traveling companions.

In other words, we need a whole new program. This program will take care of all the activities that go on when you're not in the dungeon itself—creating and equipping new characters, gathering a party for an expedition, even (very far future) selecting which dungeon (or whatever) you want to visit. But that's for later. For now, it will be enough if we can teach Joe to jump from one universe to another without getting mangled in the process.

Teleporting between Worlds. When the Apple or the Atari loads a new program from disk, it erases certain areas in memory to give the new program a clear space to work in. One of the areas that gets erased is the space where variables are stored. This protects the new program from being confused by the old program's variables. In most cases this is a good thing, but it creates a problem when you want to transfer variables from one program to another. If we want to transfer Joe (who is really only a set of variables, like Automan) from one program to another, we have to find a special way to do it.

We will take advantage of one of the loopholes in the erasing process mentioned above. When the computer loads a new program, it erases certain parts of memory, but it also preserves certain areas since it needs the information stored there in order to run the new program (the areas containing DOS and the Basic interpreter, for example). One of the protected areas is an "experimenter's workspace" of about two hundred bytes, starting at address $300 in the Apple, $400 in the Atari. If we poke Joe's values into that space, they remain intact.

It works like this: When you are getting ready to call the other program, transporting Joe from one universe to another in game terms, simply poke the variables you want to transfer into the protected area. Then call the new program and have it peek those same variables. From that information, the new program can reconstruct Joe just as he was. You can use this technique to transfer information between any two programs; just make sure that the two agree on what information is to be stored in which memory locations.

We will call this protected area the "shuttle," since it can travel safely between worlds. In the Apple, it begins at the hexadecimal address $300 (that's 768 in decimal, which is the form we'll need it in for peeks and pokes) and extends for 230 bytes. In the Atari, the shuttle area begins at $600.

Joe in the Shuttle. We will have to establish a data format for the shuttle so that both programs will know where to find the information they need. In a later version of these programs, we will store the whole party here during an expedition, but for the moment we want to keep things very simple. At this point, all we know about Joe is his name, how much gold he has, and whether he's found the magic amulet, so that's the information we will put in the shuttle.

The first problem we run into is in storing Joe's name. You can't store a letter of the alphabet, as such, in a memory location; you can only store a number from 0 to 255. However, that's an easy one: We simply convert the letter into an ASCII value (a number between 0 and 255) and store that. Then we instruct the receiving program to interpret those values as ASCII characters (that is, letters) when it reads them.

That sounds simple enough, but it points up something else we will have to do—namely, breaking the shuttle area into separate sections, or fields, so we can tell the receiving program to interpret one field as ASCII characters and the other fields as numbers. We'll designate five locations in the shuttle area, those with decimal addresses 770 through 774, as the name field.

Next, we want to record the amount of gold Joe has. We can store any number up to 255 in a memory location, so we can just record Joe's gold as a number. But what if he has more than 255 in gold? Just in case, we had better use two successive locations. The first will hold the integer value of Joe's gold divided by 256, and the second will hold the remainder from that division. To put that in Basic, we simply poke 775, Int(gold/256) and then poke 776, gold – 256 * Int(gold/256).

To untangle that and reconstitute Joe's gold, the receiving program makes gold = 256 * peek (775) + peek (776). That way we can store Joe's gold up to the limit of 65,536 gold pieces, which should be enough.

Finally, we need to record whether Joe has found the magic amulet. Within the dungeon program, this is indicated by the variable T1. If it is anything but zero, he has found the amulet. Thus, all we have to do is poke 777, T1. The receiving program can then reverse the process by setting T1 = peek (777).

Is Anybody There? You may be wondering what is going on in memory locations 768 and 769. We said that the shuttle area begins at 768, but we started the name field at 770. That was partly because 770 is an easier number to remember than 768, but we also have an important use for those first two locations.

When our program reads the shuttle area to reconstitute a character, it must have a way of telling whether the data in the shuttle is valid or not. It must be able to determine whether the information is garbage left over from some unrelated program, or a character stored there to be recovered. Otherwise, you might find yourself taking some very peculiar characters into the dungeon.

We solve this problem by reserving the first two spaces in the shuttle for an identification code. If this code is present, the program will assume that the rest of the data in the shuttle is correct; if the code is not there, the program will ignore the shuttle data and load Joe from disk.

The numbers we have used for a recognition code are 25 at address 768 and 250 at 769. These are easy to remember and very unlikely to occur by accident.

Joe's resting place—now very primitive, but someday to become the meeting place for adventurers from many worlds and cultures—is the Home Base program. This program starts the expedition by recalling Joe from disk (which loads him into the shuttle) and sending him on his way (by calling the dungeon program). When he has had enough adventuring, he simply walks out through the secret exit, which updates the values in the shuttle and calls the Home Base program again. Thus our dungeon machine becomes a professional-type program system; it now has a proper beginning and ending.

The Home Base program (Programs in Pulp, page 34) starts by jumping out to line 5000 to pick up some variables and print a title, then returns to line 30 for its first serious decision. This is where it checks the identity code to determine whether the shuttle area contains valid data. If the numbers aren't right, it jumps to 1000 and reloads Joe from disk. Someday, this part of the program will offer you a choice; at the moment Joe is the only character we have, though, so we've written him in.

If the identity code checks, the shuttle data is good, so (line 40) the program jumps to line 2000 to interpret it. Lines 2010 to 2070 pick up the name, and line 2050 gets rid of any trailing spaces (in case the name is less than five letters). Then line 2110 picks up the amount of gold your character is carrying, and line 2120 tells whether he or she has found the magic amulet. Lines 2200–2250 report this information, and line 2260 sends you back to the main menu at line 100. Someday, this will be where you choose to create a character, form a party for an expedition, or set out on an adventure, but for now it's the place where your character can decide whether to descend into the dungeon or knock off for a while.

If the character decides to rest, the shuttle data is recorded on disk (line 230) to be ready for next time. If the character wants to go adventuring, line 260 runs the dungeon program.

There is also a separate section of the program, starting at line 10000, which creates a file named Joe on the disk. This stores the recognition code in the shuttle area (line 10020), adds the name "Joe" (with two trailing spaces, to make five characters in all) in lines 10050 to 10090, Joe's gold (we'll start him with only ten) in lines 10100 and 10110, and the magic amulet (he doesn't have it yet) in line 10120. Finally, line 10130 stores the whole thing on disk.

You only need to use this section once, when you start a new disk. Once the disk record has been set up, the other programs will maintain and update it, but there has to be a disk file there for them to work with. To create the file, load the Home Base program and type "goto 1000"—the disk drive will whir for a moment, and Joe's file will appear on the disk.

This Isn't Joe! But suppose you've been following this series for some time, and have created your own hero—let's say her name is Diana. Can the new program deal with that?

Of course it can, or we wouldn't have brought it up. The original dungeon program has Joe's name written into it in several places, and it will be your responsibility to change those (but you have presumably done that already, for Diana's sake). The Home Base program can handle any name of five characters or less; we chose that limit because the dungeon program only provides space for five-character names in its display.

The first part of the Debriefing section (line 2010 and after) reads the name that's stored in the disk file you asked for, as described earlier. From then on, the program will address your character by his or her proper name$.

There are a couple of things you will have to change: first, the name of the disk file the program asks for. This program was written with only Joe in mind, so that's the only disk file it knows how to ask for. You will have to write Diana (or anyone else) into lines 1020 and 1040. Second, if you want to convert the section starting at line 10000 to fit your character, change the letters between the quotes in lines 10050 to 10090, and the name in line 10130.

The Dungeon Universe. Now that we've got a resting place for Joe (or whoever) to escape to, we will have to go back to the dungeon program and install that secret exit we mentioned. That will be easy enough—if you have the program. The complete listing was last published in the January–February *Softline*. If you can't get one locally (some computer stores carry back issues), send three dollars to our back issues department with a request.

Now that you have the listing in front of you, make the following changes: first, line 4730. It's a REM statement that is saving a place for a "special event" hook. We described special events last issue; they happen when the character steps on a certain square, examines a certain wall, or the like. We are now ready to install the hook that the REM was saving a place for, so change that line to

```
4730 IF SQ%(VS,HS,3) = 0 AND D = 3 THEN 21000
```

Note to faithful followers of this series: If you've already used the two hooks we left space for (lines 4730 and 4740) to create special events of your own, it will be your responsibility to find room for this one. That's the price you pay for freedom. . . .

Next, line 5030: this is in the section where the program introduces Joe and describes him. If Joe is on disk, we don't need that part; we've met him already. In fact, if someone is on disk, it might not even be Joe! We'll put in a test here to see if anyone is in the shuttle; if so, we'll bypass the introduction. Change the line to

```
5030 IF PEEK (768) = 25 AND PEEK (769) = 250 THEN WM = 1: GOTO 5100
```

Then add "5040 HOME" to replace the *Home* command in the original 5030.

The variable Wm tells the program whether there is anybody in the shuttle: If WM = 0, the shuttle is not loaded properly, so the program uses the built-in version of Joe. Wm is short for warm, as in warm-start (a computer term that means, approximately, "use what you have, don't start over"). We use this variable to select whether to read someone in from the shuttle, in the lines following 5500 (see listing 1). You will have to substitute the lines in this listing for lines 5510–5590 of the old version.

```
5510 GOSUB 1220: REM  PAUSE
5520 GOSUB 9100: REM COMBAT CHT
5530 PRINT I$"    (ANY KEY TO CONTINUE)": PRINT
5540 HTAB 20: GET Q$: PRINT

5550 REM  ** WHO IS IT? **
5555 IF NOT WM THEN 5600
5560 NAME$ - "": FOR X - 0 TO 4
5565 Q$ - CHR$ ( PEEK (770 + X))
5570 IF Q$ - " " THEN 5580
5575 NAME$ - NAME$ + Q$
5580 NEXT X
5585 GOLD - 256 * PEEK (775) + PEEK (776)
5590 IF PEEK (777) THEN T1 - 1
```

Listing 1.

Finally, add the lines in listing 2 to the end of the old program. This is the secret exit pointed to by the hook you installed at line 4730. It loads Joe into the shuttle (lines 21250–21270) and sends him home (line 21280).

Now your hero has two worlds to play in. Sure, one of them is so primitive that it hardly counts, but now that we have it we can start adding to it. Next issue, for example, we will probably add an armory, where Joe (or Diana) can buy weapons and equipment. And for the more distant future, an advanced version of the dungeon machine will support powerful spells and exotic monsters. Yes, the business of creating imaginary worlds is positively addictive—

To be continued. . . .

```
21000 REM  + + + TO THE EGRESS + + +

21020 REM  **WHAT YOU SEE **
21030 PRINT I$"THE WALL SEEMS TO BE COATED"
21040 PRINT I$" WITH A SHIMMERING, TRANSPARENT"
21050 PRINT I$" FILM, WITH NO SOLIDITY TO IT."
21060 PRINT
21070 GOSUB 1210

21100 REM  ** WHAT YOU FEEL **
21110 PRINT I$"TOUCHING IT, YOU CAN FEEL NO"
21120 PRINT I$" SOLID SURFACE, ONLY A STRANGE"
21130 PRINT I$" HUMMING VIBRATION IN YOUR"
21140 PRINT I1" FINGERTIPS. IT FEELS AS IF"
21150 PRINT I$" YOU COULD WALK RIGHT THROUGH"
21160 PRINT I$" THE WALL."
21170 PRINT : GOSUB 1220

21200 REM  ** WHAT YOU DO **
21210 PRINT I$"WANT TO TRY IT?  (Y/N)  ";
21220 GET Q$: PRINT Q$: PRINT
21230 IF Q$ < > "Y" THEN 4500
21240 PRINT I$"HERE WE GO!"
21250 GH - INT (GOLD / 256): POKE 775,GH
21270 POKE 777,T1
21280 PRINT CHR$(4)"RUN HOME BASE"
```

Listing 2.

Atari in Chains/Bill Beyermann

The Atari uses page six ($600) instead of page three, so the addresses for the currently used part of the shuttle area are 1536 to 1545 (Apple 768 to 777). See Programs in Pulp, page 34.

To store Joe or some other character on the disk, we add the name to FILE$ (line 230), which was initialized to contain the D: required for disk operations (line 5015). Lines 234 to 238 then put the information to the disk byte by byte. Lines 1040 through 1049 read Joe back in. The input is terminated by trapping for an end-of-file error, and line 1045 checks to see if it was terminated for the proper reason. The trap is turned off in 1049 (by trapping to a nonexistent line number) so that if an error should occur somewhere else in the program, a jump won't be made to line 1045.

To read from the shuttle:

```
1400 REM ** ADVANCE VIEWPOINT **
4730 IF SQ(12*VS + 4*HS + 3) - 0 AND D - 3 THEN 21000
4740 REM (MATCH & SEND #4)
5015 OPEN #1,4,0,"K:":DIM AZ$(356),BL$(37),I$(3),Q$(35),WHO$(4)
5020 I$ - "   ":REM 3 SPACES
5025 POKE 752,1:BL$ - "     ":REM 752 TURNS THE CURSOR OFF, BL$ IS
     37 SPACES
5030 ?"↖":POSITION 2,4:REM ↖ IS SHIFT CLEAR
5040 IF PEEK(1536) - 25 AND PEEK(1537) - 250 THEN WM - 1:GOTO 5100
5208 DIM NAME$(5),PROF$(1),Z$(1),E$(18),P$(21),ST$(38)
5210 NAME$ - "JOE":PROF$ - "F"
5530 PRINT I$;"  (ANY KEY TO CONTINUE)":PRINT
5540 POKE 752,0:POSITION 18,23:PRINT " ";:GET #1,Q:POKE 752,1:PRINT
5550 REM ** WHO IS IT? **
5555 IF  NOT WM THEN 5600
5560 NAME$ - "":FOR X - 1 TO 5
5565 Q$ - CHR$(PEEK(1537 + X))
5570 IF Q$ - " " THEN 5580
5575 NAME$(X,X) - Q$
5580 NEXT X
5585 GOLD - 256*PEEK(1543) + PEEK(1544)
5590 IF PEEK(1545) THEN T1 - 1
5625 REM DELETE 5625
```

Listing 1.

By the way, the room descriptions come up rather slowly (especially for the large room) because they are stored in a single large string of 356 bytes. It takes Basic a while to search through all those characters looking for the delimiters. In the future we may add a machine language routine to do the search faster, or we may do a little redesigning. The redesign could require more variables (with a space limit of 128 variables), or we could make the storage area for each description the same size, which will use up more memory. Expansion cartridges, anyone?

Add to end of Atari Dungeon 3:

```
21000 REM + + + TO THE EGRESS + + +
21005 REM CUT
21020 REM ** WHAT YOU SEE **
21030 PRINT I$;"THE WALL SEEMS TO BE COATED"
21040 PRINT I$;" WITH A SHIMMERING, TRANSPARENT"
21050 PRINT I$;" FILM, WITH NO SOLIDITY TO IT."
21060 PRINT
21070 GOSUB 1210
21095 REM CUT
21100 REM ** WHAT YOU FEEL **
21110 PRINT I$;"TOUCHING IT, YOU FEEL NO"
21120 PRINT I$;" SOLID SURFACE, ONLY A STRANGE"
21130 PRINT I$;" HUMMING VIBRATION IN YOUR"
21140 PRINT I$;" FINGERTIPS. IT FEELS AS IF"
21150 PRINT I$;" YOU COULD WALK RIGHT THROUGH"
21160 PRINT I$;" THE WALL."
21170 PRINT : GOSUB 1220
21195 REM CUT
21200 REM ** WHAT YOU DO **
21210 PRINT I$;"WANT TO TRY IT?  (Y/N)  ";
21220 POKE 752,0:GET #1,Q:POKE 752,1:PRINT " ";:Q$ - CHR$(Q):PRINT
      Q$:PRINT
21230 IF Q$< >"Y" THEN 4500
21240 PRINT I$;"HERE WE GO!"
21250 GH - INT(GOLD/256):POKE 1543,GH
21260 POKE 1544,GOLD - 256*GH
21270 POKE 1545,T1
21290 RUN "D:HOMEBASE"
```

Listing 2.

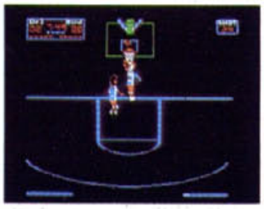
SOFTWARE ARTISTS?

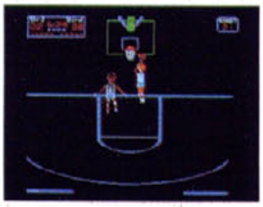
How we got this year's hottest sports game out of two rather inexperienced designers.

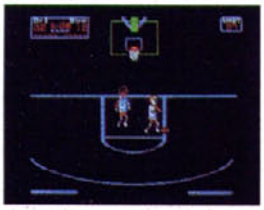
TO MAKE THE FIRST BASKETBALL PROGRAM that feels like the real thing, it helps to start with two guys who know what the real thing feels like.

Enter Larry Bird and Julius Erving. Bird — the hustler, the strong man, deadly from outside. Erving — The Doctor, maybe the most explosive player in the history of the game.

We talked to them, photographed them in action, studied their moves and their stats and their styles. Then we set out to create on computer disc an event which may never happen in real life. We put the two of them together on a dream court of light, for an electronic afternoon of one-on-one.





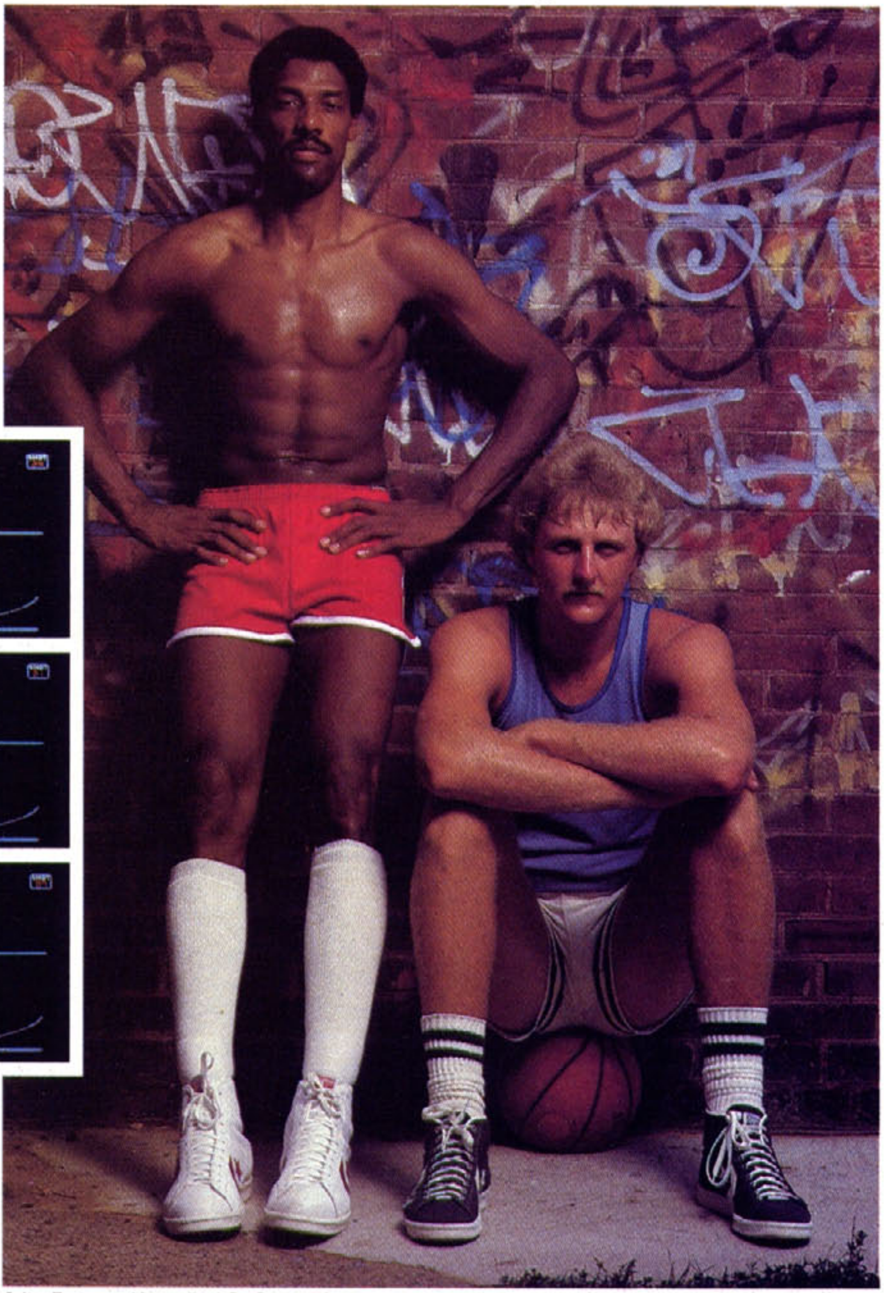


It wasn't easy. When they talked, we listened. When they criticized, we made big changes. When they gave suggestions, we took them.

And it shows. This thing is absolutely uncanny. You actually take on all the skills and characteristics of Bird or The Doctor — their own particular moves, shooting abilities, even strength and speed.

You'll meet with fatigue factors, hot and cold streaks, turnaround jump shots, and 360-degree slam dunks. But there's some whimsy in here, too — a funny referee, a shattering backboard, even instant replay.

It's called *Julius Erving and Larry Bird Go One-on-One.*™ You're Bird. Or you're The Doctor. And that's the last decision you'll have plenty of time to make.



Julius Erving and Larry Bird Go One-on-One is now available on diskette for Apple II, II+, and IIe computers. Apple is a registered trademark of Apple Computer. To find out more about Electronic Arts and its products, write us at 2755 Campus Drive, San Mateo, CA 94403 or call (415) 571-7171. **For a free catalog, send a stamped, self addressed #10 envelope.**

NOW AVAILABLE FOR THE COMMODORE-64

ELECTRONIC ARTS™

continued from page 18

Well, what is CHECKX? As we said last time, it's pretty simple. Just do bounds checking on the top and bottom; if the LINEX value is greater than BOTTOM OF SCREEN, set LINEX to HORIZON + 1. And, if the LINEX value is less than HORIZON, set LINEX to BOTTOM − 1. See listing 3.

CHECKY is just a little harder (listing 4). If the line reaches some terminal point CENTER + TERM or CENTER − TERM (we have symmetry here), we kill it and add it on the other side, slightly advanced; if at CENTER + TERM, we put it at CENTER − TERM + 1; and if at CENTER − TERM, it goes at CENTER + TERM − 1 (the hard part is coming up).

At this point, if reading the Tarot isn't helping, try I Ching. It's more binary.

```
CHECKX   cmp  YSCRNMAX
         bcs  MAX.X
         cmp  YSCRNMIN
         beq  MIN.X
         bcs  NOCHKX
MIN.X    lda  YSCRNMAX
         sec
         sbc  #$01
         sta  LINEX,X
         rts
MAX.X    lda  YSCRNMIN
         clc
         adc  #$01
         sta  LINEX,X
         rts
```

Listing 3.

```
CHECKY   cmp  LTERM
         beq  LFTY
         bcc  LFTY
         cmp  RTERM
         bcc  NOCHKY
RTY      lda  LTERM
         clc
         adc  #$01
         sta  LINEY,X
         rts
LFTY     lda  RTERM
         sec
         sbc  #$01
         sta  LINEY,X
NOCHKY   rts
```

Listing 4.

You've Got To Sink Before You Can Swim. Now, I know that some of you are saying, "But what about the real stuff, like CLRXLINE/PLTXLINE and CLRYLINE/PLTYLINE? Isn't this guy ever going to get anywhere?" Well, we've all heard the saying about how you can't see the forest for the trees. That's a real problem when attacking a large game. The low-level routines are the backbone of the game, and are where a large part of the effort is expended for speed and efficiency, but they have to connect to the rest of the game; and, thus, the game has to be defined first. So, since we've defined some of the high-level stuff and can at least see what we need, let's attack line erasing/plotting.

First, the easy stuff: CLRXLINE and PLTXLINE (see listing 5). As you can tell, these are little more than calls to LINE, with proper parameters set up. Of course, LINE isn't fully defined yet (next issue), but let it slide for now. Tricky stuff explained: Notice that BYT $2C instruction. That creates BIT absolute op-code; what that does is to hide the next instruction from CLRXLINE, since LDA #$01 is two bytes long, and a BIT absolute is three bytes long (in other words, when the entry point is CLRXLINE, the 6502 thinks that LDA #$01 is just the operand to the BIT instruction, so it doesn't execute it). So, when we call CLRXLINE, we get

```
LDA #$00
BIT $01A9
STA LINECOL
```

which stores #$00 in LINECOL. When we call PLTXLINE, we get

```
LDA #$01
STA LINECOL
```

We save a byte at the cost of a cycle, but one microsecond won't kill us (not here, but there are places). And one more note: Even though we called the XLINE line table LINEX, we are storing Y coordinates. Since we know these are horizontal lines (for now), we know the X coordinates, so why store them? And the converse is true for the YLINE table LINEY.

Now, CLRYLINE/PLTYLINE. The information we have on the line is its starting X and Y value—the Y value is fixed (the horizon line), and the X coordinate is from the LINEY table. Now, the line we want to draw can be determined from that position (and by Z coordinate, yaw, pitch, and roll; but let's not go in over our heads for now, eh?) by one of those magical functions. Let's think about it. The closer to the center the line is, the more vertical it is; the further from the center, the more horizontal it gets. It's not a straight linear function; we need to do clipping, and there aren't that many positions (140, to be exact). What does that suggest? A table! We'll compute that next time; for now, let's do the routine (listing 6). As you can see, it's pretty straightforward; the table hides the hairy stuff (that's what tables are for). We get starting coordinates, then use BX as an index into two tables to get EX and EY, and draw that line. Simple, elegant, fast, but with a problem relating to the other four degrees of freedom. That's for next time.

```
CLRXLINE  lda  #$00
          byt  $2C
PLTXLINE  lda  #$01
 ;
          sta  LINECOL
          stx  PLTXTMP
          sty  PLTYTMP
          lda  LINEX,X
          sta  BY
          sta  EY
          lda  XSCRNMIN
          sta  BX
          lda  XSCRNMAX
          sta  EX
          jsr  LINE
          ldx  PLTXTMP
          ldy  PLTYTMP
          rts
```

Listing 5.

```
CLRYLINE  lda  #$00
          hex  2C
PLTYLINE  lda  #$01
 ;
          sta  LINECOL
          stx  PLTXTMP
          sty  PLTYTMP
          lda  LINEY,X
          sta  BX
          tay
          lda  PYENDX,Y
          sta  EX
          lda  HORIZY
          sta  BY
          lda  PYENDY,Y
          sta  EY
          jsr  LINE
          ldx  PLTXTMP
          ldy  PLTYTMP
          rts
```

Listing 6.

What's next? Well, what about those degrees of freedom? And, more important, when will something work? Next issue we'll wrap up the simple stuff and present a working X,Y movement program with joystick interface so you can zoom over the surface of our world. And with that, a *real* line plotter, fast and not elegant. But hey, you usually pay heavy for elegance. And there is much to be said for something that works (a phrase borrowed from FNE of Digital Acoustics).

So, your assignment, Mr. Phelps, should you decide to accept it, and even if you don't, is to find the problems in our approach when extended to include all six degrees of freedom. And next time, we won't be so lenient on the slackers.

So, as they say in Chermany, ourfeetarestained. Oh reservoir. Chow. Speaking of food . . .

Converting LISA listings to other assemblers.

Lisa	SC-Assembler	Marlin/Big Mac	Apple Toolkit Assembler
#label	#label	#<label	#<label
/label	/label	#>label	#>label
EPZ	.EQ	EQU	EQU
EQU	.EQ	EQU	EQU
ORG	.OR	ORG	ORG
OBJ	.TA	OBJ	OBJ
END	.EN	END	END
LST	.LIST ON	LST ON	LST ON
NLS	.LIST OFF	LST OFF	LST OFF
ASC	.AS	ASC	ASC
ADR	.DA	DA	DW
HEX	.HS	HEX	DFB
DFS	.BS	RMB	DS

MARCH-APRIL 1984

ST.GAME

PROGRAMS IN PULP

Special Throwaway Section for Nonprogrammers

ADVENTURES IN ADVENTURING

```
10 CALL -936: REM CLEARS THE SCREEN
20 PRINT "ULTIMATE FULL COLOR NO-RES ADVENTURE #1"
30 PRINT "                    BY"
40 PRINT "                 KEN ROSE"
50 FOR A - 1 TO 1000
60 NEXT A
70 CALL -936
80 GOTO 5800
100 REM     PARSER STARTS HERE
110 Y - 0
120 X - 0
130 V1$ - ""
140 N1$ - ""
150 PRINT
160 PRINT "WHAT NOW? ";
170 INPUT A$
180 IF LEN(A$) - 0 THEN GOTO 100
190 FOR A - 1 TO LEN(A$)
200 IF A$(A.A) - " " THEN V1$ - A$(1.A - 1)
210 IF A$(A.A) - " " THEN X - 1
220 IF A$(A.A) - " " THEN X - 1
230 IF X - 1 THEN N1$ - A$(A + 1. LEN(A$))
240 IF X - 1 THEN A - 0
250 IF X - 1 THEN GOTO 280
260 NEXT A
270 V1$ - A$
280 X - 0
290 REM     NEXT TWO LINES GET RID OF DIRECTION PREFIX
300 IF V1$ - XD$ THEN GOTO 400
310 IF N1$ - W1$ OR N1$ - W2$ OR N1$ - W3$ OR N1$ - W4$ OR
    N1$ - W5$ OR N1$ - W6$ OR N1$ - W7$ OR N1$ - W8$ OR
    N1$ - W9$ THEN V1$ - N1$
320 IF N1$ - WA$ OR N1$ - WB$ OR N1$ - WC$ THEN V1$ - N1$
400 REM     VERB HANDLING ROUTINES
410 REM     NEXT TWO LINES CHECK FOR MOVEMENT
420 IF V1$ - W1$ OR V1$ - W2$ OR V1$ - W3$ OR V1$ - W4$ OR
    V1$ - W5$ OR V1$ - W6$ OR V1$ - W7$ OR V1$ - W8$ THEN
    GOTO 1200
430 IF V1$ - W9$ OR V1$ - WA$ OR V1$ - WB$ OR V1$ - WC$ THEN
    GOTO 1200
440 REM     LOOKING AND READING
450 IF V1$ - WN$ OR V1$ - WO$ OR V1$ - WR$ THEN PRINT
460 IF V1$ - WN$ OR V1$ - WO$ OR V1$ - WR$ THEN GOTO 4100
470 REM     GETTING THINGS
480 IF V1$ - WJ$ OR V1$ - WK$ THEN PRINT
490 IF V1$ - WJ$ OR V1$ - WK$ THEN GOTO 4400
500 REM     DROPPING THINGS
510 IF V1$ - WL$ THEN GOTO 4600
520 REM     INVENTORY ROUTINES
530 IF V1$ - WG$ OR V1$ - WH$ OR V1$ - WI$ THEN GOTO 4800
540 REM     QUITTING
550 IF V1$ - WE$ THEN GOTO 4900
560 REM     SAVING GAME
570 IF V1$ - WD$ THEN GOTO 5000
580 REM     CLIMBING
590 IF V1$ - WT$ AND R - 6 THEN R - 7
600 IF V1$ - WT$ AND R - 7 THEN GOTO 1200
610 IF V1$ - WT$ THEN GOTO 8700
620 REM     SAYING THINGS
630 IF N1$ - WW$ AND R - 7 THEN GOTO 7600
640 IF V1$ - WW$ AND R - 7 THEN GOTO 7600
650 IF V1$ - WX$ AND R - 7 THEN GOTO 7670
660 IF N1$ - WX$ AND R - 7 THEN GOTO 7670
670 REM     LONG/SHORT DESCRIPTIONS
680 IF V1$ - WY$ THEN PRINT
690 IF V1$ - WZ$ THEN PRINT
700 IF V1$ - WY$ THEN WO - 1
710 IF V1$ - WZ$ THEN WO - 0
720 IF V1$ - WY$ THEN PRINT "LONG DESCRIPTIONS WILL ALWAYS
    BE SHOWN."
730 IF V1$ - WZ$ THEN PRINT "SHORT DESCRIPTIONS WILL ALWAYS
    BE SHOWN.";
740 IF V1$ - WY$ THEN GOTO 100
750 REM     EATING
760 IF V1$ - X3$ THEN GOTO 7800
770 REM     PUZZLE SOLVING
780 IF V1$ - X7$ AND N1$ - X2$ THEN GOTO 8180
790 REM     PUSHING AND THROWING
800 IF V1$ - XD$ AND R - 29 THEN GOTO 8410
810 IF V1$ - XF$ AND R - 30 THEN GOTO 8570
820 IF V1$ - XF$ AND R - 19 THEN GOTO 8910
830 IF V1$ - XF$ AND N1$ - XG$ AND R - 36 THEN GOTO 9900
840 IF V1$ - XD$ OR V1$ - XF$ THEN PRINT
850 IF V1$ - XD$ OR V1$ - XF$ THEN PRINT "GRUNT . . . NOTHING
    SEEMS TO HAPPEN!"
860 IF V1$ - XD$ OR V1$ - XF$ THEN GOTO 100
870 REM     JUMPING
880 IF WV$ - V1$ THEN PRINT
890 IF WV$ - V1$ AND R - 7 THEN PRINT "YIIIIII...!!!"
900 IF WV$ - V1$ AND R - 7 THEN PRINT
910 IF WV$ - V1$ AND R - 7 THEN GOTO 4960
920 IF WV$ - V1$ AND R - 22 THEN PRINT "THERE'S SOMETHING UP
    THERE BUT I CAN'T QUITE MAKE IT OUT."
930 IF WV$ - V1$ AND R - 22 THEN GOTO 100
940 IF WV$ - V1$ THEN PRINT "THERE'S NO REASON TO DO
    THAT HERE."
950 IF WV$ - V1$ THEN GOTO 100
960 REM     OPENING THINGS
970 IF V1$ - XH$ AND R - 19 THEN GOTO 8910
980 REM     INSERTING THINGS
990 IF V1$ - XK$ THEN GOTO 9100
1000 REM     TURNING THINGS
1010 IF V1$ - XJ$ THEN GOTO 9400
1020 REM     SMACKING THINGS
1030 IF WT - 0 THEN GOTO 1120
1040 IF R - 10 AND V1$ - XO$ AND WU$ - N1$ AND WT - 1 THEN
    OB(1) - 91
1050 IF R - 10 AND V1$ - XO$ AND WU$ - N1$ AND WT - 1 THEN WT - 0
1060 IF R - 10 AND V1$ - XO$ AND WU$ - N1$ AND WT - 0 THEN
    CALL -936
```

```
1070 IF R = 10 AND V1$ = XO$ AND WU$ = N1$ AND WT = 0 THEN PRINT
     "POOF! POOF!"
1080 IF R = 10 AND V1$ = XO$ AND WU$ = N1$ AND WT = 0 THEN FOR
     A = 1 TO 75
1090 IF R = 10 AND V1$ = XO$ AND WU$ = N1$ AND WT = 0 THEN
     NEXT A
1100 IF R = 10 AND V1$ = XO$ AND WU$ = N1$ AND WT = 0 THEN R = 7
1110 IF R = 7 AND V1$ = XO$ AND WU$ = N1$ AND WT = 0 THEN
     GOTO 1320
1120 IF V1$ = XO$ THEN PRINT
1130 IF V1$ = XO$ THEN PRINT "YUCK!"
1140 IF V1$ = XO$ THEN GOTO 100
1150 PRINT
1160 PRINT "I DON'T UNDERSTAND"
1170 GOTO 100
1200 REM     MOVING AROUND ROUTINES
1210 X = R
1220 IF V1$ = W5$ AND R = 3 THEN GOTO 7510
1230 IF V1$ = WB$ AND R = 3 THEN GOTO 7510
1240 IF V1$ = W6$ AND R = 7 THEN GOTO 7550
1250 IF V1$ = WC$ AND R = 7 THEN GOTO 7550
1260 IF V1$ = W1$ OR V1$ = W7$ THEN R = N(R)
1270 IF V1$ = W2$ OR V1$ = W8$ THEN R = S(R)
1280 IF V1$ = W3$ OR V1$ = W9$ THEN R = E(R)
1290 IF V1$ = W4$ OR V1$ = WA$ THEN R = W(R)
1300 IF V1$ = W5$ OR V1$ = WB$ THEN R = U(R)
1310 IF V1$ = W6$ OR V1$ = WC$ THEN R = D(R)
1320 IF R>0 THEN X = R
1330 IF R>0 THEN CALL -936
1340 IF R>0 THEN GOTO 1410
1350 IF R = 0 THEN Y = 1
1360 IF Y = 1 THEN PRINT
1370 IF Y = 1 THEN PRINT "YOU CAN'T MOVE THAT WAY . . . NOW!!!"
1380 IF Y = 1 THEN R = X
1390 IF Y = 1 THEN X = 0
1400 IF Y = 1 THEN GOTO 100
1410 IF WO = 1 THEN RD(R) = 0: REM     RESET LONG DESCRIPTION
1420 IF R = 1 THEN GOSUB 1910
1430 IF R = 2 THEN GOSUB 1970
1440 IF R = 3 THEN GOSUB 2030
1450 IF R = 4 THEN GOSUB 2110
1460 IF R = 5 THEN GOSUB 2190
1470 IF R = 6 THEN GOSUB 2250
1480 IF R = 7 THEN GOSUB 2310
1490 IF R = 8 THEN GOSUB 2390
1500 IF R = 9 THEN GOSUB 2450
1510 IF R = 10 THEN GOSUB 2510
1520 IF R = 11 THEN GOSUB 2550
1530 IF R = 12 THEN GOSUB 2570
1540 IF R = 13 THEN GOSUB 2590
1550 IF R = 14 THEN GOSUB 2650
1560 IF R = 15 THEN GOSUB 2710
1570 IF R = 16 THEN GOSUB 2770
1580 IF R = 17 THEN GOSUB 2790
1590 IF R = 18 THEN GOSUB 2850
1600 IF R = 19 THEN GOSUB 2970
1610 IF R = 20 THEN GOSUB 3090
1620 IF R = 21 THEN GOSUB 2850
1630 IF R = 22 THEN GOSUB 3170
1640 IF R = 23 THEN GOSUB 3110
1650 IF R = 24 THEN GOSUB 3210
1660 IF R = 25 THEN GOSUB 3270
1670 IF R = 26 THEN GOSUB 3330
1680 IF R = 27 THEN GOSUB 3350
1690 IF R = 28 THEN GOSUB 3410
1700 IF R = 29 THEN GOSUB 3480
1710 IF R = 30 THEN GOSUB 3570
1720 IF R = 31 THEN GOSUB 3670
1730 IF R = 32 THEN GOSUB 3690
1740 IF R = 33 THEN GOSUB 3760
1750 IF R = 34 THEN GOSUB 3780
1760 IF R = 35 THEN GOSUB 3800
1770 IF R = 36 THEN GOSUB 3840
1780 IF R = 37 THEN GOSUB 3930
1800 REM     LISTING OF OBJECTS ROUTINE
1810 PRINT
1820 PRINT "OBJECTS HERE ARE: "
1830 PRINT
```

```
1840 X = R
1850 GOSUB 5500
1860 X = 0
1870 IF R = 10 AND PP > 1 THEN GOTO 8000
1880 GOTO 100
1890 REM     END OF LISTING OBJECTS ROUTINE
1900 REM     ROOM DESCRIPTIONS
1910 PRINT "YOU ARE AT THE BASE OF HIGH CLIFF"
1920 IF RD(R) = 1 THEN RETURN
1930 PRINT
1940 PRINT "THERE IS A SIGN CARVED IN THE CLIFF."
1950 RD(R) = 1
1960 RETURN
1970 PRINT "YOU ARE AT THE BASE OF A CLIFF"
1980 IF RD(R) = 1 THEN RETURN
1990 PRINT
2000 PRINT "THERE IS A WEB HERE THAT WEIGHS A TON. IT IS A
     CLIFF-TON WEB."
2010 RD(R) = 1
2020 RETURN
2030 PRINT "YOU ARE OVERLOOKING A BAY"
2040 IF RD(R) = 1 THEN RETURN
2050 PRINT
2060 PRINT "YOU ARE HIGH ABOVE A COVE, PERCHED ON A VERY
     STEEP LEDGE. SMALL YELLOW-COLORED"
2070 PRINT "RIVULETS (THE COLORING A TRICK OF THE LIGHT,
     PERHAPS?) TUMBLE INTO A LARGE"
2080 PRINT "BAY. FAR IN THE DISTANCE, YOU CAN SEE A SEQUENCE
     OF CAYS CURVE GENTLY TOWARDS THE LEFT."
2090 RD(R) = 1
2100 RETURN
2110 PRINT "YOU ARE AT THE BASE OF A CLIFF"
2120 PRINT
2130 PRINT "THE ROCKS ABOVE ARE PRECARIOUSLY BALANCED."
2140 PRINT
2150 IF R = 4 AND CL = 1 THEN GOTO 7730
2160 PRINT "I WOULD TIPTOE OUT AND NOT RETURN"
2170 CL = 1
2180 RETURN
2190 PRINT "YOU ARE AT THE BASE OF A CLIFF"
2200 IF RD(R) = 1 THEN RETURN
2210 PRINT
2220 PRINT "AT ONE TIME THEY THREW A BALL HERE. IT WAS A
     BASE-BALL."
2230 RD(R) = 1
2240 RETURN
2250 PRINT "YOU ARE AT THE BASE OF A CLIFF"
2260 IF RD(R) = 1 THEN RETURN
2270 PRINT
2280 PRINT "YOU MIGHT BE ABLE TO START CLIMBING HERE."
2290 RD(R) = 1
2300 RETURN
2310 PRINT "YOU ARE IN THE EAGLE'S NEST"
2320 PRINT
2330 PRINT "YOU ARE VERY PRECARIOUSLY PERCHED IN THE NEST.
     YOU SEE BITS OF FLOTSAM AND JETSAM LITTERING THE NEST."
2340 PRINT
2350 PRINT "YOU ALSO SEE THE BONES OF EARLIER ADVENTURERS
     LITTERING THE NEST."
2360 PRINT "ALL IN ALL IT IS A MESSY NEST."
2370 PRINT "AND TO TOP IT ALL OFF, AN EAGLE IS CIRCLING
     OVERHEAD."
2380 GOTO 1800
2390 PRINT "YOU ARE AT THE YELLOW BRICK CROSSROADS"
2400 IF RD(R) = 1 THEN RETURN
2410 PRINT
2420 PRINT "THE ROAD WAS MADE BY MUNCHKINS FROM OZ
     LEFTOVERS. IT WAVERS AND WANDERS IN ALL DIRECTIONS.
     NORTH LOOKS PROMISING."
2430 RD(R) = 1
2440 RETURN
2450 PRINT "YOU ARE STANDING ON YELLOW HUMMOCK #1"
2460 IF RD(R) = 1 THEN RETURN
2470 PRINT
2480 PRINT "THE GROUND SHAKES AND SHIVERS BENEATH YOUR
     FEET."
2490 RD(R) = 1
2500 RETURN
```

```
2510 PRINT "YOU ARE WADING THROUGH YELLOW MIRE"
2520 PRINT
2530 PRINT "YELLOW MIRE IS VERY STICKY. IT SHOULD ALSO NEVER,
     BUT NEVER, BE EATEN."
2540 RETURN
2550 PRINT "YOU ARE STANDING ON YELLOW HUMMOCK #2"
2560 RETURN
2570 PRINT "YOU ARE STANDING ON YELLOW HUMMOCK #3"
2580 RETURN
2590 PRINT "YOU ARE ON A YELLOW BRICK ROAD"
2600 IF RD(R) = 1 THEN RETURN
2610 PRINT
2620 PRINT "YOU ARE NOT OFF TO SEE THE WIZARD. HE DOESN'T
     WANT TO SEE YOU AND BESIDES, HE ISN'T IN THIS
     ADVENTURE."
2630 RD(R) = 1
2640 RETURN
2650 PRINT "YOU ARE WADING THROUGH YELLOW MUCK"
2660 IF RD(R) = 1 THEN RETURN
2670 PRINT
2680 PRINT "THERE IS A YELLOW BRICK ROAD TO THE EAST, NORTH,
     AND SOUTH OF YOU."
2690 RD(R) = 1
2700 RETURN
2710 PRINT "YOU ARE ON A YELLOW BRICK ROAD"
2720 IF RD(R) = 1 THEN RETURN
2730 PRINT
2740 PRINT "YOU HEAR MUNCHKINS SOFTLY SINGING IN THE
     DISTANCE, A LOUD MUNCHING SOUND . . . THEN SILENCE!"
2750 RD(R) = 1
2760 RETURN
2770 PRINT "YOU ARE ON A YELLOW BRICK ROAD"
2780 RETURN
2790 PRINT "YELLOW BRICK FORK IN THE ROAD"
2800 IF RD(R) = 1 THEN RETURN
2810 PRINT
2820 PRINT  "A ROAD SIGN IS HERE WITH ARROWS POINTING IN ALL
     DIRECTIONS. THE DIRECTIONS ARE FADED AND UNREADABLE."
2830 RD(R) = 1
2840 RETURN
2850 PRINT "YOU ARE IN A YELLOW GROVE"
2860 PRINT
2870 PRINT "THERE IS A HUGE YELLOW TREE HERE."
2880 IF RD(R) = 1 THEN RETURN
2890 PRINT
2900 PRINT "YOU HEAR A YELLOW BIRD HIGH UP IN THE TREE. . .
     SINGING. THERE IS A LOUD MUNCHING SOUND, THEN SILENCE."
2910 RD(R) = 1
2920 RETURN
2930 PRINT "YOU ARE IN A PURPLE GREENHOUSE"
2940 PRINT
2950 PRINT "THERE IS A SHELF ABOVE YOUR HEAD."
2960 RETURN
2970 PRINT "YOU ARE STANDING ON YELLOW HUMMOCK #4"
2980 PRINT
2990 IF RD(R) = 1 THEN GOTO 3030
3000 PRINT "I AM VERY FRIGHTENED AND WOULD ADVISE YOU NOT
     TO MOVE IN ANY DIRECTION."
3010 PRINT "IF I COULD ONLY REMEMBER HOW TO GET OUT OF
     HERE, BUT I FORGOT."
3020 PRINT
3030 RD(R) = 1
3040 IF HB = 0 THEN PRINT "OH. . . THERE IS A CLOSED METAL BOX
     BUILT INTO THE WALL WHICH SEEMS TO BE DOING THE
     HUMMING."
3050 IF HB = 1 THEN PRINT "THERE IS A METAL BOX BUILT INTO THE
     WALL."
3060 IF BO = 1 THEN PRINT "THE BOX IS OPEN EXPOSING A KEYHOLE."
3070 IF OB(8) = 90 THEN PRINT "A KEY IS IN THE KEYHOLE."
3080 RETURN
3090 PRINT "YOU ARE ON THE YELLOW BRICK ROAD"
3100 RETURN
3110 PRINT "YOU ARE ON THE PURPLE BRICK ROAD"
3120 IF RD(R) = 1 THEN RETURN
3130 PRINT
3140 PRINT "THEY RAN OUT OF YELLOW BRICKS HERE AND TO THE
     EAST."
3150 RD(R) = 1
3160 RETURN
3170 PRINT "YOU ARE IN A PURPLE GREENHOUSE."
3180 PRINT
3190 PRINT "THERE IS A SHELF ABOVE YOUR HEAD."
3200 RETURN
3210 PRINT "YOU ARE ON THE PURPLE BRICK ROAD"
3220 IF RD(R) = 1 THEN RETURN
3230 PRINT
3240 PRINT "AS YOU'LL SOON FIND OUT, THERE IS A DOOR TO THE
     EAST. WE CALL THAT THE EAST DOOR.";
3250 RD(R) = 1
3260 RETURN
3270 PRINT "YOU ARE BEHIND THE YELLOW DOOR."
3280 IF RD(R) = 1 THEN RETURN
3290 PRINT
3300 PRINT "YOU ARE IN A YELLOW CHAMBER."
3310 RD(R) = 1
3320 RETURN
3330 PRINT "YOU ARE IN THE YELLOW ENTRYWAY"
3340 RETURN
3350 PRINT "YOU ARE IN THE YELLOW LABORATORY"
3360 IF RD(R) = 1 THEN RETURN
3370 PRINT
3380 PRINT "NOTE: THIS IS PRONOUNCED LA-BOR-A-TORY, NOT
     LA-BRA-TORY. THIS IS A HIGH-CLASS ADVENTURE."
3390 RD(R) = 1
3400 RETURN
3410 PRINT "YOU ARE IN THE YELLOW ATTIC"
3420 IF RD(R) = 1 THEN RETURN
3430 PRINT
3440 PRINT "IF THIS WERE A HIGH-RESOLUTION ADVENTURE AND
     YOU HAD A COLOR MONITOR, YOU COULD SEE THIS IS REALLY
     MORE"
3450 PRINT "PALE YELLOW THAN THE DEEP YELLOW YOU IMAGINE
     IT TO BE."
3460 RD(R) = 1
3470 RETURN
3480 PRINT "YOU ARE IN THE YELLOW POWER ROOM"
3490 PRINT
3500 PRINT "NOTE: THIS IS THE 'POWER' ROOM, NOT THE 'POWDER'
     ROOM. PLEASE ACT APPROPRIATELY.":
3510 PRINT
3520 PRINT "THERE IS A HUGE KNIFE SWITCH ON THE WALL THAT
     CAN BE THROWN UP OR DOWN."
3530 PRINT
3540 IF SW = 1 THEN PRINT "THE SWITCH IS IN THE 'ON' (UP)
     POSITION.";
3550 IF SW = 0 THEN PRINT "THE SWITCH IS IN THE 'OFF' (DOWN)
     POSITION."
3560 RETURN
3570 PRINT "YOU ARE IN THE YELLOW SAFETY ROOM"
3580 PRINT
3590 PRINT "THERE IS A LARGE YELLOW BUTTON ON THE WALL."
3600 PRINT
3610 PRINT "A SIGN ABOVE THE BUTTON READS:"
3620 PRINT
3630 PRINT " 'DUE TO TEMPORARY WIRING PROBLEMS WE ARE NOT
     RESPONSIBLE FOR PUSHING THE DISCUMBOBULATOR AT THIS
     TIME.' "
3640 PRINT
3650 " '(IN FACT WE ARE NOT RESPONSIBLE FOR PUSHING IT AT ANY
     TIME)' "
3660 RETURN
3670 PRINT "YOU ARE IN A COBWEBBY ANNEX"
3680 RETURN
3690 PRINT "YOU STAND BEFORE A YELLOW WALL"
3700 PRINT
3710 IF OD = 1 THEN PRINT "A YELLOW HOLE PIERCES THE YELLOW
     WALL TO THE NORTH."
3720 IF OB(6) = 90 THEN PRINT "A STRANGE OBJECT PROTRUDES FROM
     THE KEYHOLE."
3730 IF OD = 1 OR OB(6) = 90 THEN RETURN
3740 PRINT "A STRANGELY SHAPED KEYHOLE IS IN THE WALL."
3750 RETURN
3760 PRINT "YOU ARE UP IN A TREE"
3770 RETURN
3780 PRINT "YOU ARE HIGH UP IN THE YELLOW TREE"
3790 RETURN
```

```
3800 PRINT "YOU ARE PERCHED ON THE TURTLE'S BACK"
3810 PRINT
3820 PRINT "YOU CAN NOW SEE THE TOP OF THE SHELF VERY
     CLEARLY."
3830 RETURN
3840 PRINT "YOU ARE IN THE MAINTENANCE ROOM"
3850 PRINT
3860 PRINT "IN THE CORNER LIES A KEESHOND, SOUND ASLEEP. ON
     THE WALL IS A RED BUTTON."
3870 PRINT
3880 PRINT "A LARGE SIGN UNDER THE BUTTON READS: "
3890 PRINT
3900 PRINT " ' RESET BUTTON' "
3910 PRINT " FOR EMERGENCY USE ONLY"
3920 RETURN
3930 PRINT "YOU ARE IN THE TREASURE CAVE"
3940 PRINT "YOU HAVE FABULOUS RICHES BEYOND YOUR WILDEST
     DREAMS. TO THE EAST IS AN EXIT LEADING TO PITTSBURGH. "
3950 PRINT
3960 IF OB(1) = 91 THEN PRINT "THE ";WJ$;" TURNS INTO A
     BEAUTIFUL "
3970 IF OB(1) = 91 THEN PRINT "PRINCESS (OR PRINCE) AND YOU
     SCORE 123.5 OUT OF 123.5 POINTS . . . A PERFECT SCORE."
3980 IF OB(1) = 91 THEN END
3990 PRINT "YOU HAVE SCORED 1 OUT OF A POSSIBLE 123.5 POINTS.
     NOT BAD."
4000 END
4100 REM     LOOK, DESCRIBE, READ ROUTINES
4110 IF N1$ = "" THEN GOTO 4330
4120 IF R = 1 AND N1$ = WS$ THEN PRINT "IT READS, 'HI, CLIFF!' "
4130 IF R = 1 AND N1$ = WS$ THEN GOTO 4350
4140 IF N1$ = WU$ AND OB(1) = 91 THEN PRINT "SOMEONE HAS
     WRITTEN ON THE TURTLE'S BACK THE VERSE:"
4150 IF N1$ = WU$ AND OB(1) = 91 THEN PRINT
4160 IF N1$ = WU$ AND OB(1) = 91 THEN PRINT "  WHERE EAGLES
     SOAR, THERE MAGICKS BE,"
4170 IF N1$ = WU$ AND OB(1) = 91 THEN PRINT "  BE CERTAIN THAT I'M
     THERE WITH THEE,"
4180 IF N1$ = WU$ AND OB(1) = 91 THEN PRINT "  JUST SAYEST THAT IT
     MAY BE RED"
4190 IF N1$ = WU$ AND OB(1) = 91 THEN PRINT "  THOUGH YELLOW
     MAY DO IN ITS STEAD."
4200 IF N1$ = WU$ AND OB(1) = 91 THEN PRINT
4210 IF N1$ = WU$ AND OB(1) = 91 THEN PRINT "INCIDENTALLY, THE
     TURTLE HAS A PURPLE STRIPE ON ITS BACK."
4220 IF N1$ = WU$ AND OB(1) = 91 THEN PRINT
4230 IF N1$ = WU$ AND OB(1) = 91 THEN PRINT "A FINAL NOTE: THIS IS
     A MUD TURTLE."
4240 IF N1$ = WU$ AND OB(1) = 91 THEN GOTO 4350
4250 IF N1$ = X6$ OR N1$ = X1$ THEN GOTO 8120
4260 IF N1$ = X2$ AND OB(6) = 91 THEN GOTO 8310
4270 IF N1$ = XC$ AND OB(7) = 91 THEN PRINT "ISN'T THAT'
     INTERESTING. THE LEAF STRETCHES LIKE A RUBBER BAND.
     BOING . . . BOING."
4280 IF N1$ = XC$ AND OB(7) = 91 THEN GOTO 100
4290 IF R = 22 AND N1$ = XI$ THEN PRINT "YOU CAN'T SEE WHAT'S UP
     THERE AS IT'S TOO HIGH."
4300 IF R = 22 AND N1$ = XI$ THEN GOTO 100
4310 IF R = 32 AND OD = 0 AND N1$ = XN$ THEN PRINT "AS I SAID, IT IS
     SHAPED VERY QUEERLY. THIS IS EXTREMELY PUZZLING."
4320 IF R = 32 AND OD = 0 AND N1$ = XN$ THEN GOTO 100
4330 PRINT "I GUESS YOU JUST WANT A GENERAL PEEK."
4340 PRINT
4350 GOTO 1410
4400 REM     GETTING ROUTINES
4410 IF N1$ = "" THEN PRINT "YOU FORGOT TO ENTER A NOUN"
4420 IF N1$ = "" THEN GOTO 100
4430 REM     NOW GO TO CHECK IF NOUN MEANS ANYTHING
4440 GOSUB 5300
4450 IF X = 1 AND WT = 1 THEN GOTO 7980
4460 IF OB(X) = 91 THEN PRINT "YOU'VE ALREADY GOT THE ";N1$"."
4470 IF OB(X) = 91 THEN GOTO 100
4480 IF OB(X) = R THEN OB(X) = 91
4490 IF OB(8) = 90 AND N1$ = XA$ THEN OB(8) = 91
4500 IF OB(8) = 91 AND N1$ = XA$ THEN X = 8
4510 IF OB(6) = 90 AND N1$ = X2$ THEN OB(6) = 91
4520 IF OB(6) = 91 AND N1$ = X2$ THEN X = 6
4530 IF OB(X) = 91 THEN PRINT "OKAY, YOU HAVE THE ";N1$
4540 IF X = 2 OR X = 3 OR X = 4 OR X = 5 THEN PP = PP + 1
4550 IF OB(X) = 91 THEN GOTO 100
4560 PRINT "THERE IS NO ";N1$;" HERE."
4570 GOTO 100
4580 REM     END OF GETTING ROUTINE
4600 REM     DROPPING ROUTINES
4610 IF N1$ = "" THEN PRINT "YOU FORGOT TO ENTER A NOUN"
4620 IF N1$ = "" THEN GOTO 100
4630 GOSUB 5300
4640 IF OB(X) = 91 THEN GOTO 4680
4650 PRINT
4660 PRINT "YOU'RE NOT CARRYING THE ";N1$;"."
4670 GOTO 100
4680 OB(X) = R
4690 PRINT
4700 IF WT = 1 THEN GOTO 4720
4710 IF R = 10 AND OB(1) = 10 THEN GOTO 7900
4720 PRINT "YOU'VE DROPPED THE ";N1$;"."
4730 IF X = 2 OR X = 3 OR X = 4 OR X = 5 THEN PP = PP - 1
4740 GOTO 100
4800 REM     INVENTORY ROUTINES
4810 X = 91
4830 PRINT
4840 PRINT "YOU ARE CARRYING:"
4850 PRINT
4860 GOSUB 5500
4880 GOTO 100
4900 REM     QUITTING
4910 PRINT
4920 PRINT "DO YOU WANT TO SAVE THIS GAME? ";
4930 INPUT A$
4940 IF A$ = "YES" THEN GOTO 5000
4950 CALL -936
4960 PRINT
4970 PRINT "FARE THEE WELL"
4980 END
5000 REM     SAVING GAMES ROUTINE
5010 REM     THIS IS FOR THE APPLE
5020 REM     BUT ALL YOU HAVE TO DO
5030 REM     IS SAVE THE VARIABLE SHOWN
5040 REM     IN LINES 6180 TO 6190
5050 REM     WITHIN QUOTES IS A CONTROL-D
5060 PRINT "■";"OPEN EASYFILE"
5070 PRINT "■";"WRITE EASYFILE"
5080 PRINT R
5090 FOR A = 1 TO 10: PRINT OB(A): NEXT A
5100 PRINT BO: PRINT BT: PRINT CL: PRINT ET: PRINT FL: PRINT HB:
     PRINT OD: PRINT PC: PRINT PP: PRINT R: PRINT SO: PRINT SW:
     PRINT TK: PRINT WO: PRINT WT
5105 FOR A = 1 TO 9: PRINT OB(A): NEXT A
5110 PRINT "■";"CLOSE EASYFILE"
5120 PRINT
5130 PRINT "DO YOU WANT TO CONTINUE PLAYING? ";
5140 INPUT A$
5150 IF A$ = "YES" THEN CALL -936
5160 IF A$ = "YES" THEN GOTO 1200
5170 GOTO 4960
5200 REM     RESTORING GAMES ROUTINE
5210 REM     THIS IS FOR THE APPLE
5220 REM     THE QUOTES BELOW CONTAIN A CONTROL-D
5230 PRINT "■";"OPEN EASYFILE"
5240 PRINT "■";"READ EASYFILE"
5250 INPUT R
5260 FOR A = 1 TO 10: INPUT OB(A): NEXT A
5270 INPUT BO: INPUT BT: INPUT CL: INPUT ET: INPUT FL: INPUT HB:
     INPUT OD: INPUT PC: INPUT PP: INPUT R: INPUT SO: INPUT SW:
     INPUT TK: INPUT WO: INPUT WT
5275 FOR A = 1 TO 9: INPUT OB(A): NEXT A
5280 PRINT "■";"CLOSE EASYFILE"
5290 GOTO 1200
5300 REM     SUBROUTINE TO CHECK NOUNS
5310 IF N1$ = WU$ THEN X = 1
5320 IF N1$ = X1$ AND OB(2) = 91 THEN X = 2
5330 IF N1$ = X1$ AND OB(3) = 91 THEN X = 3
5340 IF N1$ = X1$ AND OB(4) = 91 THEN X = 4
5350 IF N1$ = X1$ AND OB(5) = 91 THEN X = 5
5360 IF V1$ = WL$ THEN GOTO 5410: REM     SKIP NEXT SECTION IF
     DROPPING PUZZLE
```

```
5370 IF N1$ = X1$ AND OB(2) = R THEN X = 2
5380 IF N1$ = X1$ AND OB(3) = R THEN X = 3
5390 IF N1$ = X1$ AND OB(4) = R THEN X = 4
5400 IF N1$ = X1$ AND OB(5) = R THEN X = 5
5410 IF N1$ = XC$ THEN X = 7
5420 IF N1$ = XA$ THEN X = 8
5430 IF N1$ = XM$ THEN X = 9
5440 RETURN
5500 REM      SUBROUTINE TO PRINT OUT INVENTORY OR OBJECT
     IN ROOM
5510 TS = PP
5520 PP = 0
5530 IF X = OB(1) THEN PRINT "A TURTLE"
5540 IF X = OB(1) THEN Y = 1
5550 FOR A = 2 TO 5
5560 IF OB(A) = X THEN PP = PP + 1
5570 NEXT A
5580 IF PP > 0 THEN PRINT PP;" PIECE(S) OF A JIGSAW PUZZLE"
5590 IF PP > 0 THEN Y = 1
5600 IF X = OB(6) THEN PRINT "A COMPLETE JIGSAW PUZZLE"
5610 IF X = OB(6) THEN Y = 1
5620 IF X = OB(7) THEN PRINT "A LARGE LEAF"
5630 IF X = OB(7) THEN Y = 1
5640 IF X = OB(8) THEN PRINT "A METAL KEY"
5650 IF X = OB(8) THEN Y = 1
5660 IF X = OB(9) THEN PRINT "A LUMP OF TWISTED AND
     SMOLDERING CARDBOARD"
5670 IF X = OB(9) THEN Y = 1
5680 IF Y = 0 THEN PRINT "NOTHING"
5690 Y = 0
5700 PP = TS
5710 RETURN
5800 REM      SET UP ROUTINES
5810 DIM N(90),S(90),E(90),W(90),U(90),D(90),X(4),Y(4),RD(90),OB(20)
5820 DIM A$(20),V1$(20),N1$(20),W1$(5),W2$(5),W3$(4),W4$(4),W5$(2),
     W6$(4),W7$(1),W8$(1),W9$(1),WA$(1),WB$(1)
5830 DIM WC$(1),WD$(4),WE$(4),WF$(4),WG$(1),WH$(3),WI$(9),WJ$(3),
     WK$(4),WL$(4),WM$(4),WN$(4),D$(5)
5840 DIM WO$(8),WP$(7),WQ$(5),WR$(4),WS$(4),WT$(5),WU$(6),WV$(4),
     WW$(3),WX$(6),WY$(4),WZ$(5),X1$(5),X2$(6),X3$(3)
5850 DIM X4$(4),X5$(4),X6$(6),X7$(5),XA$(3),X8$(8),X9$(4),XB$(3),XC$(4),
     XD$(5),XE$(6),XF$(4),XG$(6),XH$(4),XI$(5)
5860 DIM XJ$(4),XK$(6),XL$(8),XM$(4),X0$(4),W0$(5),XN$(7),XO$(4),XP$(8)
5870 OB(1) = 2
5880 OB(2) = 9
5890 OB(3) = 11
5900 OB(4) = 12
5910 OB(5) = 19
5920 OB(7) = 34
5930 OB(8) = 35
5940 OB(6) = 0
5950 OB(10) = 0
5960 FOR A = 1 TO 50
5970 RD(A) = 0
5980 NEXT A
5990 SW = 1
6000 REM      DIRECTION LOADING
6010 FOR A = 1 TO 40
6020 N(A) = 0
6030 S(A) = 0
6040 E(A) = 0
6050 W(A) = 0
6060 U(A) = 0
6070 D(A) = 0
6080 NEXT A
6090 N(1) = 4
6100 E(1) = 2
6110 N(2) = 5
6120 E(2) = 3
6130 W(2) = 1
6140 N(3) = 6
6150 W(3) = 2
6160 S(4) = 1
6170 E(4) = 5
6180 S(5) = 2
6190 E(5) = 6
6200 W(5) = 4
6210 S(6) = 3
```

```
6220 W(6) = 5
6230 N(8) = 14
6240 S(8) = 10
6250 E(9) = 10
6260 W(11) = 10
6270 N(12) = 10
6280 N(13) = 16
6290 E(13) = 14
6300 N(14) = 17
6310 S(14) = 8
6320 E(14) = 15
6330 W(14) = 13
6340 W(15) = 14
6350 S(16) = 13
6360 E(16) = 17
6370 N(17) = 20
6380 S(17) = 14
6390 E(17) = 18
6400 W(17) = 16
6410 U(18) = 33
6420 W(18) = 17
6430 N(20) = 23
6440 S(20) = 17
6450 E(20) = 21
6460 N(21) = 24
6470 U(21) = 34
6480 W(21) = 20
6490 E(22) = 23
6500 S(23) = 20
6510 E(23) = 24
6520 W(23) = 22
6530 S(24) = 21
6540 E(24) = 25
6550 W(24) = 23
6560 E(25) = 26
6570 W(25) = 24
6580 N(26) = 28
6590 W(26) = 25
6600 U(27) = 29
6610 E(27) = 28
6620 S(28) = 26
6630 W(28) = 27
6640 D(29) = 31
6650 E(29) = 30
6660 W(30) = 29
6670 S(30) = 28
6680 E(31) = 32
6690 W(32) = 31
6700 D(33) = 19
6710 D(34) = 21
6720 D(35) = 22
6730 REM      WORD LIST
6740 W1$ = "NORTH"
6750 W2$ = "SOUTH"
6760 W3$ = "EAST"
6770 W4$ = "WEST"
6780 W5$ = "UP"
6790 W6$ = "DOWN"
6800 W7$ = "N"
6810 W8$ = "S"
6820 W9$ = "E"
6830 WA$ = "W"
6840 WB$ = "U"
6850 WC$ = "D"
6860 W0$ = "RIGHT"
6870 WD$ = "SAVE"
6880 WE$ = "QUIT"
6890 WF$ = "READ"
6900 WG$ = "I"
6910 WH$ = "INV"
6920 WI$ = "INVENTORY"
6930 WJ$ = "GET"
6940 WK$ = "TAKE"
6950 WL$ = "DROP"
6960 WM$ = "WAIT"
6970 WN$ = "LOOK"
6980 WO$ = "DESCRIBE"
6990 WP$ = "RESTORE"
```

```
7000 WQ$ = "CLIFF"
7010 WR$ = "READ"
7020 WS$ = "SIGN"
7030 WT$ = "CLIMB"
7040 WU$ = "TURTLE"
7050 WV$ = "JUMP"
7060 WW$ = "RED"
7070 WX$ = "YELLOW"
7080 WY$ = "LONG"
7090 WZ$ = "SHORT"
7100 X0$ = "LEFT"
7110 X1$ = "PIECE"
7120 X2$ = "PUZZLE"
7130 X3$ = "EAT"
7140 X4$ = "MIRE"
7150 X5$ = "MUCK"
7160 X6$ = "PIECES"
7170 X7$ = "SOLVE"
7180 XA$ = "KEY"
7190 X8$ = "EUTHERIA"
7200 X9$ = "OPEN"
7210 XB$ = "BOX"
7220 XC$ = "LEAF"
7230 XD$ = "THROW"
7240 XE$ = "SWITCH"
7250 XF$ = "PUSH"
7260 XG$ = "BUTTON"
7270 XH$ = "OPEN"
7280 XI$ = "SHELF"
7290 XJ$ = "TURN"
7300 XK$ = "INSERT"
7310 XL$ = "ASSEMBLE"
7320 XM$ = "LUMP"
7330 XO$ = "KISS"
7340 XP$ = "KEESHOND"
7350 REM     SETTING FLAGS
7360 R = 1
7370 Y = 0
7380 X = 0
7390 PRINT "RESTORE PREVIOUS GAME? ('YES' OR 'NO') ";
7400 INPUT A$
7410 CALL  - 936
7420 IF A$ = "YES" THEN GOTO 5200
7430 GOSUB 1900
7440 GOTO 1800
7500 REM     SPECIAL HANDLING SECTION
7510 IF R = 3 THEN PRINT
7520 IF V1$ = W5$ AND R = 3 THEN PRINT "THAT'S THE RIGHT IDEA,
     BUT THE WRONG WORD."
7530 IF V1$ = WB$ AND R = 3 THEN PRINT "THAT'S THE RIGHT IDEA,
     BUT THE WRONG WORD."
7540 IF R = 3 THEN GOTO 100
7550 IF R = 7 THEN PRINT
7560 IF V1$ = W6$ AND R = 7 THEN PRINT "YIIIIIII.........."
7570 IF V1$ = WC$ AND R = 7 THEN PRINT "YIIIIIII.........."
7580 IF R = 7 THEN PRINT
7590 IF R = 7 THEN GOTO 4960
7600 IF SO = 1 THEN FOR A = 1 TO 100
7610 IF SO = 1 THEN NEXT A
7620 IF SO = 1 THEN R = 37
7630 IF SO = 1 THEN GOTO 1320
7640 PRINT
7650 IF R = 7 THEN PRINT "A VOICE WHISPERS IN YOUR EAR . . . 'THAT
     SECTION IS UNDER REPAIR . . . NOT OPEN YET.' "
7660 IF R = 7 THEN GOTO 100
7670 IF R = 7 THEN PRINT
7680 IF R = 7 THEN PRINT "POOF . . . POOF!!!"
7690 IF R = 7 THEN FOR A = 1 TO 100
7700 IF R = 7 THEN NEXT A
7710 IF R = 7 THEN R = 8
7720 GOTO 1320
7730 PRINT "THERE IS A MIGHTY TREMBLING, SHIVERING, SHAKING,
     AND SHUDDERING AS THE WALLS COME TUMBLING DOWN."
7740 PRINT
7750 PRINT "YOUR LAST THOUGHT IS THAT SOME OF THE SIGNS ARE
     FOR REAL."
7760 PRINT
7770 GOTO 4960
7800 REM     RESULTS OF EATING
7810 PRINT
7820 IF R = 10 AND N1$ = X4$ THEN PRINT "YUM, YUM, THAT WAS
     GOOD . . . YOU'RE STRONGER . . . BUT IT WASN'T QUITE
     ENOUGH."
7830 IF R = 10 AND N1$ = X4$ THEN ET = ET + 1
7840 IF ET = 4 THEN PRINT
7850 IF ET = 4 THEN PRINT "NO, THAT WAS ENOUGH! YOU SLOWLY
     SINK OUT OF SIGHT . . . NEVER TO BE SEEN AGAIN."
7860 IF ET = 4 THEN GOTO 4960
7870 IF R = 10 AND N1$ = X4$ THEN GOTO 100
7880 PRINT "DON'T BE SILLY. YOU SIMPLY CAN'T EAT ";N1$;"S."
7890 GOTO 100
7900 REM     WU HANDLING
7910 WT = 1
7920 PRINT "THE ";WU$;" LANDS WITH A PLOP IN THE ";
     X4$;"PERMITTING YOU TO CROSS."
7930 N(10) = 8
7940 S(10) = 12
7950 E(10) = 11
7960 W(10) = 9
7970 GOTO 100
7980 PRINT "THE ";WU$;" IS PEFECTLY HAPPY WHERE HE IS AND
     REFUSES TO COME WITH YOU."
7990 GOTO 100
8000 PRINT
8010 PRINT "THE WEIGHT OF THE ";X1$;"S, CAUSES YOU, THE ";
     WU$;" AND THE ";X1$;"S TO SINK SLOWLY OUT OF SIGHT."
8020 GOTO 4960
8100 REM     PUZZLE HANDLING
8110 PRINT
8120 IF PP = 0 THEN PRINT "YOU AREN'T CARRYING ANY PIECES."
8130 IF PP = 0 THEN GOTO 100
8140 IF R = 9 OR R = 10 OR R = 11 OR R = 12 THEN PRINT "THE LIGHT
     ISN'T VERY GOOD HERE AND I CAN'T MAKE IT OUT."
8150 IF R = 9 OR R = 10 OR R = 11 OR R = 12 THEN GOTO 100
8160 PRINT "THE PUZZLE PIECES ARE VERY CONFUSING. PERHAPS IF
     YOU SOLVED THE PUZZLE."
8170 GOTO 100
8180 PRINT
8190 IF PP = 0 THEN PRINT "YOU HAVE NO PUZZLE PIECES."
8200 IF PP = 0 THEN GOTO 100
8210 IF PP = 4 THEN OB(2) = 0
8220 IF PP = 4 THEN OB(3) = 0
8230 IF PP = 4 THEN OB(4) = 0
8240 IF PP = 4 THEN OB(5) = 0
8250 IF PP = 4 THEN OB(6) = 91
8260 IF PP = 4 THEN PRINT "BY GOLLY . . . I'VE SUCCESSFULLY FITTED
     TOGETHER THE PUZZLE PIECES."
8270 IF PP = 4 THEN PP = 0
8280 IF PP = 0 THEN GOTO 100
8290 PRINT "YOU DON'T HAVE ALL THE PIECES."
8300 GOTO 100
8310 IF PC = 3 THEN PC = 0
8320 PC = PC + 1
8330 IF PC = 1 THEN PRINT "THE PUZZLE WAVERS AND SHIFTS. IT
     SHOWS A GREAT TREASURE IN A RED CAVE."
8340 IF PC = 1 THEN GOTO 100
8350 IF PC = 2 THEN PRINT "THE PUZZLE WAVERS AND SHIFTS. THE
     WORD 'EUTHERIA' FORMS ON ITS SURFACE."
8360 IF PC = 2 THEN GOTO 100
8370 PRINT "YOU SEE THE ";X21;" IS SHAPED LIKE A ";XA$;"."
8380 GOTO 100
8400 REM     SWITCHES AND BUTTONS
8410 IF N1$ = XE$ AND BT = 0 THEN PRINT
8420 IF N1$ = XE$ AND BT = 0 THEN PRINT "80 ZILLION FURFLES
     OF ELECTRICITY SURGE THROUGH WHAT USED TO BE
     YOU!"
8430 IF N1$ = XE$ AND BT = 0 THEN GOTO 4960
8440 IF N1$ = XE$ AND BT = 1 THEN PRINT
8450 IF N1$ = XE$ AND BT = 1 THEN PRINT "THERE IS A LOUD
     'KERCHUNK' AS A RELAY OPENS SOMEWHERE AND THE
     LIGHTS DIM."
8460 PRINT
8470 IF N1$ = XE$ AND BT = 1 AND SW = 0 THEN FL = 1
8480 IF N1$ = XE$ AND BT = 1 AND SW = 1 THEN FL = 0
```

```
8490 IF N1$ = XE$ AND BT = 1 THEN SW = FL
8500 IF N1$ = XE$ AND BT = 1 AND SW = 1 THEN PRINT "THE SWITCH
     IS ON."
8510 IF N1$ = XE$ AND BT = 1 AND SW = 0 THEN PRINT "THE SWITCH
     IS OFF."
8520 IF SW = 1 THEN HB = 0
8530 IF SW = 0 THEN HB = 1
8540 IF N1$ = XE$ THEN GOTO 100
8550 PRINT "YOU DON'T HAVE ANY ";N1$;"S."
8560 GOTO 100
8570 PRINT
8580 IF OB(7) = 30 AND N1$ = XG$ THEN PRINT "STANDING ON THE ";
     XC$;", YOU PUSH THE BUTTON. THERE IS A LOUD KLUNK
     SOMEWHERE.";
8590 IF OB(7) = 30 AND N1$ = XG$ AND BT = 0 THEN FL = 1
8600 IF OB(7) = 30 AND N1$ = XG$ AND BT = 1 THEN FL = 0
8610 IF OB(7) = 30 AND N1$ = XG$ THEN BT = FL
8620 IF OB(7) = 30 AND N1$ = XG$ THEN GOTO 100
8630 IF N1$ = XG$ THEN PRINT "80 ZILLION FURFLES OF ELECTRICITY
     SURGE THROUGH WHAT USED TO BE YOU!"
8640 IF N1$ = XG$ THEN GOTO 4960
8650 PRINT "PUSHING THE ";N1$;" DOESN'T WORK. TRY PUSHING THE
     BUTTON."
8660 GOTO 100
8700 REM     CLIMBING SPECIAL ROUTINES
8710 PRINT
8720 IF R = 22 AND N1$ = WU$ AND OB(1) = 91 THEN PRINT "HOW
     CAN YOU CLIMB ON SOMETHING YOU'RE CRADLING IN
     YOUR ARMS?"
8730 IF R = 22 AND N1$ = WU$ AND OB(1) = 91 THEN GOTO 100
8740 IF R = 22 AND N1$ = WU$ AND OB(1) = 22 THEN R = 35
8750 IF R = 35 THEN GOTO 1200
8760 IF R = 21 THEN R = 34
8770 IF R = 34 THEN GOTO 1200
8780 IF R = 18 THEN R = 33
8790 IF R = 33 THEN GOTO 1200
8800 PRINT "I SIMPLY CAN'T DO THAT HERE."
8810 GOTO 100
8900 REM     SPECIAL OPENING ROUTINES
8910 PRINT
8920 IF N1$ = XB$ AND R = 19 AND HB = 0 THEN PRINT "YOU PULL AND
     TUG BUT IT IS CLOSED BY AN ELECTROMAGNETIC FORCE."
8930 IF N1$ = XB$ AND R = 19 and HB = 0 THEN GOTO 100
8940 IF BO = 1 THEN PRINT "THE BOX IS OPEN."
8950 IF BO = 1 THEN GOTO 100
8960 IF N1$ = XB$ AND R = 19 AND HB = 1 THEN PRINT "THE COVER
     COMES OFF EASILY EXPOSING A KEYHOLE."
8970 IF N1$ = XB$ AND R = 19 AND HB = 1 THEN BO = 1
8980 IF N1$ = XB$ AND R = 19 AND HB = 1 THEN GOTO 100
8990 PRINT "WHAT DO YOU WANT ME TO OPEN?"
9000 GOTO 100
9100 REM     SPECIAL INSERTING ROUTINES
9110 IF R = 19 OR R = 32 THEN GOTO 9150
9120 PRINT
9130 PRINT "THAT WON'T WORK HERE."
9140 GOTO 100
9150 IF R = 32 THEN GOTO 9290
9160 IF N1$ = XA$ AND OB(8) = 91 THEN OB(8) = 90
9170 IF OB(8) = 90 THEN PRINT
9180 IF OB(8) = 90 THEN PRINT "THE KEY IS IN THE KEYHOLE."
9190 IF OB(8) = 90 THEN GOTO 100
9200 IF N1$ = X2$ AND OB(6) = 91 THEN PRINT
9210 IF N1$ = X2$ AND OB(6) = 91 THEN PRINT "THERE IS A SIZZLING
     SOUND AS SOMETHING FALLS FROM THE KEYHOLE."
9220 IF N1$ = X2$ AND OB(6) = 91 THEN OB(9) = 19
9230 IF N1$ = X2$ AND OB(6) = 91 THEN OB(6) = 0
9240 IF N1$ = X2$ AND OB(9) = 19 THEN PRINT
9250 IF N1$ = X2$ AND OB(9) = 19 THEN GOTO 1410
9260 PRINT
9270 PRINT "THAT'S THE RIGHT IDEA BUT SOMETHING IS MISSING."
9280 GOTO 100
9290 IF OB(6) = 91 AND N1$ = X2$ THEN OB(6) = 90
9300 IF OB(6) = 90 THEN PRINT
9310 IF OB(6) = 90 THEN PRINT "THE ";X2$;" IS IN THE KEYHOLE. PART
     OF IT PROTRUDES FROM THE KEYHOLE."
9320 IF OB(6) = 90 THEN GOTO 100
9330 IF OB(8) = 91 AND N1$ = XA$ THEN PRINT
9340 IF OB(8) = 91 AND N1$ = XA$ THEN PRINT "OH NO!!! SOMETHING
     ON THE OTHER SIDE OF THE WALL"
9350 IF OB(8) = 91 AND N1$ = XA$ THEN PRINT "HAS GRABBED YOU
     AND IS EATING THE KEY. IT EATS YOU FOR DESSERT."
9360 IF OB(8) = 91 AND N1$ = XA$ THEN GOTO 4960
9370 PRINT
9380 PRINT "THAT'S THE RIGHT IDEA BUT SOMETHING IS MISSING."
9390 GOTO 100
9400 REM     TURNING THINGS
9410 IF R = 19 OR R = 32 THEN GOTO 9450
9420 PRINT
9430 PRINT "ALL I CAN DO IS TURN AROUND HERE..I'LL DO IT.
     WHOOPEE!!!"
9440 GOTO 100
9450 IF R = 32 THEN GOTO 9680
9460 IF BO = 0 THEN PRINT
9470 IF BO = 0 THEN PRINT "I SEE NO KEYHOLE HERE."
9480 IF BO = 0 THEN GOTO 100
9490 IF OB(8) = 90 THEN GOTO 9530
9500 PRINT
9510 PRINT "THERE'S NOTHING IN THE KEYHOLE."
9520 GOTO 100
9530 IF N1$ = W0$ THEN PRINT
9540 IF N1$ = W0$ THEN PRINT "THE HUMMOCK'S GARBAGE CHUTE
     OPENS AND YOU SLIDE RIGHT OUT OF THE GAME."
9550 IF N1$ = W0$ THEN GOTO 4960
9560 IF TK = 1 THEN PRINT
9570 IF TK = 1 THEN PRINT "THE MACHINERY IS JAMMED."
9580 IF TK = 1 THEN GOTO 100
9590 IF N1$ = X0$ THEN PRINT
9600 IF N1$ = X0$ THEN PRINT "HIDDEN MACHINERY RAISES THE
     HUMMOCK UP TO THE LEVEL OF THE REST OF THE GAME."
9610 IF N1$ = X0$ THEN S(22) = 19
9620 IF S(22) = 19 THEN TK = 1
9630 IF N1$ = X0$ THEN N(19) = 22
9640 IF N1$ = X0$ THEN GOTO 100
9650 PRINT
9660 PRINT "YOU HAVE TO TELL ME TO TURN THE ";XA$;" RIGHT
     OR LEFT."
9670 GOTO 100
9680 REM     ROOM 32
9690 IF OB(6) = 90 THEN GOTO 9730
9700 PRINT
9710 PRINT "THERE'S NOTHING IN THE KEYHOLE."
9720 GOTO 100
9730 IF N1$ = W0$ THEN PRINT
9740 IF N1$ = W0$ THEN PRINT "THE WALLS BEGIN TO MOVE IN ON
     YOU. YOU STRUGGLE FRANTICALLY, BUT IT'S TOO LATE."
9750 IF N1$ = W0$ THEN GOTO 4960
9760 IF N1$ = X0$ THEN PRINT
9770 IF N1$ = X0$ THEN PRINT "A WELL-OILED OPENING APPEARS IN
     THE WALL."
9780 IF N1$ = X0$ THEN N(32) = 36
9790 IF N1$ = X0$ THEN S(36) = 32
9800 IF N1$ = X0$ THEN OB(6) = 0
9810 IF N1$ = X0$ THEN OD = 1
9820 IF N1$ = X0$ THEN GOTO 100
9830 PRINT
9840 PRINT "YOU HAVE TO TELL ME TO TURN THE ";X2$;" RIGHT OR
     LEFT."
9850 GOTO 100
9900 REM     MAGIC
9910 SO = 1
9920 CALL -936
9930 PRINT "POOF! POOF!"
9940 FOR A = 1 TO 200
9950 NEXT A
9960 R = 7
9970 GOTO 1320
```

DUNGEON FUN LEVEL FOUR

APPLE

```
2 REM
3 REM  HOME BASE
4 REM
6 REM

20 GOTO 5000: REM  INITIALIZE
50 IF PEEK (768) < > 25 OR PEEK (769) < >
   250 THEN 1000: REM NOBODY HERE YET,
   GET JOE
60 GOTO 2000: REM  ELSE A PARTY HAS
   JUST RETURNED, SO DEBRIEF THEM

100 REM  + + + MAIN MENU + + +
110 PRINT : PRINT
120 PRINT "YOU MAY ENTER THE ";
130 INVERSE : PRINT "C";: NORMAL
140 PRINT "AVERNS,": PRINT
150 PRINT "              OR TAKE A ";
160 INVERSE : PRINT "R";: NORMAL
170 PRINT "EST:": PRINT
180 PRINT "WHAT IS YOUR CHOICE?  ";
190 GET Q$: PRINT Q$: PRINT

200 REM  ** DECODE & SEND **
210 PRINT : IF Q$ - "C" THEN 250
220 PRINT "SO BE IT. REST WELL, "
    NAME$". . ."
230 PRINT D$"BSAVE "NAME$",A$300,L$A"
240 END
250 PRINT "MOVING OUT . . ."
260 PRINT D$"RUN DM 4"
290 END

1000 REM  + + + PERSONNEL + + +
1010 PRINT "THE ONLY FIGHTER
     AVAILABLE TODAY IS"
1020 PRINT : PRINT "  JOE."
1030 PRINT : PRINT "HE WILL BE WITH YOU
     IN A MOMENT."
1040 PRINT D$"BLOAD JOE"
1050 PRINT
1060 REM  THEN FALL INTO

2000 REM  + + + DEBRIEFING + + +

2010 REM  ** DEBRIEF WHOM? **
2020 NAME$ - ""
2030 FOR X - 0 TO 4
2040 Q$ - CHR$ ( PEEK (770 + X))
2050 IF Q$ - " " THEN 2070
2060 NAME$ - NAME$ + Q$
2070 NEXT X

2100 REM  ** CARRYING WHAT? **
2110 GOLD - 256 * PEEK (775) + PEEK (776)
2120 ANKH - PEEK (777)

2200 REM  ** REPORT **
2210 PRINT : PRINT "WELCOME BACK, "
     NAME$"!"
2220 PRINT : PRINT "YOU NOW HAVE "
     GOLD" IN GOLD";
2230 IF ANKH - 0 THEN PRINT ".": GOTO
     2260
2240 PRINT ","
2250 PRINT " AND A GOLDEN ANKH."
2260 PRINT

2300 REM  ** RECORD **
2310 PRINT D$"BSAVE "NAME$",A$300,L$A"
2350 GOTO 100

5000 REM  + + + INITIALIZE + + +

5010 REM  ** VARIABLES ETC. **
5020 D$ - CHR$ (4): REM  CTRL-D

5100 REM  ** TITLE **
5110 HOME
5120 VTAB 5: HTAB 11
5130 PRINT "THE DUNGEON MACHINE"
5140 VTAB 8: HTAB 13
5150 PRINT "(PRACTICE AREA)"
5180 VTAB 15
5190 GOTO 50

10000 REM  + + + TO INSTALL JOE + + +
10020 POKE 768,25: POKE 769,250
10050 POKE 770, ASC ("J")
10060 POKE 771, ASC ("O")
10070 POKE 772, ASC ("E")
10080 POKE 773, ASC (" ")
10090 POKE 774, ASC (" ")
10100 POKE 775,0
10110 POKE 776,10
10120 POKE 777,0
10130 PRINT CHR$ (4)"BSAVE JOE,
      A$300,L$A"
```

ATARI

```
2 REM
3 REM HOME BASE
4 REM
6 REM
20 GOTO 5000:REM INITIALIZE
30 IF PEEK(1536)< >25 OR
   PEEK(1537)< >250 THEN 1000:REM
   NOBODY HERE YET, GET JOE
40 GOTO 2000:REM ELSE A PARTY HAS
   JUST RETURNED, SO DEBRIEF THEM

100 REM + + + MAIN MENU + + +
110 PRINT :PRINT
120 PRINT "YOU MAY ENTER THE ";
130 PRINT "■";:REM THE CHARACTER IN
    QUOTES IS AN INVERSE C
140 PRINT "AVERNS,":PRINT
150 PRINT "              OR TAKE A ";
160 PRINT "■";:REM THE CHARACTER IN
    QUOTES IS AN INVERSE R
170 PRINT "EST:":PRINT
180 PRINT "WHAT IS YOUR CHOICE? ";
190 POKE 752,0:PRINT " ";:GET #1,Q:POKE
    752,1:Q$ - CHR$(Q):PRINT Q$:PRINT

200 REM ** DECODE & SEND **
210 PRINT :IF Q$ - "C" THEN 250
220 PRINT "SO BE IT. REST WELL. ";
    NAME$;". . ."
230 FILE$(3) - NAME$
232 OPEN #2,8,0,FILE$
234 FOR X - 1536 TO 1545
236 PUT #2,PEEK(X)
238 NEXT X:PUT #2,155:CLOSE #2
240 END
250 PRINT "MOVING OUT. . ."
260 RUN "D:DM4"
290 END

1000 REM + + + PERSONNEL + + +
1010 PRINT "THE ONLY FIGHTER
     AVAILABLE TODAY IS"
1020 PRINT :PRINT "  JOE."
1030 PRINT :PRINT "HE WILL BE WITH YOU
     IN A MOMENT."
1040 I - 1536:TRAP 1044:OPEN #2,4,0,"D:JOE"
1041 GET #2,Q
1042 POKE I,Q:I - I + 1
1043 GOTO 1041
1044 ERR - PEEK(195)
1045 IF ERR - 136 THEN 1049
1046 PRINT "ERROR NUMBER ";ERR;" HAS
     OCCURRED!"
1049 CLOSE #2:TRAP 40000
1050 PRINT
1060 REM THEN FALL INTO

2000 REM + + + DEBRIEFING + + +

2010 REM ** DEBRIEF WHOM? **
2020 NAME$ - ""
2030 FOR X - 1 TO 5
2040 Q$ - CHR$(PEEK(1537 + X))
2050 IF Q$ - " " THEN 2070
2060 NAME$(X,X) - Q$
2070 NEXT X

2100 REM ** CARRYING WHAT? **
2110 GOLD - 256*PEEK(1543) + PEEK(1544)
2120 ANKH - PEEK(1545)

2200 REM ** REPORT **
2210 PRINT :PRINT "WELCOME BACK, ";
     NAME$;"!"
2220 PRINT :PRINT "YOU NOW HAVE ";
     GOLD;" IN GOLD";
2230 IF ANKH - 0 THEN PRINT ".":GOTO
     2260
2240 PRINT ","
2250 PRINT " AND A GOLDEN ANKH."
2260 PRINT :GOTO 100

5000 REM + + + INITIALIZE + + +

5010 REM ** VARIABLES ETC. **
5015 POKE 752,1:OPEN #1,4,0,"K:":DIM
     NAME$(5),FILE$(7),Q$(1):FILE$ - "D:"

5100 REM ** TITLE **
5110 PRINT "↖":REM THE ↖ IS SHIFT
     CLEAR
5120 POSITION 10,4
5130 PRINT "THE DUNGEON MACHINE"
5140 POSITION 12,7
5150 PRINT "(PRACTICE AREA)"
5180 POSITION PEEK(85),14
5190 GOTO 30

10000 REM + + + TO INSTALL JOE + + +
10020 POKE 1536,25:POKE 1537,250
10050 POKE 1538,ASC("J")
10060 POKE 1539,ASC("O")
10070 POKE 1540,ASC("E")
10080 POKE 1541,ASC(" ")
10090 POKE 1542,ASC(" ")
10100 POKE 1543,0
10110 POKE 1544,10
10120 POKE 1545,0
10130 OPEN #2,8,0,"D:JOE"
10140 FOR X - 1536 TO 1545
10150 PUT #2,PEEK(X)
10160 NEXT X
10170 PUT #2,155:REM END OF FILE
      CHARACTER
10180 CLOSE #2
```

FAIRGAME

Graphics: Nancy James

Do You Believe in Magic?

Sorcerer

By Steve Meretzky.

Most computers.

Sorcerer is the second part of Infocom's magical trilogy (no, not *Zork V*!). Much like the first program of the series, *Enchanter*, *Sorcerer* requires the player to become conversant in matters arcane. Magic is everywhere. Quick wits are rarely as useful as the casting of an appropriate spell. Before long, mere mortals are quite at ease blasting open doors with Rezrov spells, floating in the air courtesy of Izyuk, and eavesdropping on minds through Yomin. (After all, what kind of dreams do gnomes really have?) Even the future may be pierced via the Vezza oracle spell.

The plot: Guildmaster Belboz has been abducted by a powerful demon and must be rescued. The player must first determine how to magically leave the building within a certain time limit, or else suffer eternal torture at the hands of the demon.

The game is divided into a series of unconnected modules. Once you go from one setting to another, the doorway slams shut behind you forever. Save the game often, or you will be constantly restarting it to seek out something you did not find in a previous module and now need. Also, pay strict attention to subtleties of descriptions and events. Some puzzles are only solvable if these nuances are observed and their import correctly interpreted.

The world of the Sorcerer is rich in detail and wonderment. The magical experiences resemble the exotic adventures of Carlos Castaneda. Steve Meretzky, who also wrote the hilarious sci-fi adventure comedy, *Planetfall*, has crafted several diabolical traps and puzzles. Marvel at the three-dimensional, transparent, glass cube maze! Commit hara-kiri for points and profit! Make extra money parting the Red Sea and waxing floors! Speaking of red, red herrings abound in every puzzle. Often the false trails appear more logical than the actual solution path. Meretzky has really achieved considerable design subtlety.

One very clever new addition to the Infocom parser is the ability of the program to remember exactly the manner in which you performed an act. Later, the program will give you a replay of certain acts, but from a different perspective. This feat brings the concept of true interactive fiction one more step toward reality. "Interlogic Marches On!" intones the Movietone newsreel.

Plan on spending many delightful evenings and weekends playing *Sorcerer*. The final solution is like a delicate orchid achieving full bloom. Long after the game is over, the heady fragrance stays with you. RRA

Apple, Atari, Commodore 64, DEC, IBM, NEC, Osborne, TRS-80. $49.95 from Infocom, 55 Wheeler Street, Cambridge, MA 02138; (617) 492-1031.

Looting the Louvre

The Heist

By Mike Livesay and Mike Mooney.

Adam, Apple, Commodore 64.

Now wait a minute. Aren't games supposed to make us feel good? Aren't they supposed to provide a socially acceptable way to release pent-up tension? Aren't they supposed to let us heave a satisfying "Oh, yeah!" after we're done playing them? They are? Thought so.

If that's what games are supposed to be, then *The Heist* is the antithesis of what the game gods had in mind. *The Heist* is a fine game. Programmers Livesay and Mooney worked diligently on the game, and it shows. Animation is smooth; colors are as bright as they are plentiful; sound effects are kept to an undistracting minimum. In other words, everything works. Everything except for flow.

Graham Crackers is *The Heist*'s do-good hero. His job is to pick up every piece of artwork displayed in a museum, in a search for—are you ready?—a micro dot to save the world. Fortunately, the milquetoast plot isn't reflected in the game itself (thumbs down to the Micro Fun employee whose idea it was). The game was designed before the story was imposed, and it's more fun than the plot sounds.

The museum consists of three levels, each of which contains sixteen different screens, and each screen shows three rooms. Math whizzes will say the game has one hundred forty-four rooms, which it does, but some rooms are just slight variations of rooms on other levels. However, all sixteen screens are completely different. How can anyone complain about a game with "just" sixteen screens? There's no getting bored in *The Heist*.

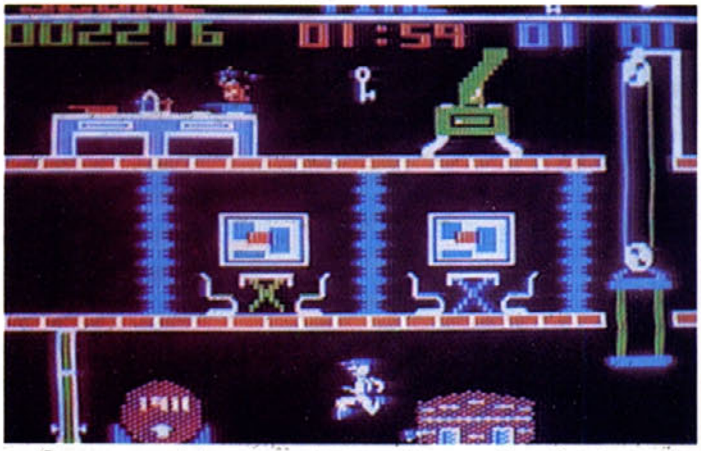
Livesay's last two works were the Apple and Coleco versions of the 1983 runaway hit, *Miner 2049er*. *The Heist* is different, but it's obvious that there's still a little *Miner* blood in Livesay's veins. It's a passive game; no one is chasing or attacking you. If Graham meets his demise, it will be his (the player's) own fault. Thematically, it's not too different from *Miner* (collect things, avoid things, and do it before time runs out), but they aren't by any means the same game.

The Heist is big. Three rooms take up the space where twelve would have fit in *Lode Runner* or *Hard Hat Mack* proportions. The plus is that conveyor belts, paintings, sculptures, plants, and dinette sets contain much more visual detail than would have been possible on a smaller scale. Paintings look just as weird as anything you'd find in a gallery of modern art. Even the furniture has that "museum" look.

The minus is that the game goes to the disk to change screens every time Graham Crackers gets to the end of a room, which takes about seven seconds from one side to the other.

Escalators, elevators, and gaps between floors are in strategic locations. To go from one room to the one above might require running several rooms across, two flights up, and one flight down. A map of the museum sure helps.

An aspect of *Miner* that had some players grumbling was that if they blew it they'd have to start the level over again, covering the same ground that had been covered before. *The Heist* doesn't do that; objects that have been collected don't have to be collected again after you lose a Graham. However, he does have to start over from the first room of the museum, which means he'll encounter the same hazards he managed to squeak by the last time around.



Components of the game are executed very well—animation, color, layout, and point schemes. What's missing from the game is a feeling of continuity. You *know* that the game takes place inside a museum consistinġ of sixteen three-room screens, but it feels more like sixteen different three-room museums. What a difference a few seconds between screens makes!

Nonetheless, *The Heist* is fun, captivating, challenging, and quite playable. Don't be fooled by how simple it looks. A lot of strategy and concentration is required to do even moderately well. MY

Adam, Apple, $40; Commodore 64, $35. Joystick required. Micro Fun, 2699 Skokie Valley Road, Highland Park, IL 60035; (312) 433-7550.

Lights! Camera! Action!

Movie Maker

By Eric Podietz and Guy Nouri.

Atari.

The screen darkens, a leader counts down from ten to one, and *Movie Maker*'s demo disk is off and running. Several short cartoons, each lasting about twenty seconds, exploit the utility's capabilities. By far the best sequence is a Chaplinesque saga with The Little Tramp gathering flowers for his ladylove.

Impressive stuff—even more so after examining the program itself. A microscopic, five-frame "movie" of a log falling down—accompanied by the descending pitch of a single sound track—took an hour to produce. An unbelievable amount of painstaking work must have gone into the creation of that demo.

Movie Maker—available later this spring for the Apple, IBM PC, and Commodore 64—is an enormously clever package that is, alas, completely impractical for the average user on two counts: (1) Most home systems have one disk drive, and while it's possible to manipulate the program with just one, the necessary disk swaps are tedious and will quickly wear out the hatchway springs. (2) Most users don't possess the artistic skills needed to produce smooth animated action.

Animation with *Movie Maker* works as it does in the Disney studios. Pages of drawn images, each slightly different from the last, are rapidly flipped (in this case, by the computer) to produce an illusion of movement. The user must first create the image and then modify it to produce a series of composite pictures. Even a simple goldfish is hard to animate cleanly; fin and tail movement must be coordinated with the ripple of the body.

Movie Maker does nothing to help build those intermediate images. Design is accomplished with a joystick in graphics 7, and smooth curves are particularly difficult to draw. Once the shapes are complete, they are put together into a sequence. At this point, special effects may be added: slow motion, zoom (image enlargement), or sound effects. A background is added. A final smoothing process removes the flicker, and the finished film is stored for subsequent viewing.

Movies can only be watched, however, with the *Movie Maker* disk. To add finished sequences to a program in progress, you'll need the professional version from Interactive Picture Systems.

The package comes with several shape pages, all of which were used in the creation of the demo programs. Practice sequences can be made with these shapes, but very little reward will come from close duplicates of preexisting cartoons. Eventually, users will wish to design their own shapes; that will be difficult without extensive knowledge of perspective, scaling, and anatomy.

A further limitation is the sound library. *Movie Maker* has four voices, each with eight noises; those thirty-two pops and whistles represent the total spectrum of available sound. As the instructions state, "It is not possible to make new (sound) files." Sixteen of the thirty-two are two eight-tone scales of differing pitch; the others are primitive explosions and putt-putt-putts.

Nor is color used with much cleverness. Because of the graphics 7 mode, each finished film is limited to four colors. The colors can be changed easily, which allows for shading effects that give the illusion of added colors, assuming that one understands the principles of shading.

The ninety-page instruction manual, though logically constructed and easy to read, occasionally omits important steps. Sound effects can't be used (or previewed) without first putting a shape file in memory, and the introductory example ignores a critical menu command. A left-handed adjustment is made by pressing escape, rather than select, as the book instructs.

Difficult as it is to dislike the idea behind *Movie Maker*, it's just as difficult to use the program. DB

Atari, 48K; disk. $60 from Interactive Picture Systems, Reston Publishing, 11480 Sunset Hills Road, Reston, VA 22090; (703) 437-8900.

Play Ball! A Preview

Digit-Ball

Apple, IBM.

Anyone who has played video games in the local arcade has probably noticed that the track ball used in games like *Centipede* and *Missile Command* gives more accurate control than the various buttons, levers, and joysticks used by other games. Even the Apple II's hi-res joystick can't really match a track ball for control. But while the industry biggies have yet to overcome the problems of putting a track ball in an Apple II, a newcomer already has.

Interfirm Systems Corporation of San Jose, California, has unleashed the Digit-Ball on an unsuspecting market, pulling off tricks that TG Products and Wico have so far been unable to perform. Wico's effort is a good solid track ball that works with no existing games. It only gives the Apple input through a peripheral slot, and without game I/O port support you might as well try to use your printer as a game controller. The TG track ball is—well, has anybody ever really *seen* a TG track ball?

In fact, technical experts and other arcade junkies at *St.Game* have seen a prototype of the Digit-Ball, and we are impressed. Rumor has it that representatives of Wico saw it in action at a computer show and were afflicted with white faces and shaky knees. But that's just rumor, and we deal in facts at *St.Game*, pretty much.

The fact is that the Digit-Ball is one impressive product. It plugs into a slot in the Apple, making it at least superficially similar to the Wico ball—similar in all the right ways. Wico uses the advantage of the slot to make the track ball moderately intelligent. That is, it offers a few different resolutions, electronic positioning, and other goodies. Digit-Ball uses its interface in similar ways. Where the two balls differ is the game I/O port question. The Digit-Ball has a strap from its interface card to the game port, so it will work with all present and future joystick-based games.

Anyone at all electronically oriented will appreciate this: A joystick takes the analog signal created by your hand moving the stick and sends it to the game I/O port, where the computer must translate it into digital terms before it can deal with it. The Digit-Ball interface card takes the analog signal from the ball, translates it into digital form, acts on it (performing autocentering, resolution, and wraparound transformations), and then converts it back to a pseudoanalog signal before sending it to the game I/O port. In other words, this track ball seems to operate on an analog-to-digital-to-analog-to-digital concept.

This might sound inefficient, but that's the price of joystick compatibility. Actually, it's no less efficient than regular joystick input, because all of the preprocessing is performed by the interface card, not the Apple's 6502. The Apple uses a delay timer to read the joystick, which inevitably slows the game down.

If a program is written specifically for the Digit-Ball, it can bypass the delay timer and read the ball considerably faster by looking directly at the interface card instead of the game I/O port. It's so simple to read the ball directly, in fact, that publishers of joystick games would be well advised to provide it as an option, the way they support the Mockingboard sound and speech synthesizers. The time saved could be used to speed up the game or give it fancier sound effects.

The main advantages of the Digit-Ball are the various custom settings it allows. A joystick can have selectable autocentering and slidewheels to control where the center is, but that's about it. Further, the more features of that kind it has, the more moving parts are involved, and the more moving parts, the greater the likelihood of a breakdown. The Digit-Ball has those abilities and more, but as they are controlled electronically rather than mechanically, they are likely to be more reliable in the long run.

Default settings for all features are set by DIP switches on the interface card. These settings determine how the ball will behave when the machine is first booted. For instance, the recommended setting makes the ball behave like a joystick with centering. This setting is useful for games like *Pac-Man* that only respond to movements of the stick away from the center. Autocentering mode is practically useless for programs in which a cursor on the screen follows the movements of the ball precisely; unless the ball is actually rolling, the position it returns is always the one defined as center (in order for the ball to be compatible with as many games as possible, the center can be set to any point in the 256-by-256 range).

With autocentering, the ball position is always "springing" back to the center. The Digit-Ball also offers software centering, in which the position only centers when a program tells it to. For situations where autocentering is inappropriate, it can be turned off entirely.

There are a lot of other options available for customizing the Digit-Ball's behavior. Some may be used frequently; others you may never use at all. For instance, it has seven different "resolutions," with ratios ranging from 1:1 to 64:1. This means that you can have each turn of the ball translate to either a small motion on the screen, or to a larger one. The lower-resolution setting simulates very fast spinning of the ball, which is good for some arcade games. The higher resolution allows the fine control that is useful for graphics design or mouse-type applications. Resolution is set with a combination of switches and jumpers on the interface card.

Since the screen has a finite number of positions, there is a way to have the ball react intelligently when the tracking position reaches an "edge." There are two options: It can stick to the edge position, or it can wrap around—that is, jump to the opposite position. However, since there are four edges, the Digit-Ball has independent settings for each edge. This is an example of an option you may never use, but if you ever want to, you'll be glad it's there.

Another option is freezing an axis. This allows you to draw perfect horizontal or vertical lines if you're using a drawing program. A related option will be appreciated by left-handers. Normally, the buttons are on the side of the ball that would be most comfortable for a right-handed person. Turn it around and a left-hander is comfortable with it, except that it moves the ship/cursor/whatever in the wrong direction. Change a jumper on the interface card and it's a true left-handed track ball.

The Digit-Ball is a sturdy device. It seems to be well put together and capable of surviving normally violent game-playing activity. (However, bashing it on the table when you miss the high score by five points is considered to be *abnormally* violent, in case there was any doubt.) DD

Apple, IBM. $99 from Interfirm Systems, 1899 Montford Court, San Jose, CA 95132; (408) 923-3338.

Dragons 3, Threads 0

Dragonriders of Pern

By The Connally Group.

Atari, Commodore 64.

As in the series of novels by Anne McCaffrey on which it is based, *Dragonriders of Pern* re-creates a climactic encounter between Man and Nature.

For two generations, the people of the planet Pern have paid little attention to Red Star, a planet with an irregular orbit—an orbit that occasionally enters Pern's own. Now, however, the life forms on Red Star have come to desire the more hospitable climate of Pern and are seeking to bridge the gap that separates the two planets. Whenever Red Star comes close, silver Threads rain down through Pern's atmosphere upon the unsuspecting population.

These aren't ordinary threads. They're Threads. Threads devour all organic matter that gets in their way. That includes plants, animals, and people. In *Dragonriders*, the object is to assure the people of Pern that you can effectively battle the deadly Threads.

This game reveals itself not only as a faithful rendition of Anne McCaffrey's trilogy, but also as a marvelous strategy and arcade game. As many as four people can play, each responsible for a different Weyr (battlestation), which protects the colonies.

As Weyrleader, you must forge alliances with Hold (colony) leaders and craftmasters who can aid you in battling the Threads. Here's where good judgment of character comes in handy. Delicate negotiation with a Hold leader may work, but sometimes a direct approach is a better bet. Sometimes inviting a leader to a dragon hatching will put you in the leader's favor; sometimes inviting one to a wedding will score points. Politics is involved, and playing the social game is important.

The ultimate threat of falling Threads is impossible to ignore. The name of the Hold and the Weyr sworn to protect it are revealed onscreen, and the first player to push the fire button becomes committed to that Hold's defense. Now it's fightin' time. Thread fighting is executed in arcade fashion, with dragons responding to players' joysticks. The arcade action is engrossing and adds much to the richness of what is otherwise a strategy game.

Thread fighting is vital to your success as a Weyrmaster. The Thread attack adjusts automatically to your expertise as the game progresses, becoming more challenging each time.

Dragonriders of Pern is much more than entertainment—it's an experience. Once play is under way, hours pass with little notice. Outstanding programming skills have brought the excitement of McCaffrey's fantasy adventures to the computer. This one's destined for fame, as are those who become successful Weyrleaders. HGL

Atari, 32K, Commodore 64 disk. $40 from Epyx, 1043 Kiel Court, Sunnyvale, CA 94089; (408) 745-0700.

Call Him Dr. Joystick

Julius Erving and Larry Bird Go One-on-One

By Eric Hammond, Julius Erving, and Larry Bird.

Apple, Commodore 64.

This is *the* sports game of 1984.

Apples have seen their share of computerized bowling, race car driving, football, baseball, horse racing, skiiing, decathlon, and who-knows-how-many golf simulations. Some were good, some were bad. Trouble is, most of them looked and felt like computer games.

Put 'em away. Here's *One-on-One*, a game that feels like the real thing.

It's Julius Erving against Larry Bird—two of basketball's finest. One person can play against a computer-controlled Julius Erving or Larry Bird, or two people can play, each controlling a different player. Big deal, right? Right.

The big deal is all the stuff Eric Hammond put into *One-on-One*, with some advice from the game's title characters. Slam dunks, finger rolls, fade-away jumpers, instant replay, fouls, shattered backboards. They're all there.

Erving's and Bird's respective strengths and weaknesses are what take an already fantastic game and turn it into something excitingly real. For example, Bird, the more powerful of the two, is able to bully Erving under the net more, while Erving's slick moves often leave Bird in the dust. Bird's a better outside shooter than Erving, but it's hard to tell unless you're playing against a human opponent. Fatigue, hot streaks, and cold spells also enter into the game.

Although each player's shooting percentages are included in *One-on-One*, the program also checks out a number of things when calculating if the shot should go in: How close is the defensive player? Are you releasing the ball at the apex of your jump? Were you shooting on the run?

Four levels of play (park and rec, varsity, college, and pro) make it easy to learn the subtleties of the game gradually while enjoying victory. Or defeat. Play the pro level for a challenge; go back to park and rec if you feel like wiping out a computer opponent.

Even nonathletes have a fair chance at beating the computer or another player, but knowing a little about real-life basketball doesn't hurt. Remember, the computer knows almost as much about basketball as does Hammond, a formidable player himself.

Playing against another person produces a fairer game than playing against the computer. For one thing, the computer knows the optimal time to release the ball; on defense, it moves when you move; and at the college and pro levels it steals the ball easily, leaving you disgruntled after having made a great defensive rebound.

On the other hand, it's not too hard to screen the computer's player after it shoots, leaving the loose ball yours for the taking. Running out the clock (if you're leading or if you want to reduce your player's fatigue) is a cinch; the computer doesn't know enough to come after you and make you play.

A sorely missed feature is the real-life ability to stop dribbling and hold the ball; the only time you can stop dribbling is when you're jumping to take a shot. Because of this, it's not possible to fake the opponent into going the wrong way while you drive for the hoop.

But we'll let Hammond off the hook for that. It would be like criticizing George Lucas for not filming *Return of the Jedi* on location in outer space.

The weird thing is that after a few rounds with *One-on-One*, you'll forget about the computer and really feel like you're on the court with Dr. J or Mr. Bird. Don't be surprised if you get the urge to go out and shoot some jumpers in the driveway after you've finished playing. No lie.

Highly addictive, thrilling, sometimes frustrating, and always involving are ways to describe *One-on-One*. It's a winner. MY

Apple, Commodore 64, disk. One or two joysticks required. $40 from Electronic Arts, 2755 Campus Drive, San Mateo, CA 94403; (415) 571-7171.

No Restaurant at the End

Universe

By Thomas R. Carbone and William G.M. Leslie III

Atari.

16,000,000,000 years ago . . . the universe begins
4,000,000 years ago . . . man emerges
380 years ago . . . the USSC *Gerardus* enters hyperspace
320 years ago . . . the hyperspace booster is discovered
221 years ago . . . Axia is colonized
4 months ago . . . communication with Earth is mysteriously broken
15 days ago . . . evidence of a second hyperspace booster is found
Now . . . the need for a hero has never been greater

That dramatic introduction scrolling up a darkened screen, accon panied by Brian D. Swan's pulsating electronic prelude, sets the stage for Omnitrend's *Universe*.

This four-disk package is a complex space simulation with more detail than most players could imagine. Every facet of space exploration is covered thoroughly, from asteroid mining to orbital piracy. The game does not just provide methods for space captains to earn their first million credits; additional intrigue comes with the unexpected loss of contact with home-planet Earth.

Nowhere is it written, however, that a simulation should take the same time expected of its real-life counterpart. (Who would want to play an eight-hour game that simulated the process of sleeping?) Although *Universe* is fascinating for its intricacy, a lethargic running speed and endless disk swaps make it very, *very* slow going.

Universe is packaged in a big three-ring binder that contains an eighty-page instruction manual and several reference charts. The instructions are clearly written and intelligently organized, but the volume of information makes quick comprehension a feat of truly superhuman understanding. Neophyte captains should plan on several false starts and at least one wasted evening.

The hero-to-be begins in the Central Bank of Axia, in search of a loan (this game doesn't omit *anything*). After establishing a payback schedule, the player receives a standard "merchant's loan package," which includes the bare essentials necessary to buy and operate a starship. The ship itself may be one of ten styles, each uniquely fast, large, or expensive.

The next step is to load all that new equipment. Each section of the ship must be stuffed with hulls; the player starts with thirty-one hulls, exactly enough to contain the initial supply of parts. More hulls must be purchased to make room for additional equipment.

Aside from the machinery necessary to operate the ship, other parts include an orbital shuttle (for hops between the ship and various planets), a cryogenic vault (for the transport of passengers), crew quarters, and a supply of fuel. Immediately after stocking the ship, the player must purchase provisions and find a crew (a minimum of fifteen is suggested, ten to man the ship and five to control the shuttle). Attempting to do anything without a crew immediately terminates the game.

Preparation time can take up to an hour. If anything is done out of sequence, the player must start all over and suffer through another round of lengthy introduction.

Some of the remaining credits should be spent on Ore IV, which is used as fuel. The fifteen thousand units of ore included in the loan package will not suffice for even one warp jump to the nearest star.

Having stocked his ship, the new captain will find himself in a standard orbit around the planet Axia of the star system Hope. As in real life, everything revolves around the mighty buck (or in this case, credit), so the first requirement is a source of income.

There are several such sources, although only two are practical for new captains: passenger transport and trading. The former is performed easily; emigrants are ferried from a planet with surplus population to one that needs folks. Potential travelers must, however, have a sophistication level (rated on a scale from 1 to 100) of at least 1 but no more than 10 higher than the planet to which they are taken.

By a happy coincidence, the nearest star (Prisa) has a planet, Zeath, which is four sophistication points lower than Axia. Each emigrant from Axia to Zeath is worth roughly two thousand credits. It doesn't work the other way around, though; passengers may not be taken from Zeath to Axia. Another planet, one with an even lower sophistication level, must be found for Zeath's eager travelers.

Although the ship's cryogenic vault can store up to one hundred people, the shuttle only holds three at a time. (Due to a programming glitch, however, that same shuttle *can* take more than three down from the ship to a planet surface.) "No problem," thinks the fledgling captain. "Thirty-three quick shuttle trips from starport to ship will have that vault filled in no time."

The operative word, alas, is "quick," and a shuttle trip is anything but. A one-way journey takes between three and five minutes (real time), during which the only action on the screen is the boring altitude indicator: "315 KM . . . 307 . . . 298 . . . 297. . . ." Ten-minute round trips make the notion of thirty-three hops rather unsettling. Some shuttles have larger capacities, but (of course) they cost more.

Trading is also accomplished at a starport. This is the game's best—and fastest—feature. The "Buyer's Board" resembles a huge stock exchange: Hundreds of weirdly named commodities zip in and out of ten slots while the captain blinks in confusion. Selections are made by hitting the corresponding keyboard number, but be sharp: While the eyes drop from the screen to the keyboard, that 500-credit lot of Indrusegg may have been replaced with a 200,000-credit lot of Qui-bieh Kiff-dro.

Commodities are not identified at the time of purchase and can be classified only if the ship computer possesses the Marketing Program (not included in the basic package). At 1,500 credits it's a wise investment, because all commodities have unusual names; it would be difficult to guess whether *baefiscz* is food, medicine, or an alien life form.

Other forms of income include mining and piracy. Both require considerable equipment not included in the loan package, so they are not immediately practical. Mining is accomplished on any suitable planet, although hostile natives first must be "persuaded" to move. Ore obtained from the mining operation can be taken to a starport and converted into credits. Piracy involves boarding and taking control of another spaceship. If the skirmish is successful, the ship's contents are awarded to the marauder.

Journeying to another star system involves several time-consuming steps, all of which must be performed in the proper order. After shuttling from a starport back to the ship, the captain must return to a standard orbit of at least 20,000 km. (Trying to warp out in a lower orbit will damage the warp engines.) The computer then sets a course and displays the energy needed for the trip. (That fuel requirement is given in Energy Units rather than Ore IV; the instructions do not explain the correspondence between the two, and it's not one to one.)

If the computer finds enough Ore IV for the trip, the ship enters hyperspace and zooms to the chosen star system. After leaving warp space, a solar scan finds the desired planet and charts a course for it (more energy required). Warp travel is not precise, so this portion of the trip varies; you might land directly adjacent to the target, or halfway across the solar system. After establishing a standard orbit, descent to a lower orbit for shuttle access completes the trip.

Total time required here is about half an hour. Much of that is wasted while slow graphics routines tediously draw pictures that have no bearing on the game. Also aggravating are the numerous disk swaps needed to perform any operation. Every individual command involves transferring information from a game disk to the player disk. The instructions contain no reference to using a second disk drive; that option should be first on Omnitrend's list of improvements.

Universe can be saved and reloaded at a later date. Since the entire player disk is filled with information, a game cannot be saved more than once per disk. A backup copy (which would eliminate the need to start from scratch in case of a disaster) can be made at the same time, but the process takes more than five disk swaps. The DOS function "Dup Disk" performs the job much faster.

The only other drawback is the rather capricious manner in which death can strike. A routine shuttle flight might end in tragedy when the craft is destroyed; this random element is uncontrollable and results in much anguish, considering the time invested. It is therefore advisable to copy that player disk every day or so.

The concept, imagination, and detail behind *Universe* are superb. Were it not for the slow speed, it would be a perfect game. As it is, the time factor might make it inaccessible to those with mundane requirements in their lives—such as working, eating, and sleeping. DB

Atari, 48K; disk. $89.95 from Omnitrend Software, Box 3, West Simsbury, CT 06092; (203) 658-2944.

Snug in De Rug

D-Bug

By Child Ware.

Atari.

Big fleas have little fleas upon their backs to bite 'em;
And little fleas have littler fleas, and so on infinitum.

—Robert A. Heinlein

Definition of the word "feature": a bug as described by the marketing department.

—*Apple II Reference Manual*

A game within a game, *D-Bug* begins on a page representing the various parts of a computer with RAM, graphics, sound, and CPU modules neatly compartmentalized. Even the I/O port and power supply (complete with rotating fan) are represented.

Here you may select your player options: one or two players, bug or no-bug mode, skill level, and your animated game character—Moe Dem, Dot Matrix, or M.A.X. the Mechanized Antibug eXpert.

Once you've made your selections, you may either take a tour of the system or go straight to the game—Gotcha!

Gotcha is a deceptively simple-looking strategy game composed of a screen that displays rows of two distinctly different shapes, butterflies and sailboats, randomly scrambled. The object of the game is to "collect" more of your objects than your opponent can of his.

By selectively collecting only those pieces that force the cursor into an area surrounded by your men, you can force your opponent into picking up your pieces, thus giving you the points.

When no more pieces can be picked up within the cursor range, the game is finished and the player with the most points wins.

Simple, right?

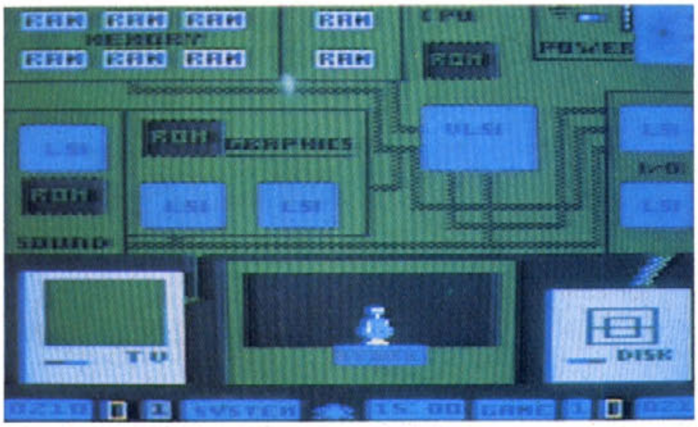
That's not why they call it *D-Bug*.

Just when you really get into it and think you've mastered the strategy, all hell breaks loose!

The screen breaks up, or flashes, or fills with random garbage. The musical game accompaniment crackles with static. Boot errors occur. Glitches of all description crash the system. Just like the real thing!

So it's back to the system page, climb out onto the screen, and try to hunt down those pesky bugs.

Running your man across the screen to the problem module causes him to flash brightly—on the lower skill levels only; after that you're on your own. Press the fire button and, like Tron, you enter the nerve complexes of the "computer."



Each uniquely individual module logically depicts its true function, in both graphics and symbols.

The graphics module has that distinctive Atari rainbow rippling down across its background. Game pieces can be selected here and will later appear in Gotcha.

Scrolling smoothly in and out, I/O's background of arrows visually represents the flow of information through the computer. The sound module has musical notes that will alter the game sounds when selected. RAM's got binary bits updating at high speeds in symbolic bytes. The power supply has a fuse and the occasional wiring short.

Perhaps the most interesting and informative module in the system is the CPU itself, with its active logic gates switching away in their own logical fashion.

Now to find the bug.

The modules usually have chips (power supply notwithstanding)—either RAM, ROM, or specialized ones that often go "flaky," producing the glitch. Again your little man goes running around the module looking for the bug, and once again—on the lower two skill levels—he flashes when he is standing over the fault.

Often simply "stomping" on the chip will be enough to fix it. If the chip does need replacing, for a nominal price (paid in points), there is a computer store at the bottom of the screen where parts and services (including those of a costly repairman) are readily available.

Of course, time to find and debug the problem costs you points, and there is a time limit.

GOTO page 51

WHAT ARE YOU DOING? STAY AWAY FROM THOSE CONTROLS!
TRICI VENOLA

THE STRATEGY SCENE

Then & Now... by William Harrington

Nowhere has the personal computer had a greater impact than in the way we play games. When the books are balanced, the financial statements made, and the reports written, the computer is ready to play another role, every bit as important, as an imagination machine. For centuries, games of chance and strategy have provided entertainment for both kings and paupers. Yet while our games may have changed over the years from the Sennet of the ancient Egyptians to Monopoly, the style and content of gaming have changed little—until now. In the last few years, the introduction of personal computers into our homes, schools, and businesses has created a true revolution. This is the first machine that expands not only our strength or our senses, but our minds as well.

But You Can't Fold Them Up. Unlike their board game ancestors, many computer games provide significant extras. Games can often be saved in progress, without the worry of finding a place secure from curious pets and the like. Some games can "kibitz" on occasion and suggest a move to a perplexed player, while the more forgiving variety will even allow you to take back an injudicious move—something Uncle Harry would never have allowed! And last but not least, the personal computer is programmable, unlike dedicated game machines that play only one game. Personal computers can play anything that the will and whim of their owners dictate. And they promise to remain as current as the best efforts of a legion of talented programming types can ensure.

The ability to provide a ready opponent, and a challenging one at that, would be reason enough for many avid gamers to buy a computer. Beyond this, however, the continuing quest to give computerized opponents enhanced capabilities places the game player at the very forefront of the quest for "artificial intelligence," that Holy Grail of computer scientists. Some computers, it seems, take their games very seriously.

The man-machine battle is already over in that centuries-old game of chance and strategy, backgammon. A few years ago, the reigning human backgammon champion unceremoniously lost his crown to a computer. From all accounts, the gentleman took the loss pretty hard; the computer was unavailable for comment.

It is in the ancient and venerable game of chess where the battle rages on. Since the very beginning of the modern computer age in the late 1940s, the creation of a competitive chess-playing machine has been a dream shared by computer scientists and chess players alike. This modern "John Henry" match-up of human versus machine intellect goes on to this day.

You might think that only supercomputers such as the dual processor Cray, running state-of-the-art chess programs, would be competitive in this rarefied world. After all, no machine or program has yet closed in on the Grand Master level of chess. Indeed, the universe probably won't be around long enough for even the fastest computer to fully exhaust all the possibilities in one chess game. Somehow, the human mind seems to analyze a chess game spatially, rapidly discounting certain possibilities while pursuing a promising line of attack to great depth. While some chess programs attempt to emulate this "intuitive" style of play, the brute-force approach has been the most prevalent—and the most successful.

The level of chess played on today's personal computers is rapidly approaching tournament class. The Sargon family of chess programs by Kathy and Dan Spracklin, culminating in *Sargon III*, has been joined by Larry Atkins's *Chess 7.0* as the premier microcomputer-based chess programs available. Recently, playing on an accelerated Apple II, *Sargon III* was entered in a chess tournament, and for the first time a microcomputer chess program defeated a rated Chess Master in tournament play. At this writing, another chess master, with equally strong programming credentials, is developing a microcomputer-based chess program that he believes will finally go on to beat a Grand Master. Who knows, the reigning Russian chess champions may soon face a tougher American competitor then Bobby Fischer for the world championship; the next American challenger may be a computer!

Old Players Never Die. Just as personal computers have brought new and exciting capabilities to ancient board games, so modern twentieth century games of strategy and tactics have found a place in the new medium.

Modern board gaming began in the 1950s with the introduction of Tactics, a classic game of contemporary land army tactics, now twenty-five years old. These early games by a young, upstart company, Avalon Hill, and a growing host of competitors, soon created a considerable industry and not a few sleepless nights for those who eventually became hooked. Modeled after the professional games long used by military planners, Avalon Hill's games had three key distinctions: They were based initially on historical battles or campaigns, were rigorously accurate, and, due to the economies of mass production, were easily affordable. (Why should the Joint Chiefs of Staff have all the fun?) Essentially, games of conflict resolution (war games) are mathematical models of all the variables inherent in a particular conflict. While the outcome of any particular game may vary greatly from the original battle or conflict, the result must be valid—that is, based on all known facts, it *could* have happened that way.

Historical board games offer the unique ability to play and replay the great conflicts and battles of history. The haunting what-ifs of history serve as a dramatic backdrop for games that suspend the present and transport the player to other places and other times. These sometimes highly sophisticated simulations combine chance and skill in a demanding mental contest that often consumes entire weekends. Ranging from simulated military conflicts to games of political and economic competition, such as Diplomacy, they provide players with mind-stretching excitement and rare insight into the personalities and conflicts that have shaped the past and may well shape our future. However, a particularly warlike disposition isn't necessary to appreciate the excitement and intellectual challenge of tactical and strategic command.

Games that attempt to model or simulate actual conflicts have long been used by strategic planners after gaining wide acceptance in the German state of Prussia during the Napoleonic Era. Perhaps the most famous example of modern war gaming occurred in 1942 when the Japanese navy commissioned a game to evaluate the outcome of a proposed attack on the American naval forces guarding the vital American base at Midway Island. To the mounting shock of the admiral assigned to referee the game, three of the emperor's aircraft carriers were quickly sunk by the "American" side. The referee promptly disallowed the disquieting results, but a few months and four Japanese carriers later, the game was vindicated; the Japanese navy had suffered a humiliating defeat.

The Postwar Game World. With the creation of affordable, highly detailed historic board games in the 1950s, a whole new class of challenging and popular board games entered the entertainment market. In these manual game systems, a hexagonal map board was usually supplied depicting things like variable terrain, roads, towns, and key areas. (The virtue of the hexagon, by the way, is that it can conveniently provide a reference system while allowing the maximum degree of movement and positioning of individual units.) In addition, the number of variables concerned with unit movement and tactical positioning is manageable and understandable by the player or players. Precut cardboard counters were used to record individual units, their location, movement, and status. To further add to the realism of the simulation, "nature" or "chance" was represented by a pair of dice. This basic form of the modern recreational war game has remained unchanged since the 1950s. Today, conflict simulations enjoy wide appeal, with a score of manufacturers supporting tens of thousands of historical board gamers.

In exchange for all this realism and excitement, players must face considerable complexity and have more than a little patience. Needless to say, the rules and regulations for this kind of gaming are both detailed and extensive. Becoming familiar with the rules of a game can be a major undertaking, and of course the rules vary from game to game. Fur-

Illustration: Pieter Breugel the Elder, Jean Clouet, Trici Venola

thermore, the opportunity for an unintentional error is great. More than a few friendships have suffered undue strain due to an innocent error on someone's part. Suspending the play of a manual board game presents its own set of problems. Having discovered that your spouse has just vacuumed under the bed where your game board was "safely" hidden can reduce even the strongest among us to tears!

Look! Up on the Screen! What was needed, of course, was an electronic Moses to lead us out of the wilderness of manual board gaming. Enter the microcomputer. The first microcomputer war game, which adhered to traditional board game conventions, was *Computer Bismarck*, from a then-unknown company called Strategic Simulations, or SSI, as it came to be known. *Computer Bismarck* was based on the classic hunt for the World War II German battleship *Bismarck* by the British navy. This new computer-based game was a revelation. Gone was the traditional map board. In its place was a hi-res color screen of the North Atlantic. The computer served as the perfect game administrator; only "legal" moves were allowed, making game play infinitely more enjoyable. In addition, a save-game feature allowed a game to be saved in progress and reconstructed in a matter of seconds. And perhaps most revolutionary of all, provision was made for solitaire play, with the computer taking the role of opponent. The personal computer had discovered the personal war game. The most innovative feature of this and subsequent computer war games was the implementation of that most elusive concept, the fog of war.

Long before the advent of computers (or war games, for that matter), military strategists had recognized that war and the practice thereof constituted something less than an exact science. The unknown factors that prey on the mind of every commander are sometimes referred to as "the fog of war." The leader of any military force, whether commanding a naval task force, an army, or a simple infantry squad, faces imponderable dilemmas that no amount of training or planning can resolve. Not only are the true intentions and capabilities of the opponent frequently unknown, but the commander's knowledge of the state of his own forces may be imperfect. At any one time the disposition of friendly forces, their status and condition are known only in some vague, often dated sense. Communications, when available, may be garbled, intelligence estimates wildly inaccurate, and the command and control of dispersed forces less than perfect. And beyond the human element lies blind chance. The vagaries of weather or a stray bullet killing a key commander represent factors that can make hash of even the most intricate battleplan.

Until recently, war game systems, both the professional and recreational varieties, could only simulate in the most crude fashion the fog of war. As we pointed out earlier, the use of dice to simulate the random nature of chance in combat has long been a part of military-type games. Traditional manual systems usually imply a map board with all participating units readily visible to participants—hardly a likely condition in a real battle. With the introduction of the computer, the missing elements of confusion and chance could be simulated without burdening game play or administration. The computer could easily conceal enemy dispositions, calculate weather, cloud cover, and so on, and in general simulate the feel of actual command. It may be that the computer has at last emerged as the medium of choice for historical board games.

The Never-Ending Battle. The development of sophisticated solitaire play with the computer as opponent has not ceased. One serious impediment, however, is the lack of a simple means to evaluate the sophistication of computer players. Unlike chess or backgammon, which have a consistent set of rules and a game board, historical board games vary wildly. There simply is no good way to compare or evaluate the play of two computer opponents using different game systems. Here, extensive play testing by human players of proven capability is the only means of establishing and judging the capability of a computer opponent.

Today, historical board gaming is dominated by Avalon Hill and SSI. Game systems have become generally more sophisticated, and computer game authors have attained some of the popular acclaim that formerly belonged to manual game designers. The settings of these modern computer simulations range from Avalon Hill's *Legionnaire*, a real-time simulation of ancient tactics in Gallic-Roman times, to SSI's *Cosmic Balance*, a game of far-future intergalactic empire building. While SSI has, with some exceptions, stayed true to its all-computer game format, Avalon Hill has experimented with both all-computer and hybrid formats. For instance, Avalon Hill's *Close Assault*, a computerized version of the popular *Squad Leader*, employs a sophisticated computer opponent and administrator with a manual map board and counters—the best, perhaps, of both worlds.

The latest development in computer war gaming is the introduction of so-called "monster" games, such as SSI's acclaimed *Guadalcanal Campaign*, that take anywhere from eighty hours or more to play and include every ship, plane, and unit known to have engaged in an entire campaign. Even the Joint Chiefs of Staff don't put in those kinds of hours!

The present state of both traditional and modern board games is dynamic, to say the least. The continuing competition among game manufacturers ensures that the personal computer will have at least as significant an impact on our leisure time as on our professional lives. And for the future, the new sixteen- and thirty-two-bit computers from IBM and Apple offer some exciting possibilities. With faster processors and more memory, graphics should improve considerably. The mouse, that captivating little pointing device, seems an ideal game instrument. And of course anyone who has tried the user interface of Apple's Lisa and Macintosh can only wonder what games with "windows" would be like.

All us Walter Mitty types, our imaginations running wild while staring from our office windows, have finally found what we've been looking for—an imagination machine. At last!

William Harrington has been, at various times, a war correspondent, a social worker, and manager of a retail computer store. Prior to his current employment as director of product training with Hayden Software, he was employed by a research and development firm in Cambridge, Massachusetts, where, among other things, he worked with strategic war game systems.

...And Later On by Steve Jackson

Twenty years ago, if you had discussed the idea of computer games with a programmer (or anyone else knowledgeable in the field), you would have gotten a discussion of chess—and some speculations about the highly advanced strategy games that computers might offer by the 1980s.

That hypothetical 1960s programmer would no doubt have been utterly amazed by a few of the developments we've seen since then. Some would have delighted him—the personal computer, for instance. Who could have dreamed, twenty years ago, that devices with this much speed and power would be common in *homes*? Today we take them for granted. In 1960, they would have been science fiction.

But our '60s programmer, if we brought him to 1984 and showed him the sights, would probably be crushed to see some of the things we do with our wonderful computers.

Granted, arcade games are fun. But they use the computer's processing capacity, not to challenge our intellects, but to show us pretty pictures. That's why a lot of us are so pleased to see that, at long last, strategy games are getting the attention they deserve from programmers and buyers.

Generally speaking, a strategy game is any game in which the player controls forces in opposition to those controlled by another player, and where the player's intelligence is more important than his reflexes. A strategy game may be abstract, like chess, or a simulation of reality, like most war games. Or, like many fantasy and science-fiction games, it may be somewhere in between. A strategy game is not a "puzzle," though it will usually include elements of problem solving. This article will draw its examples primarily from the war game field, but its predictions apply to games on any subject.

The Computer as Referee. This is a function common to all game programs. By definition, the computer "knows" the rules of the game and applies them impartially. Whether your opponent is another human being or the computer program itself, part of any game program keeps track of the situation. It tells you where the units are (assuming you are entitled to know); informs you of each unit's current status; prompts you for orders for each unit; assesses each order given to determine whether it is legal (accepting only legal orders and correcting you when you attempt an illegal action); allocates new resources (raw materials, reinforcements, or whatever); refers to charts and tables to determine the correct outcome of some actions and randomizes the results of others; and determines the final interactive outcome of all moves by both players before displaying the status for the next move. Perhaps you've never given much thought to just how much referee work a good game program must do, but you've almost certainly cursed the failings of a poor program.

So what can we expect from the next generation of robotic referees? Quite a bit, really. For instance:

More complexity. That is, more variety in every phase of the game. The future should bring us faster hardware with more memory. That will free designers from the awful tradeoff decisions. More speed? Better graphics? Smarter strategies? More hard data? They can have them all.

For one thing, more facts can be presented. Whereas today a military unit might be described by three or four factors, tomorrow's games will involve a dozen. Of course, some players don't want to deal with rations, ammunition supply, and the quality of leadership; they would rather stick with the tried-and-true factors—strength, unit type, and morale. Fine. For them, we have a variety of *results*. There are dozens of things that can happen when a unit is attacked. We won't have to limit ourselves to forced retreats and numerical losses. We can have berserk counterattacks; lost leaders who force a healthy unit to hold in place until command can be reestablished; surrenders (with the administrative problems that all those prisoners will cause the enemy); abandonment of equipment during retreat; improvement of unit quality due to combat experience . . . and much more. All in one game. And the beauty of it is that the player doesn't need to worry about it. The referee, his faithful computer, is rolling all the dice and looking at all the tables for him.

Likewise, players will be permitted a much greater variety of options. These may take the form of bargaining strategies in a political game, physical actions in an adventure, or unit orders in a war game. Instead of "Move up two hexes and attack," a typical order might be "Move cautiously north, sending out scouts. Dig in and hold if light resistance is met, but withdraw immediately if confronted by a clearly superior force. If you travel three miles without meeting resistance, dig in and hold at the next stream crossing you encounter."

More simplicity. No, this isn't a contradiction. As hardware changes to allow for more game complexity, it is to be expected (and devoutly hoped) that designers will learn more about user-friendliness. Of course, hardware can help here: The mouse is a perfect example. But creative programming, with icons, windows, and whatnot, will take gaming farther than the hardware will. Again, the object is transparency. The game may be complex, but players won't have to worry about the details. They just play.

More speed. Already, some strategy games allow only a limited time for decision, blurring the distinction between "strategy" and "arcade" formats. *Legionnaire* comes to mind. This trend will continue, and in time we will see games that are true real-time simulations of battlefield or diplomatic situations. Using the information they would have in a real-life situation, players will have only as much time to make a decision as the real situation would allow. In fact, "game" may be the wrong word for these programs; they will be used for training and competency tests by businesses and the military.

The Computer as Opponent. Most modern strategy programs allow the user to "play against the computer." Different levels of difficulty are frequently offered, usually controlled by changing play balance: At higher difficulty levels the computer has more units, "fires" to greater effect, and so on. Only in chess programs is the user likely to find an opponent that plays more or less *skillfully* as the difficulty of the game changes. This is because chess algorithms are highly developed, allowing the computer to "look ahead" a greater or lesser number of moves. The more you allow the computer to look ahead, the more skillfully it plays. The future should see similar developments for other games. Imagine a game of *Eastern Front* where, using the exact same setup and rules, you have your choice of facing a brilliant, good, average, or incompetent Russian general.

A further development would allow players to shape their own opponent by modifying the decision algorithms that the computer uses to determine its moves. Can you teach a program to play so well that you can't beat it? We have already seen one small step toward this idea in *Robotwar*, which allows players to design their own combat robot programs and set them on one another.

The Computer as Mapboard. It almost goes without saying that increased speed and memory will lead to bigger and better map displays. The day is not far off when a designer will be able to spread his game map over as many screens as he likes and jump back and forth between them at will. Players will be able to see the whole of Europe at once, with points of light picking out the front lines; zero in on the Fifth Army's section of that front; and then pick out a single sector of a few square miles.

Graphics will also be improved, the better to serve players. Here is where the years of arcade-gaming zaps and flashes will pay off. Never again will computer map displays look like copies of paper gameboards, with smudgy counters moved by an invisible hand. Shells will zip across the battlefield and explode on their targets; armies will march across the desert, raising clouds of dust; stricken ships will catch fire, list, and finally sink in a cloud of steam. It will all seem *real*.

The Computer as Facilitator. A *facilitator* is someone who makes things happen. By itself, a computer can be a good opponent. Add a modem, and the computer becomes a window to a world full of opponents (or allies), all waiting for a game. And it remains the perfect referee—error-free and utterly fair—for all those games.

Today, some services let users phone in to a mainframe and pick a variety of game services. Using Delphi, Plato, GameMaster, or InfoNet, a player can challenge the computer at a variety of games or play against one of the other "live" users. Systems like this are sure to become more numerous and more sophisticated with each passing year. In time, groups of players from all over the country will be able to get together—taking the parts of World War I doughboys, World War II fighter pilots, World War III generals, or present-day negotiators in a geopolitical simulation—against similar groups of foes or against a very smart mainframe.

And just as some play-by-modem services currently sponsor competitions for high arcade scores, the services of the future will allow players to compete for the best performance in *Ogre*, *Eastern Front*, or *Midway*. In fact, there will be two types of tournaments—those against computers and live ones involving human opponents. Face-to-face game tournaments are likely to be largely replaced by modem tourneys in which players will challenge opponents thousands of miles away—without leaving home.

Summing Up. One last prediction. The developments outlined above seem logical, so they're probably *very* conservative. Remember how surprised our 1960s computerman was to see home computers, laser disk animation, and arcade games? The computer scene in the year 2000 is likely to be just as surprising. Hang on; it's going to be an interesting ride.

Steve Jackson works, perhaps inevitably, at Steve Jackson Games in Austin, Texas, where he publishes Space Gamer, Fantasy Gamer, *and* Fire and Movement, *and takes Fridays off whenever he feels like it. Lately, he's been working with Andy Greenberg and Lord British on bringing his games* Ogre *and* Car Wars *to the home computer.*

NO CONTEST

You love 'em; you hate 'em; but now you can get paid for 'em. We warned you. We told you they were coming. And now they're here. It's time for Krossword Kapers, featuring John Redfield's winning Apple puzzle stylings from last July's Playtime contest, plus our very own Atari version for the rest of you. Can you puzzle out the game names from the clues provided? Apple people play Apple; Atari people play Atari; Commodore, TI, TRS-80, IBM, and Adam folks play whichever you want. That's just our advice, of course; you can choose the system you're least familiar with for extra challenge.

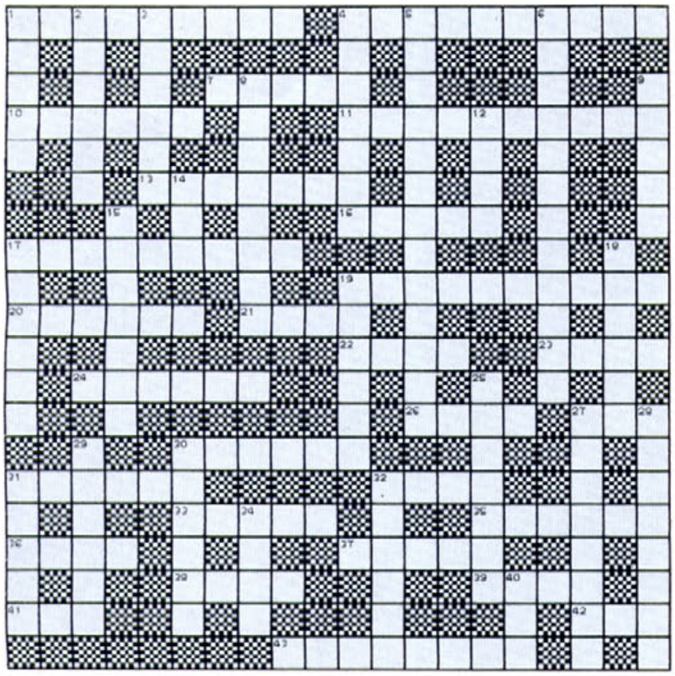
Send in a facsimile of your results; we'll worm out the infirm (some of you might get a perfect score), then draw from the highest scorers for the two winners—each receiving $100 cash or the equivalent value in software from advertisers in this magazine.

Send your puzzles to:

St.Krossword (Apple, Atari)
Box 60
North Hollywood, CA 91603
Postmark deadline: April 10, 1984.

Generic Jitters. We've been getting some . . . interesting entries to the Plain Wrap contest. Eventually, we may have to open them. If you still haven't entered, you have until April l. In case you missed it the first time around, the idea is that you're supposed to send us something indicating your desire to enter a contest. If we think your entry is worthy of entering, you win first prize in the contest ($500), without there ever having actually been a contest. Nifty, no? Explaining it to the IRS is your problem.

In Other News. The interestingly titled "Live Your Game" sweepstakes has been announced by Micro Lab as a promotion for its game, *Dino Eggs*. The sweepstakes winner and two guests will get a trip to Chicago to tour the Field Museum dinosaur exhibit. This includes air fare, a snazzy hotel room, pocket money, guided sightseeing, a visit to Micro Lab hq, and one certified genuine fossil. See your neighborhood software dealers for entry blanks.



Puzzles created with *Crossword Magic*, copyright 1981 by L & S Computerware.

Apple Softgames

ACROSS CLUES

1. SUCCESSFUL BOWLING BY PLANE.
4. SNAKELIKE.
7. THE RAY TO SPRAY A GOBLIN.
10. THE THING TO DO AT THE VULCAN'S ISLE.
11. JAMES F. COOPER MIGHT HAVE WRITTEN A GAME LIKE THIS.
13. SEEKING SUNKEN TREASURE; AN ABLE SONG TO SING.
16. IN WARS AND BATTLES THEY ATTACK NEVER TO BORE IF THEY GET BACK
17. FIGHT THE EVIL BAD GUYS RAVEIN' BUT THE NOISE MAY CAUSE A CAVE IN.
19. YOU PLAY .. ON YOUR CRT BUT WATCH IT KILL ON YOUR T.V.
20. THE CODED CLUES RANGE NEAR AND FAR BUT THAT'S THE WAY THEM BEZ FOLKS ARE.
21. THE ATTACK WAVE OF THE SPACE EGGS FUZZBALLS.
22. IF SAGEBRUSH DON'T CAUSE SCREAMING FITS . . . THE HOLES ARE WHAT YOU'LL FIND THAT IT'S.
23. UP THE CLIFF, IT GETS YOUR GOAT BUT PLACE THIS ON EACH SIDE OF MOAT.
24. THE KOMMRADS NEED A LOT OF LUCK WHEN CHASING THIS CRIMSON DUCK.
26. WHEN FIGHTING HARD WITHOUT A FEAR, THE PLACE TO GUARD BEHIND IS HERE.
27. THIS IS THE PLACE WHERE ALL GAMES STOP.
30. NO KNOCK AS YOU JUMP UPON WHAT YOU CAN ALWAYS GET BACK ON.
31. DE VOTE WAS TOOK DE NAYS TOO FEW DIS VOTE PASSED DE SPY'S RE-VIEW.
32. CHESS AND CHECKERS, OTHELLO, TOO, ARE PLAYED HERE, SO WHAT ELSE IS NEW???
33. ITS MISSION IS AFTER IT FALLS, FEMALE HORSE PICTURES ADORN ALL ITS WALLS.
35. TO SOME IT'S A JOYSTICK, A BALL OR A PADDLE . . . BUT ALL NEED SOME WAY FOR A GAME TO STRADDLE.
36. SHOOT 'EM OR JUMP 'EM I'LL GRANT YOU THIS BUT YOU'LL DIE LIKE A DOG IF YOU CONTINUALLY - - - - .
37. THERE'S ONE ON OCTOS WITH MANY A BEAST, AND THERE MUST BE 3,000 AT LEAST.
38. A GAME LIMITATION; KEEPS EVERYTHING FROM HAPPENING AT ONCE.
39. WHAT YOU DO TO THE GAME PADDLE POTENTIOMETERS TO CONTROL YOUR PLAYER, SHIP, OR PIECE.
41. THE FINAL LIMITATION TO ANY PLAY NO MATTER THE SCORE.
42. ASK VENUS AT DAYBREAK IF HER DEMON IS THIS COLD.
43. A GREAT PLACE FOR A CRISIS.

DOWN CLUES.

1. THEY DESCEND LIKE THE RAIN, AMBUSH AND DEFEND THEMSELVES . . . PASSPORTS OR GREEN CARDS NOTWITHSTANDING.
2. THE CHIEF REQUIREMENT OF TOURISTS ON RIGEL.
3. THE BOSS, GRASSHOPPER, IF YOU HAVE A YEN TO GO TO SEA.
4. A CRACK SHOT FOR AN INTERGALACTIC SWAT.
5. A DEVICE USED TO SHOOT PIXELS.
6. DAS POLAR BIRD'S EAR-SCHPLITTEN-LOUDEN-BOOMER.
8. A SINNER JUST HAS TO ESCAPE FROM HERE.
9. DAT'S-A-MOST-AMAZEY TING I EBBER FELT.
12. THE LAST STAND FOR A REVELATION.
14. WHAT YOU HAVE TO BE TO EVEN BEGIN TO PLAY A GAME.
15. AN AERODYNAMIC EQUINE; CLUE: WING IT.
17. AM NOT TREE, AM SMALLER . . .
18. FAST FELLER, TRAVELS LIGHT, AND PACKS ON THE RUN.
19. NOT PER SE A CLUE, SO BACK · TOWARD YOU.
25. MACK'S OCCUPATION & APPAREL FROM SHOCKING DESIGNS.
27. STRATEGICALLY SPREAD TO GARGANTUAN PROPORTIONS.
28. RIDER OF THE STARGATES, RETURNER OF HOSTAGES.
29. ONE WHO HAS BEEN FREITAGED, FIREBUGGED, AND FRAZZLED.
30. GEE. MY BELLY COMES TO A POINT IN THE STRANGEST PLACE.
31. THEY COME AS SEEDS OR ICED BUT THEY'RE EVIL AND SPICED.
32. MEDUSA WAS ONE, A PROTECTOR TO DREAD, TILL PERSEUS CAME AND CUT OFF HER HEAD.
34. A CONDITION WHERE THERE ARE GOALS, OPPONENTS, BARRIERS, AND RULES, ALL TOGETHER ON A PLAYING FIELD.
40. WHEN YOU ARE OUT OF AMMO OR YOUR GUNS HAVE HEATED UP AND THE BAD GUYS ARE AFTER YOU 6 TO 1 . . .

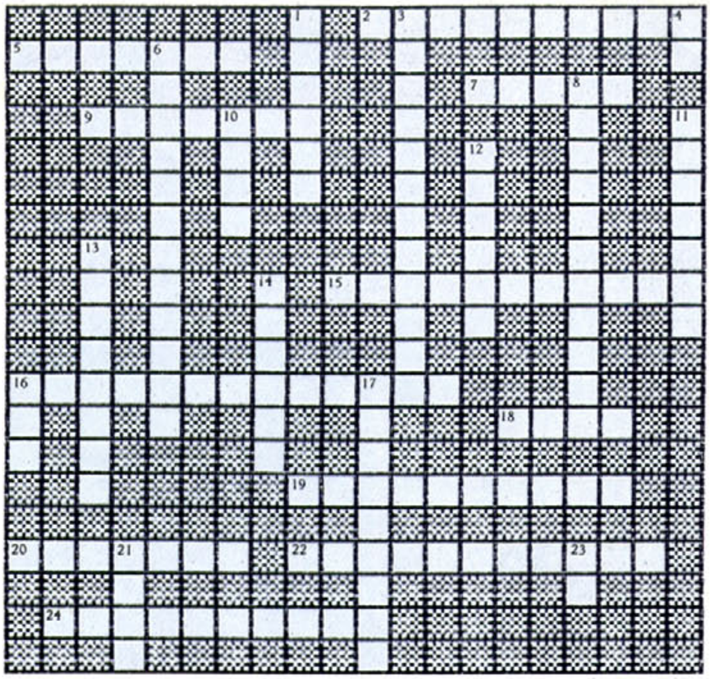
Atari Attack

ACROSS CLUES

2. HORRIBLY SCARE ATHOS WHILE WELLINGTON BEATS QUAKERS MELODIOUSLY IN SPACE.
5. REVERSE SOUNDS LIKE MALE, LEAP THROUGH BIG CIRCLE RANDOMLY REMEMBERING THE MINES.
7. SHHH. CUBISTIC THREE-FIFTHS REBEL SURROUNDED ON THE QT.
9. EFFORTLESSLY BUT BADLY FLUB EXAM IN 3-D BIPLANE INCIDENT.
15. ROMANS FIGHT BARBARIANS ON HILL; IN SANTA CATALINA'S CAPITAL, TOO MANY FRENCH, FOREIGN, GROUP.
16. ALICE'S NEMESIS IS CONSIDERABLY FASTER THAN TIGERS IN THE SNOW; PIN BALL TO ROYAL SYMPATHY.
18. OK, ROZ, MESSY BUT LOGICALLY ADVENTURESOME; PARTLY, WITHOUT HESITATING, COME IN FOR TEA.
19. SLOPPY PINE NESTER IS BESET BY SNAKELY CRITTERS.
20. SAND BIT MALICIOUSLY BY THIEVES IN SHOOT-OUT.
22. LEAD PIPE CINCH TO BESET MAC MOVERS.
24. MONTGOMERY'S LAST MINGLES WITH DREAMER SOUNDING LIKE RAZOR ON LEG OF LAMB.

DOWN CLUES

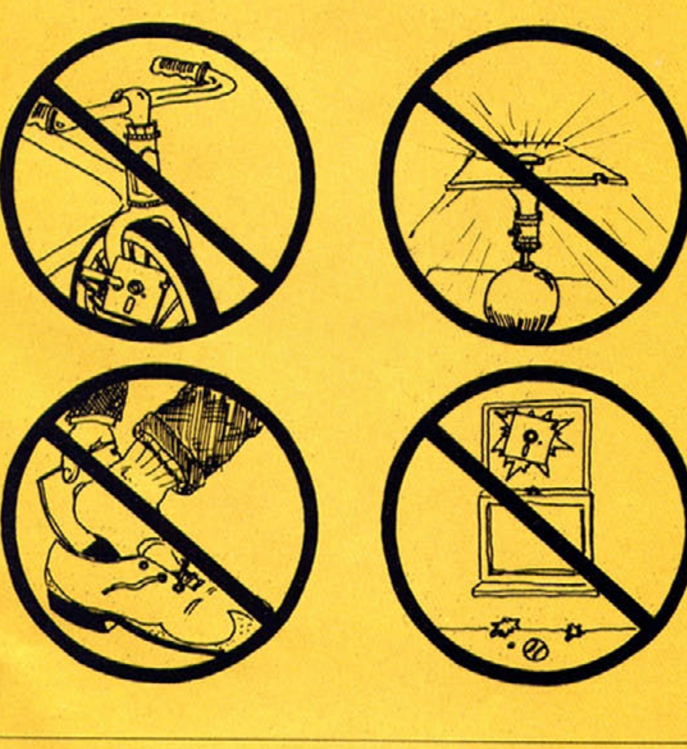
1. TAKE HATCHET TO RETARDED OX BETWEEN BRIEF SNORE AND NO ANSWER.
3. CLEOPATRA'S FLOOR NEEDS SWEEPING BECAUSE OF SINATRA'S HOTEL IN CAIRO; QUITE AN ADVENTURE.
4. FILMY SUCCESS BUT MONITORED WORTHLESSLY; AND CAESAR.
6. LISTEN TO ROYAL HILL WITHOUT CATCHING FIRE; MING CIRCA UN AMONG POOR NATO WITH HALF A KICK.
8. VRAI, DERRIERE ET MAL ON SEINE INSECT REPELLENT USER.
10. HALF THE ASSETS MAKE GAME ILLUME NASTILY LIKE HAMMURABI.
11. DEADHEADS BOMB THREE-QUARTERS SIZE, TERRIBLE CROSSES AND CROWNS.
12. BO SHOPS CONFUSEDLY IN THE BEST STORES, BLASTING EVERYTHING IN SIGHT.
13. GOODYEAR'S OMEGA PEP LINE FLOATS THROUGH PSYCHEDELIA WITH SHAMUS'S MAKER.
14. EVIL CHARON PITS CHESS AGAINST ARCADE.
16. LISTEN TO FOOTBALL PLAYS WITHOUT YOU; DIVIDE FIELD AND AVOID SHOCKING PLAYS AND FLYING STUFF.
17. BACK WATCH WARNS OF VESUVIUS; SHOOT THEM FIRST.
21. SHORTLY FUNNY GAME IS LORD OF REVERSALS, DESPITE FAMILY THEME.
23. AMERICAN EXPRESS CARD PAYS INITIALLY FOR PROGRESS INTO FAR SPACE.



Winners' Circle

Disk Don'ts. Someone once said, "Experience is the best teacher. She gives the test first and then the lesson afterwards." If that's true, then some contestants in the Symbolism contest have sure had some strange lessons.

We refer, of course, to what not to do with a disk. Most disk sleeves warn you against the obvious: don't fold disks; don't magnetize them; don't leave them in the hot sun; don't use them to pick the spinach of a spinach quiche from between your incisors. It's the less obvious ones and more imaginative ones that catch our eye, tickle our fancy, make us smile, and prompt us to throw $200 into some reader's face.



A few of these were sent in by Bob Martin (Rochester, NY), whose four best disk don'ts appear here. (By the way, the January–February issue was the first one Martin had seen. It's nice to see that he's getting into the *St.Game* spirit already.)

As promised, Martin will receive a box of ten disks and $200 worth of games for his Atari 800, along with some warnings about what not to do with them.

This Isn't the Plain Wrap Contest. There seemed to be some confusion regarding the rules of the Symbolism contest. The Plain Wrap contest let readers make up their own rules. Symbolism did not. "Send in four of your best." That's what we said, and that's what we meant. Four does not equal six. Four does not equal twelve. And recently we verified that four does not equal fifteen. But that didn't stop people from sending in more than four disk don'ts. In the cases where contestants sent in more than four, sometimes we picked the four we liked best; other times we just picked the first four, throwing away the others that might have been better. Guess we'll never know, huh?

Then there was Rosalie Meyers (Schenectady, NY), who sent four of the same symbol. "Enclosed are 'four of my best' ideas," she wrote. "It seemed like a strange request, but no stranger than your contests." What is this, a semantics lesson? We still can't tell if her symbol was a picture of a flower pot or a popcorn popper. Whatever it is, we have four of them if anyone wants some.

Okay, so we have four copies of Meyers's idea. But we have a heck of a lot more copies of other, less original ideas. It didn't take long before the contest staff got pretty sick of seeing warnings against using disks in or as the following: coasters, dart boards, eyeglasses, sundials, sandwiches, earrings, anything to do with magnets (those are real warnings!), anything to do with extreme heat (those are real warnings, too!), fly swatters, cassette recorders, phonographs (most popular), fish bait, or place mats. Yes, originality counted.

Let's Hear It For. . . . Just so they know: You winners of the Score Cleaning shindig announced last issue can say a special thank-you to the folks at Broderbund and Micro Lab—who contributed generously to the general merriment by donating the merchandise you now enjoy—and an especially special thank-you to the folks at Infocom, whose generosity is marked by an especially special specialness, as the games most winners wanted were Infocom games. Fine companies, indeed.

INFOMANIA

By Saunders and Ernst
I USED TO BE THRILLED BECAUSE ARTHUR WAS INTO COMPUTERS!
NOW I THINK HIS "BRAVE NEW WORLD" IS DOWNRIGHT UGLY!!!!
ROM SPACEKNIGHT
BLOO
TRA
...to the dance only to show me how foolish...
THE MAN IN MY PAST
I'VE NEVER FORGOTTEN HIM! WILL HE KNOW ME AFTER ALL THESE YEARS?
WELL, THERE'S YOUR CHUCK BENTON! THE BOY YOU'VE BEEN PINING FOR!
ON A TOP SECRET COMPUTER DESIGN LABORATORY NEAR ORLANDO, FLORIDA...
JIM, THIS IS SCOTT ADAMS. THE PROJECT IS READY TO BEGIN. SEND THEM TO ME NOW.
OKAY! WE'LL GET THEM MOVING!
I'D BETTER BRING SOME EXTRA WEBBING! THIS LOOKS LIKE A BIGGY!
BEN, GET READY FOR A REAL FIRST. THIS TIME WE'RE GOING ON A COMPUTER ADVENTURE!
BUT STRETCHO, I THOUGHT WE WAS GOING TO FLORIDA!!
the AMAZING SPIDER
THE RUMORS ARE TRUE.
INTRODUCING THE NEW SPIDER
MARVEL
60¢
VERSUS

Mary Worth © 1983, Field Enterprises Inc.

Photo: Kurt Wahlner

OVER THE YEARS, COMPUTERS AND COMICS HAVE MADE **SEVERAL EFFORTS TO GET TOGETHER**, WITH **VARYING DEGREES OF SUCCESS.**

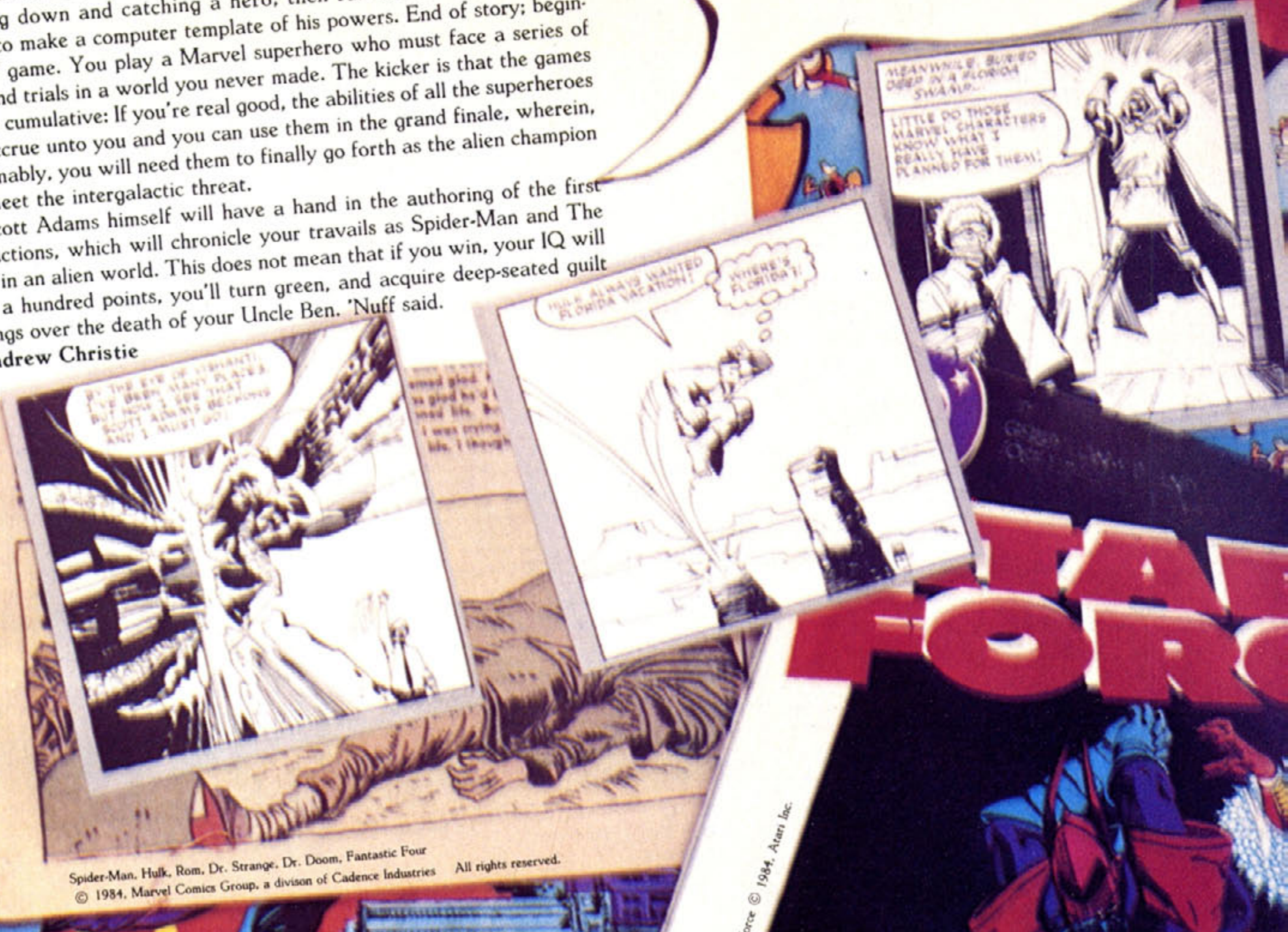
COMPUTER COMICS

As every alien race knows, when threatened with imminent extinction by another alien race, there's only one thing to do: Get next to some superheroes in a hurry.

And that's just what Chief Examiner does in *Questprobe*, the new comic book series debuting from Marvel Comics this May, accompanying the new graphic adventure series from Adventure International. The series of twelve comics will precede the events of each of twelve corresponding adventure games, to be released over the next four years. The Chief Examiner, as emissary of his race, will spend each comic book tracking down and catching a hero, then run him through a cosmic copier to make a computer template of his powers. End of story; beginning of game. You play a Marvel superhero who must face a series of tests and trials in a world you never made. The kicker is that the games will be cumulative: If you're real good, the abilities of all the superheroes will accrue unto you and you can use them in the grand finale, wherein, presumably, you will need them to finally go forth as the alien champion and meet the intergalactic threat.

Scott Adams himself will have a hand in the authoring of the first productions, which will chronicle your travails as Spider-Man and The Hulk in an alien world. This does not mean that if you win, your IQ will drop a hundred points, you'll turn green, and acquire deep-seated guilt feelings over the death of your Uncle Ben. 'Nuff said.

—Andrew Christie


BY THE EYE OF VISHANTI! I'VE BEEN MANY PLACES BUT NOW I SEE THAT SCOTT ADAMS BECKONS AND I MUST GO!
HULK ALWAYS WANTED FLORIDA VACATION!
WHERE'S FLORIDA?!
MEANWHILE, BURIED DEEP IN A FLORIDA SWAMP...
LITTLE DO THOSE MARVEL CHARACTERS KNOW WHAT I REALLY HAVE PLANNED FOR THEM!

1983 Don Bluth. All rights reserved

The Shag downs Dexter in a scene from Magicom's Space Ace.

The Laser Connection

Cinematronics blasted open the sagging arcade coin-op last year with the innovative blockbuster, *Dragon's Lair*. What made *Dragon's Lair* so spectacular was its use of a video laser disk connected to the coin-op box. The disk displayed beautiful animated pictures worthy of the Disney studios. In fact, many of the people at Don Bluth's studio (*Secret of NIMH*) are ex-Disney artists. Although the games sold for a very high $4,300 each, ten thousand of the machines were sold in the first three months. This will make quite an impact on an industry that saw kids' quarters drop off almost 50 percent during 1983!

Coleco, looking to boost sagging enthusiasm for its much delayed and plagued Adam computer, has announced that it will have Dirk the Daring available on the Adam by the end of 1984. Coleco signed an estimated $2-million deal with Bluth for the computer rights to *Dragon's Lair*, as well as future options on all of Bluth's laser disk games.

To pull this off, Coleco will need a laser disk player. While rumors link both the Dutch electronic giant Philips and the Japanese Pioneer with Coleco, Coleco is denying all. The hot rumor is that the evolving laser disk will cost only $150! That is quite a breakthrough from the standard $400–$700 range of most laser disk machines on the market today. If Coleco can in fact achieve this coup, it will outshine the original, highly touted technology of the Adam computer itself.

If this comes to pass, there are several current titles we may be fairly sure of seeing move into the home. Bluth's second game, *Space Ace*, features much faster action without the black screen pauses between sections. *Space Ace* stars a superhero named Dexter, who is reduced at the beginning of the game to an infant by the dreaded "Infanto Ray." Occasionally, he gets to change back momentarily to Space Ace. Mylstar Electronics has *M.A.C.H. 3*, which stands for Military Air Command Hunter and is a bomber game. Sega's *Astron Belt* actually includes footage from *Star Trek* and a couple of other movies. Williams's *Star Rider* features a very unusual space motorcycle racing game.

The biggest challenge to Coleco's dreams may be coming from the grandfather of arcade games, Nolan Bushnell. Bushnell's new company, Sente, is planning a series of arcade parlor game boxes called SAC (Sente Arcade Computer). Into each SAC box would go a different game cartridge. When a game gets old, only the cartridge will have to be changed. Of course, Bushnell's old company, Atari, has snapped up all the home computer rights to his new games. The SAC model III will be a laser disk machine.

Whether Coleco can actually hit its marketing target of Christmas 1984 with the first home computer laser disk game remains a matter of serious doubt. Coleco seems to thrive on climbing apparently insurmountable mountains, so the company may make it yet in spite of the competition. Competitive problems may also arise. The laser disk fever has taken hold in many segments of the home computer market, and Coleco may even find itself beaten to the store shelves by other companies. The end of 1984 could see a cluster of computer-controlled laser disk games for the home.—**Roe Adams**

Starfire We Hardly Knew Ye

They looked like such good games: *Global Thermonuclear War, Casino Master*. You almost would have thought they were real. A lot of you did. And they almost were.

This story was repeated with depressing frequency late last year: You saw the ad by Starfire Games, "Division of Omnisoft," sent in your check for $34.95, waited a while, called, were told orders were backed up/delayed/it was being shipped/reshipped. You waited another while, called again, and the phone had been disconnected. No company, no money, no game.

We agree that this ain't fun. Starfire took in a lot of people, including certain magazines who let them run big ads without paying for them. Their address turned out to be that of a condominium, in which none of the known company principals ever resided. We know you'd like to find them. Many people would like to find them. If and when we find them, you'll be the first to know. —**Andrew Christie**

Last Dance

Who sez software publishers don't know how to party? It was birthday time at Broderbund for marketing monarch Cathy Carlston. Boogying on the dance floor with Miss C. is some unnamed green monster who denies being the long-lost father of Sirius's Grud (R.I.P.).

THE MOST POPULAR PROGRAM OF 1983

The votes are in. The will of the people is served. The most and least popular programs of 1983 are now known.

The voting point system worked thus: Ten points were awarded to a program for each first place vote, nine points for second place, and so on, down to one point for each tenth place vote.

The non-Apple/Atari voting contingent is still too scattered to count; about one-tenth of one percent. Next year, though, it looks like Commodore, at least, will be represented strongly enough to make both Best and Worst into three-way events (which means *Zaxxon* could get on *six* times).

Most Popular Program: Apple

Place	Points	Program, programmer, company
1.	1,786	*Lode Runner*, Doug Smith, Broderbund
2.	1,254	*Legacy of Llylgamyn*, Andrew Greenberg and Robert Woodhead, Sir-tech
3.	1,096	*Ultima III*, Lord British, Origin Systems
4.	723	*Miner 2049er*, Mike Livesay, Micro Lab
5.	717	*Zaxxon*, John Garcia, Datasoft
6.	669	*Hard Hat Mack*, Michael Abbot and Matthew Alexander, Electronic Arts
7.	478	*Drol*, Benny Ngo, Broderbund
8.	448	*One-on-One*, Eric Hammond, Electronic Arts
9.	428	*Pinball Const. Set*, Bill Budge, Electronic Arts
10.	339	*Planetfall*, Steve Meretzky, Infocom

Most popular program of all time: *Wizardry*, Andrew Greenberg and Robert Woodhead, Sir-tech

Most Popular Program: Atari

Place	Points	Program, programmer, company
1.	506	*Archon*, Free Fall, Electronic Arts
2.	497	*Blue Max*, Bob Polin, Synapse
3.	385	*M.U.L.E.*, Ozark Softscape, Electronic Arts
4.	347	*Pole Position*, Atari
5.	266	*Lode Runner*, Doug Smith, Broderbund
6.	261	*Miner 2049er*, Bill Hogue, Big Five
7.	252	*Jumpman*, Randy Glover, Epyx
8.	227	*Ultima III*, Lord British, Origin Systems
9.	174	*Pinball Const. Set*, Bill Budge, Electronic Arts
10.	165	*Shamus*, William Mataga, Synapse

Most popular program of all time: *Star Raiders*, Atari

DOG OF THE YEAR

It's still the same old story. . . . Just as *Frogger* walked away with the contest in 1982, once again another arcade hit was translated for the home computer, upsetting folks because it didn't look like the original. Maybe if they'd called it *Zaxxon Jr.* . . .

Here, the game in the #1 spot received the most votes, meaning #10 was only one-tenth as unpleasant. Names of companies and programmers are tactfully unmentioned.

Obviously, few were able to restrict themselves to 1983 releases. We must bow to the will of the majority—except in those instances where some troublemakers insisted on including the wonderful new name of this magazine on their ballots.

The technically lucky winners of the random drawing from all ballots received are C.V. Fields, who will get *The Missing Ring* for his Apple, and Steve Koppke, who will get *Alien Garden* for his Atari—used, of course.

Apple

Place	Points	Program
1.	183	*Zaxxon*
2.	144	*Cubit*
3.	143	*Jump Jet*
4.	108	*Frogger*
5.	90	*Canyon Climber*
6.	83	*Tubeway*
7.	81	*Plasmania*
8.	77	*The Missing Ring*
9.	74	*Bouncing Kamungas*
10.	73	*Dig Dug*

Atari

Place	Points	Program
1.	136	*Congo Bongo*
2.	124	*E.T.*
3.	117	*Zaxxon*
4.	88	*Dig Dug*
5.	66	*Q*Bert*
6.	55	*Star Trek*
7.	49	*Floyd of the Jungle*
8.	48	*Spy's Demise*
	48	*Gwendolyn*
10.	39	*Wall War*

Dubious Honorable Mention:

IBM: *Paratrooper*

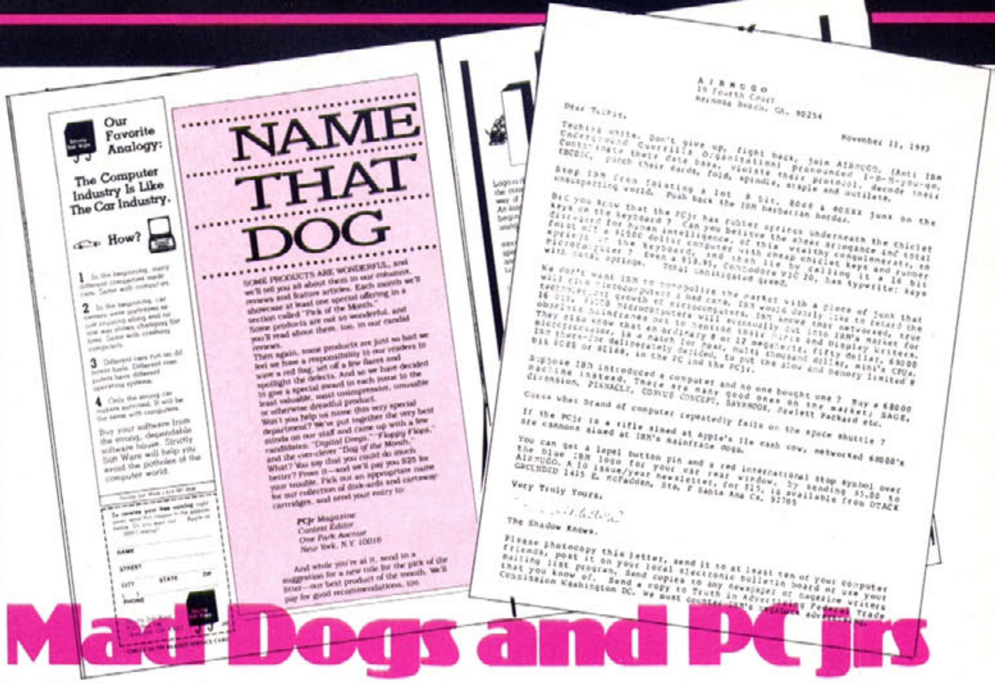
C-64: *Robber of the Lost Tomb*

ROBOT HERO

When the sound of the fearful clashing of program against program finally subsided, only one construct remained sufficiently functional to be acclaimed the National Robotwar Grand Champion. Doug Hogg's *Dragon4A* is currently the robot to beat and the first ever to be ranked grand champion. Hogg pulled off his achievement at the Postal Robotwar Championship Tournament of December 18th last—one of the convocations for disciples of Muse Software's strategic programming game, in which participants bring in their disks or mail them in to do battle with each other. Hogg's *Dragon* beat out Tom Schai's *Bucweat*; two points in front of Richard Fowell's *Norden +*, and six points in front of Stanley Chen's *Indian2*.

Frank Krogh, founder and scribe of the Robotwar Club, estimates that "in four or five years there will be enough Grand Champion robots to hold a Grand Champion tournament." Till then, Hogg's achievement stands alone—and secure.

Meanwhile, Novice, Master, and Champion tournaments go on. Info can be had by writing to Frank Krogh, Box 5337, North Hollywood, CA 91616. **—Andrew Christie**



Our Favorite Analogy:

The Computer Industry Is Like The Car Industry.

How?

1 In the beginning, many different companies made cars. Same with computers.

2 In the beginning, car owners were professionals [illegible]

3 Different cars run on different fuels. Different computers have different operating systems.

4 Only the strong car makers survived. It will be the same with computers.

Buy your software from the strong, dependable software house. Strictly Soft Ware will help you avoid the potholes of the computer world.

[illegible]

NAME

STREET

CITY STATE ZIP

PHONE

NAME THAT DOG

SOME PRODUCTS ARE WONDERFUL, and we'll tell you all about them in our columns, reviews and feature articles. Each month we'll showcase at least one special offering in a section called "Pick of the Month." Some products are not so wonderful, and you'll read about them, too, in our candid reviews.

Then again, some products are just so bad we feel we have a responsibility to our readers to wave a red flag, set off a few flares and spotlight the defects. And so we have decided to give a special award in each issue to the least valuable, most unimpressive, unusable or otherwise dreadful product.

Won't you help us name this very special department? We've put together the very best minds on our staff and come up with a few candidates: "Digital Dregs," "Floppy Flops," and the ever-clever "Dog of the Month." What? You say that you could do much better? Prove it—and we'll pay you $25 for your trouble. Pick out an appropriate name for our collection of disk-ards and castaway cartridges, and send your entry to:

PCjr Magazine
Contest Editor
One Park Avenue
New York, N.Y. 10016

And while you're at it, send in a suggestion for a new title for the pick of the litter—our best product of the month. We'll pay for good recommendations, too.

A I B M U G O
16 Fourth Court
Hermosa Beach, CA. 90254

Dear Techie,

November 11, 1983

Techies unite. Don't give up, fight back, join AIBMUGO. (Anti IBM Underground Guerrilla Organization) Pronounced I-B-M-you-go. Contaminate their data base, violate their protocol, decode their EBCDIC, punch their cards, fold, spindle, staple and mutilate.

Stop IBM from foisting a lot 8 bit, 8088 & 80XXX junk on the unsuspecting world. Push back the IBM barbarian hordes.

Did you know that the PCjr has rubber springs underneath the chiclet keys on the keyboard ? Can you believe the sheer arrogance and total disregard for human intelligence, of this wealthy conglomerate, to foist off a $1500 dollar computer with cheap chiclet keys and rubber springs on the keyboard, and then lie by calling it a 16 bit microcomputer ? Even a $19.95, Commodore VIC 20, has typewriter keys with metal springs. Total unmitigated greed.

We don't want IBM to monopolize the market with a piece of junk that will give microcomputers a bad name. IBM would dearly like to retard the technological growth of microcomputers. IBM knows that networked, true 16 bit, 68000 microcomputers will eventually cut into IBM's market for obsolete mainframes not to mention their [illegible] and display writers. They also know that an ordinary 8 or 12 megahertz, fifty dollar, 68000 microprocessor, is a match for most, multi thousand dollar, mini's CPUs. IBM therefore deliberately decided, to put the slow and memory limited 8 bit 8088 or 8088, in the PC and the PCjr.

Suppose IBM introduced a computer and no one bought one ? Buy a 68000 machine instead. There are many good ones on the market; SAGE, Dimension, PINNACLE, CORVUS CONCEPT, BAYBROOK, Hewlett Packard etc.

Guess what brand of computer repeatedly fails on the space shuttle ?

If the PCjr is a rifle aimed at Apple's IIe cash cow, networked 68000's are cannons aimed at IBM's mainframe dogs.

You can get a lapel button pin and a red international stop symbol over the blue IBM logo for your car rear window, by sending $5.00 to AIBMUGO. A 10 issue/year newsletter, for $15, is available from DTACK GROUNDED 1415 E. McFadden, Ste. F Santa Ana Ca. 92705

Very Truly Yours,

The Shadow Knows.

Please photocopy this letter, send it to at least ten of your computer friends, post it on your local electronic bulletin board or use your mailing list program. Send copies to any newspaper or magazine writers that you know of. Send a copy to Truth in Advertising Federal Trade Commission Washington DC. We must counter IBM's [illegible] advertising.

Mad Dogs and PC jrs

What does the Shadow know? Whatever he knows, he'd like to let us all know how he feels about IBM in general and the PCjr in particular. To that end, he has organized AIBMUGO, the Anti IBM Underground Guerrilla Organization, pronounced I-B-M you go. (Has this company no end of disgruntled former employees?) "Push back the IBM barbarian hordes," he exhorts in his manifesto of November 11; "Did you know that the PCjr has rubber springs underneath the chicklet keyboard?" and so on. A window sticker with the red international "stop" symbol over the blue IBM logo is available for $5, and a ten-issue subscription to the AIBMUGO newsletter for $15, from DTack Grounded, 1415 East McFadden, Suite F, Santa Ana, Ca 92705.

Meanwhile, congratulations to PCjr on the premiere of their new mag. It looks very original and unique. One of the uniquely original parts of it is the feature that invites you to "Name That Dog" among the current crop of software. What can we say? We're flattered, honored, and ennobled by this example of the sincerest form of flattery.

In all fairness, they also invite you to suggest some other name they could call it—like "Dog of the Month." By all means. **—Andrew Christie**

MA BELL GETS A CABBAGE PATCH KID

American Telephone and Telegraph (the main long-distance-line national company) recently announced a joint venture between its AT&T Consumer Products division and Coleco Industries of Hartford, Connecticut. The new venture will provide interactive games and other forms of entertainment over existing phone lines. To facilitate computer owners' use of this new service, low-cost (under $100) modems will be sold to subscribers.

Early reports concerning this venture initially caused widespread confusion in the industry. Many thought the announcement signaled the entrance of the two large companies into the emerging field of direct downloading of computer software to either retailers or residential users. (Xante is one company working directly with retailers, while Control Video Corporation's Gameline caters to the home user.) Instead, it appears that the venture is taking a page from Scott Adams's modem game of *War*.

All the games will be for two players. A player in New York inserts a special cartridge or disk into his or her game machine or computer. The player can then call a 900 number (flat rate charge) and enter his/her name via joystick selection. The telephone's computer will connect the player with another aspiring player on-line somewhere else in the United States. The two players fight it out for the top score, and the scores will be recorded. Some type of nationwide competition may evolve from this, although that has not been formally announced.

The Yankee Group, a Boston-based think tank, has predicted that the modem will be built by Bell Labs and that it will be a slow, 300-baud modem. Gameline currently uses a modem rated at 1800 baud, but apparently AT&T/Coleco believe that the 300-baud modem will be fast enough. The Yankee Group notes that according to "knowledgeable sources," no one can operate a joystick faster than fifty bits per second. They also project that, with "21 million video-game consoles and 8.4 million home computers" in residential use by the end of 1983, this new form of interactive gaming will generate $1 billion in revenues by early 1986.

To date, only three other forms of interactive modem gaming exist. One is Adventure International's *War*, a two-player strategic game. Another is CompuServe's text-only game, *Megawars*. The most ambitious, though, is Harlow Stevens's *Gamemaster* in Chicago, where, via graphic drivers, a five-player game of *Star Trek* may be played. All players see the events of the game displayed on their own separate consoles.

The biggest obstacle to the success of any of these approaches is the incredible cost of long-distance rates. (Imagine the bill for an average two or three hours of playing; whole games can be bought outright for a lot less.) Also, at the 300-baud mode, game play will proceed like molasses in winter.

Still, the concept is intriguing. Lots of people are rooting for this new AT&T/Coleco venture to succeed. And if anyone can pull off lowering the telephone costs to a tolerable level, it's AT&T. **—Roe Adams**

continued from page 39

Making matters worse is a flashy little static charge who chases your man around the screen. If he catches you enough times, you get "charged up" and will "fry"any chip you touch, requiring a quick trip to the power supply, where you can ground your static.

But the bugs are consistent with their symptoms, and so, with a little persistence and some really close scrutiny, you can debug the system, return to Gotcha, and perhaps find yourself learning a lot more about how a computer works than you might expect.

D-Bug is a slightly masochistic, wholly fascinating game of skill, strategy, and knowledge that turns program bugs into educational game features. HAS

Atari, disk. $40 from Electronic Arts, 2755 Campus Drive, San Mateo, CA 94403; (415) 571-7171.

Unite or Die

Chancellor of the Exchequer

By R.C. Chiofolo.

Atari.

Here's a game that's a prime example of the strategy genre. Detail and depth that few other Atari games can meet are part and parcel of this game. More than a decade of British history is yours to command as the nation's chief financial officer—the Chancellor of the Exchequer.

As this game starts, you are faced with a loosely knit organization of eight geographical regions encompassed by Great Britain. Your object is to industrialize Great Britain by merging these separate regions into a nation. This can only be done by employing machines and people in a manner that supports growth.

To merge regions successfully, people are needed—healthy people to work the mines, harvest the crops, man the steam engines, run the trains, and navigate the ships. Population figures are one type of statistic you must constantly be aware of for each region. There must be enough food to maintain a healthy workforce. Mechanization of the farms is of some help, but in order to build tractors, tools must be made from the raw materials dug from the mines. Everything is interdependent, making each region quite a responsibility. And there are eight regions, in addition to the nation as a whole.

Regions can trade anything with one another, from crops and coal to money itself (at a 10 percent discount rate). Should unemployment rise in a particular region, or a call be received for men-at-arms, the population may be redistributed to better meet specific circumstances. Say, for example, your coal mine in Northumberland possesses three drills, yet only has one population worker toiling in the mines. This means that most of the drills are left unused. Waste! During the allocation sequence, a good idea might be to take surplus workers from transportation and move them into the coal mines to increase productivity, since coal is one of the resources needed to manufacture tools. Tools may then be converted into tractors for the farm, drills for the mines, or steam engines for the factories. As the object of the game is to industrialize, the importance of tool production cannot be stressed enough.

At any time during nonstage play, an analysis program may be run. This program analyzes your moves up to that point and offers handy hints for improving a specific region's statistics. A graph offers visual representation of how well you're playing.

Thoroughly enjoyable. Don't expect to sit down and master this game in half an hour. Involvement could last as long as a day, depending on the level of difficulty you choose. All regional and national information is presented spreadsheet style, with certain graphic characters representing specific items. A brief musical interlude at the beginning of the game only hints at what you might attain—rule Britannia. HGL

Atari, 48K; disk. $29.95 from Mach-ina, 161 Aragon Avenue, Coral Gables, FL 33134; (305) 443-0027.

Thinkless Tasks

Space Knights

Text by David Heller. Programs by Robert Kurcina.

Atari.

Infocom coined the term *interactive prose* to describe its text adventure games. The image suggests a story in which the events are controlled by the "reader."

Reston Publishing went a step further with *Space Knights*; it's an actual novel (although a short one) accompanied by a disk of games that relate vaguely to the events of the story. Great idea; woefully inadequate execution. Infocom doesn't have much to worry about.

Space Knights exploits every cliché known to science fiction. Our hero, young Jake, is a typical space nerd anxious to complete his training and graduate into the ranks of the Space Knights. This desire is incomprehensible; no Space Knight has ever returned to Earth, and the peculiarities of faster-than-light travel result in family and friends being four hundred years dead when he waits for his first mission. This career provides an adrenaline rush for young folks? Not too likely.

Jake and his haughty female companion, Lisa, embark on a series of adventures that bear no relation to each other or to any higher purpose. One ludicrous escapade involves refugees from Marin County who have worked three generations to create "the perfect wine." The chapter is crammed with embarrassing "space hippies." Other adventures are variations on the basic "idiot plot": Things happen only because every character acts like a total idiot at all times.

One narrow escape from space pirates, for example, results in a crash-landing on the planet Weomby. After deciding that repairs are needed—for the shuttle's helicopter, among other things—Jake and Lisa begin a ten-kilometer journey, on foot, to the nearest city. After many tedious, dangerous hours, they arrive just in time to greet the repaired shuttle helicopter, which the computer sent to help them.

Why take all that trouble when they could have waited aboard the ship in safety while the helicopter was repaired, and then flown to the city just as quickly?

The conclusion involves a twist stolen from *Planet of the Apes* and countless other post-holocaust stories. Spare me, the reader sighs; another irradiated planet that turns out to be Earth?

The games, all in Basic, have clunky graphics and poor player control. Several require paddles, which offer even less control. Weomby and Impact are routine lunar lander programs, although the former becomes intriguing with an option for stronger gravity. Find Demons, Bug Buster, and Gamelon Encounter pit the player against various nasties in arcade-style action.

Navigate is the best of the lot. The player uses coordinate algebra to chart and execute courses from any one of twelve planets to another. There is an unexplained glitch, though, which prohibits travel from planet three to planet four.

One game has nothing to do with the novel. It's called the Mystery Game, and that's a perfect description. Because it has no instructions, its function or purpose is a total mystery.

Both book and games seem like rushed first drafts. Even young boys (no girl could stand the sexism) will be frustrated by the novel's lack of direction. Since the games do nothing to advance weak plot lines, they are too simple to be interesting by themselves. DB

Atari; disk. $24.95 from Reston Publishing, 11480 Sunset Hills Road, Reston, VA 22090; (703) 437-8900.

Schweigen Sie! Ich jage die Panzer.

Panzer-Jagd

By Richard W. Scorupski.

Atari, Commodore 64.

This is a game of strategy and nerves that re-creates the Russian Front, 1943, albeit on a more limited scale than the real thing. The player commands several German units. This combination of tanks, infantry, and artillery must support the main German offensive by creating a diversion through a flanking action. Such a maneuver, it is hoped, will negate some of the offensive fire that would have been brought to bear on the main German thrust.

The goal is to win the field of battle and capture as many enemy sectors as possible. This may be accomplished by either capturing the town found in each sector or by eliminating all enemy units in that particular seetor. Total enemy elimination is highly advised. Otherwise, after the town has been captured, enemy forces that have successfully retreated will strengthen units in the succeeding sector. The computer-controlled enemy consists of infantry, T-34/85 battle tanks with 85mm cannon, and several 45mm antitank guns.

A hi-res display reveals a battlefield composed of symbols representing heavily wooded areas, streams, open areas, and the town itself. Only your infantry can move through thick forests. Enemy units become visible after they have fired at one of your units, when they have moved into open terrain, or when a German unit becomes positioned adjacent to a concealed Russian position.

A turn sequence consists of five phases. Your joystick controls all of the action. First comes the artillery barrage. A word to the wise: Remember where you've directed such attacks. A foolish commander is one who moves forces onto an area targeted for a friendly shell burst. The location of the artillery attack is determined by the position of an on-screen cross hair controlled by your joystick.

The direct fire order phase is the second element of the turn. The enemy target and your firing unit must lie in an unobstructed line of sight in order for the computer to allow any exchange of hostilities to occur. The movement phase follows, with the cross hair designating the unit to be moved and its intended location.

Once this has been accomplished, the fire execution phase is begun, with the sounds of both incoming and friendly artillery shells roaring across the battlefield. Bright explosions indicate the firing zone on the screen. Staccato bursts of machine-gun fire and the explosion of tank shells announce that the direct fire orders are being carried out. Once the resolution of combat has been completed, movement of all remaining units (German and Russian) is undertaken. You'll spot enemy units that suddenly become visible as positions are changed, and suddenly they'll vanish from sight. Make note of the terrain surrounding the area where you last observed the Russian force; there's a good chance that an artillery barrage into that particular section of woods could produce a fortuitous hit on an enemy unit.

Each of your units has a certain combat strength (defensive/offensive capabilities), range of fire (effective fire on target unit), and movement factor (distance unit may travel in a single turn). These statistics are revealed to you at the start of the game and when you access a unit during play. A message window beneath the battlefield screen displays this information when you place a cross hair over a specific unit and press the fire button. Additionally, panzer units require fuel to operate. Keeping track of the fuel remaining for each tank is critical to your success. Once the fuel is gone, all movement orders to that unit will be ignored. Also, remember that moving tanks through more difficult terrain, such as streams or woods, expends more fuel.

Those playing the Atari disk version have an additional, shorter game to play. Management of fewer units with less opposition makes this a good introductory scenario.

Despite what seem to be some slow computational factors by the computer, this is a sound war strategy game. May you beat the panzers off the computer! HGL

Atari, Commodore 64; 32K cassette $25, 48K disk $30 from Microcomputer Games/Avalon Hill, 4517 Harford Road, Baltimore, MD 21214; (301) 254-9200.

Let's Be Careful in There

Murder by the Dozen

By The BrainBank.

Apple, Atari, Commodore 64, IBM PC, PCjr.

For the sleuths among us, here's a game of crime and puzzlement. Imagine yourself as one of four detectives arriving at the department some Monday morning and finding twelve unsolved cases awaiting investigation. Each is an alleged homicide, some rather gruesome in description. As Gilbert and Sullivan put it, "a policeman's lot is not a happy one."

Once you've decided on a particular case as your assignment, the computer manages your activities. For example, one homicide might involve an apparent robbery. Before leaving the police department to check various leads, a good idea might be to call the lieutenant in charge of your department's robbery division. Perhaps a concurrent investigation is already under way, due to the robbery angle, and it may have produced some viable suspects.

Your duty is to find the perpetrator and present your case for prosecution. Should you happen to miss vital facts en route to your conclusion, the case is lost. Lab and autopsy reports are crucial reading, while examination of physical evidence at numerous locations may or may not offer clues to help you solve the crime. Questioning the various individuals within your jurisdiction may prove fruitful, or it may produce nothing more than a red herring to misdirect the investigation. You must be clever enough to recognize the truth based on the facts you learn during your detective work.

Notes are critical. As each stone is overturned, numbered clues are offered and must be matched to the appropriate clue in the clue book. Here lies one of the game's problems, especially if it is being played by more than one detective. Some clues consist of six numbers, which requires a lot of page-flipping through the clue book, occasionally revealing a "no clue" statement. This can be annoying, especially if other players are waiting to read the clue manual. Highly recommended is team play in which each of the participating detectives works together to solve a case.

Players receive the final rating of "sleuth" based on the accuracy of their conclusions and the amount of time taken to reach the case's final presentation. Your case work is matched against the answer in the solution booklet.

Enjoyable, frustrating at times, thought-provoking . . . this software should provide hours of entertainment. A policeman's lot is *sometimes* a happy one. HGL

Apple, Atari, Commodore 64, IBM PC, and PCjr.; disk. $34.95 from CBS Software, One Fawcett Place, Greenwich, CT 06836; (203) 622-2500.

What's Hot, Hilly, and Wriggles?

Earthquake: San Francisco 1906

By Jyym Pearson and Norman Sailer.

Apple, Atari, CoCo, TRS-80.

The earth shook, the sky burned. It is San Francisco, 1906, and the City by the Bay has just suffered one of the most devastating earthquakes of our time. Unfortunately, this disaster has occurred on the very day you arrived in the Bay Area. You even got a room with a view of Telegraph Hill. What luck. Only minutes later, the San Andreas Fault slipped and the hotel collapsed around you, pinning your legs in the rubble.

Fires burn everywhere, and martial law is in effect throughout the city. Surviving the initial quake is one thing; living through its aftermath is another. Count your blessings; you are alive. Now you must find a way to free yourself and make your way through the ruins of a once-proud city.

Earthquake is a typical game from Adventure International, having many of the same strengths and weaknesses as the rest of the hi-res adventures in the Other Venture series. The graphics are nicely drawn and colored, adding a lot to the game in terms of visual appeal, while the vocabulary and text descriptions are just adequate for moving the game along at the pace expected by intermediate to experienced adventurers. The puzzles are great. Some are hard, some are harder. Most of them are fun. A few of them irritate before yielding solutions. *Earthquake* might not be one of AI's easier games; that's all right, though, because those who look for games offering good dollar-per-hour value are sure to get their money's worth.

Graphically, the game is pretty good. However, its small vocabulary and limited parser detract as much from the game as the graphics add. The result is that, in a few cases, guessing which words the computer wants you to use is just as hard as solving the puzzle. While it does not take long to get an idea as to what simple verb commands the program does and does not understand ("get the apple" rather than "take the apple," and "look at" the paper rather than "read it"), there are instances where the program logic comes up short and the play becomes confusing. This is especially true when attempting to deal with objects found in the course of adventure (an iron box in the rubble of an opera house was certainly identifiable, but the programmer refused to classify it as "visible").

Still, the story behind *Earthquake* is fascinating enough to put this game ahead of most other current releases. By taking an event like the San Francisco earthquake and dramatizing it in an adventure game, authors Pearson and Sailer have expanded the story potential for adventure games into areas previously unused. In this sense, despite its rough edges, *Earthquake* is an interesting and important program. MS

Atari, CoCo, TRS-80, 16K, tape; $24.95. Apple, Atari, 48K, disk; $39.95 from Adventure International, Box 3435, Longwood, FL 32750; (305) 862-6917.

On Disk, No One Can Hear You Scream

Crypt of Medea

By Allan Lamb and Arthur Britto.

Apple.

They're creepy and they're kooky, mysterious and spooky, they're altogether ooky, the adventures in the Crypt of Medea. Da da da dum (snap, snap).

Crypt of Medea is something of an Addams Family on disk. It is macabre and witty and stupid and inventive and repetitious and altogether kooky. While driving along a deserted road one rainy night, you lose control of your car and spin out. You awake in a vault of the Crypt of Medea. The objects in the room are a glass case that is too dirty to see through, an unlit candle, and several tombs. No exits, no entrances, no doors, no windows. Go to it.

Stuck? Time to consult the manual. *Crypt*'s manual acts as an adventuring tutorial; it takes the novice adventurer through the first three rooms, showing how objects are manipulated and which command words are effective. The first three rooms are something of a detour; the manual leaves you back in the original vault, only now your candle is lit and you have a shovel in your possession. The manual also provides a list of words recognized by the computer and a list of hints to help you out of particularly obscure situations. Okay, *now*. . . .

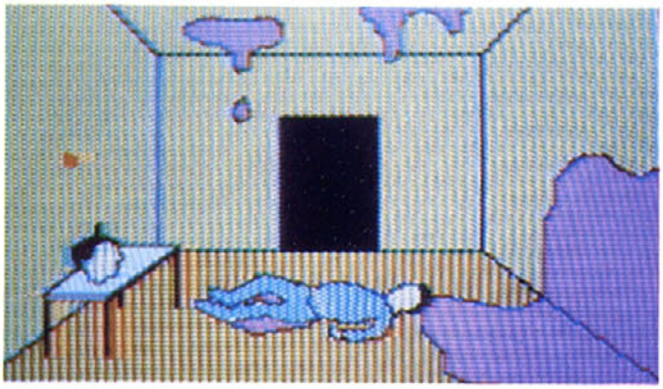
Dead, eh? Did you drown in the room of blood, or was it the spikes through the brain? Time for a generalization. The Crypt of Medea is a dangerous place in its Karloffian way. Lots of movie gore. Lots and lots.

The graphics are colorful but lack detail. The jellied mass containing human heads looks like a mound of orange sand, the decapitated dog looks like God-knows-what. Also, this is one adventure that definitely calls for animation but provides little. If games featuring lots of blood and monsters excite you, imagine how much better they would be animated. Instead of telling you about bubbling blisters or melting flesh, why not show you? You're going to have nightmares anyway.

The graphics don't remain on the screen while commands are typed in; you have to toggle back and forth between the graphics and the command level. *Crypt* has some funny sound effects that soon become annoying. Both the graphics and the sound can be turned off for faster, less annoying play.

Many of the solutions to the problems that present themselves while you try to get out of the crypt are rather . . . cryptic. For instance, to get to the room after the jellied mass, you have to play a tape you've found—but it must be played in the correct room to yield any result. If you play it in the wrong room, there is nothing to indicate that you've screwed up. And because there is no alternate solution, you could wander around that dank, dark crypt for the rest of your short life. It gets to be irritating, and eventually the hints provided in the manual are your only recourse. Because the game is too hard, it is simultaneously too easy; once you've looked up the hint for any particular problem, you've solved it. Your alternatives are to get out with a lot of help or not to get out at all.

Whereas Gomez Addams was a marvel of eloquence, the *Crypt* has trouble communicating. When presented with a butterknife that you try to take ("take butterknife"), you are promptly rewarded with nonsense. The program does not recognize the existence of a butterknife. A knife, yes; a butterknife, no. Bothersome. The evident tendency of software companies to scrimp on the services of proofreaders when producing computer adventures is also on view here.



The *Crypt* is full of funny surprises. The animation that it does have, though rare, occurs unexpectedly to tickle the weary adventurer back into good humor. When the sequences arise, they generally signal the end of the current round and the end of your current life; you die laughing.

Crypt of Medea is a program for the beginning adventurer in the family. Sophisticates like Gomez won't get much out of it, nor will the squeamish. This is for the Pugsleys and Wednesdays in your family—or better yet, the Festers and Lurches. TZ

Apple, 48K. $34.95 from Sir-tech, 6 Main Street, Ogdensburg, NY 13669; (315) 393-6633.

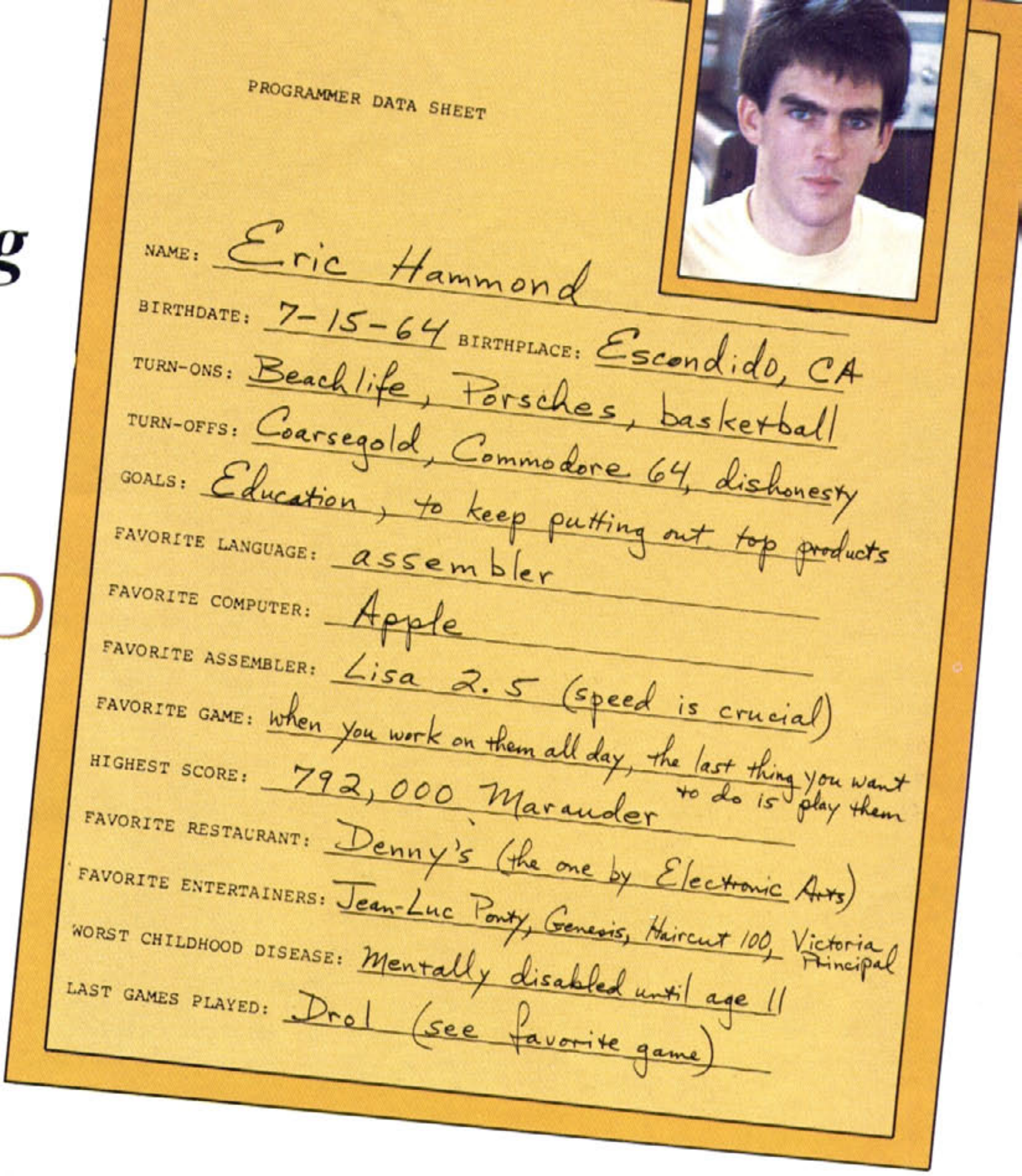
Guest reviewers this issue are Derrick Bang, Monte Schulz, and Howard A. Shore.

Hawaii
PRIN

Profiles in Programming

ERIC HAMMOND

by Matt Yuen



PROGRAMMER DATA SHEET

NAME: Eric Hammond

BIRTHDATE: 7-15-64 BIRTHPLACE: Escondido, CA

TURN-ONS: Beachlife, Porsches, basketball

TURN-OFFS: Coarsegold, Commodore 64, dishonesty

GOALS: Education, to keep putting out top products

FAVORITE LANGUAGE: assembler

FAVORITE COMPUTER: Apple

FAVORITE ASSEMBLER: Lisa 2.5 (speed is crucial)

FAVORITE GAME: when you work on them all day, the last thing you want to do is play them

HIGHEST SCORE: 792,000 Marauder

FAVORITE RESTAURANT: Denny's (the one by Electronic Arts)

FAVORITE ENTERTAINERS: Jean-Luc Ponty, Genesis, Haircut 100, Victoria Principal

WORST CHILDHOOD DISEASE: Mentally disabled until age 11

LAST GAMES PLAYED: Drol (see favorite game)

Photo: Kurt Wahlner

You just want to hit a guy like Eric Hammond, but you can't.

Remember that classmate who finished his homework in seconds flat and spent the better part of his spare time not doing much of anything? Remember the kid who got backstage passes to the concert of the year but didn't go because the stage door was too far from the parking lot? Grrr. It's not that Eric Hammond is spoiled, or that he doesn't give a hoot about things. He just takes them in stride.

Right now he's taking in stride the runaway success of his latest game, *Julius Erving and Larry Bird Go One-on-One*. Hammond is quite a talented person academically, athletically, and musically. He's also a pretty good programmer, though his previous releases enjoyed only modest success. This time, however, Hammond found the daylight, got a lead pass from Electronic Arts, and is slamming it through the hoop—right into the faces of people who want to hit him.

University of Nitchals. Hammond was introduced to Electronic Arts through fellow programmer Jim Nitchals. "We have this running joke at Electronic Arts about the School of Jim Nitchals," says *One-on-One* producer Joe Ybarra. "There's a whole wealth of programming talent in the San Diego area, and it seems like they all learned under Jim."

Electronic Arts asked Hammond in late 1982 if he'd like to do a game for them. In those days, the company was about as well known as Buddy Dunn (who?), so it was no wonder that Hammond felt wary about collaborating.

He finally decided to at least meet these people who called their programmers "artists" and see what they were all about.

"At first, they wanted me to do a football game, but I didn't think that was too cool; I told them I was more interested in basketball," says Hammond.

"When he mentioned that, our eyes lit up and everything clicked," says Ybarra, an avid basketball fan. "Later [Electronic Arts's president], Trip Hawkins came up with the idea of getting Julius 'Dr. J' Erving and Larry Bird to spice it up and endorse the game."

Hammond is very casual when he talks about the game-design sessions with the two basketball stars ("Julius was really great. . . . Larry's a bit more laid back . . ."). Maybe it's because he spent three hours with Erving talking about what special effects the game should have. Maybe it's because of the hours Hammond and Erving spent shooting baskets at a local YMCA. Or maybe it's the feeling of brotherhood that game makers share with each other—after all, Erving and Bird share coauthor billing with Hammond on *One-on-One*'s package.

Hammond doesn't mind having them listed as coauthors, because the kinds of things Erving and Bird added to *One-on-One* made it great. Erving came out to Electronic Arts during the summer of 1983. At that point, *One-on-One* was little more than two players running around on the screen and shooting baskets.

"Julius thought it was pretty neat because the players were so big and animated. He had some ideas about turning and spinning to face the basket, and a lot of ideas about how the defensive player should be able to block and steal the ball."

Scouting Reports. Not all the subtle nuances of *One-on-One* came from Erving and Bird. Hammond already knew some of the things he wanted to include in the game. He had been watching The Doctor on

television long enough to know a lot of little things about him.

Like what? "I basically knew this guy could hang in the air forever; he has a lot of cool moves," Hammond says.

He wanted to include as much "cool" stuff in *One-on-One* as he could, but computers have their limits. As a result, he ended up "cramming as much as I could into the game until the memory was full." Cram he said, and cram he did. What began as a fun computer game was turning into a programming nightmare.

Shattered backboards, instant replay, hacking fouls . . . all these features were added as extras, usually due to Hammond's big mouth. "One day I mentioned that maybe we could have the backboard shatter after a vicious slam dunk, and everyone became really enthusiastic. But then I thought, 'Wait a sec, what am I getting myself into?' " However, the special effects certainly add to the game, and he's glad he included them. In this case, the pleasure evened out the pain.

Getting to meet two of basketball's greatest forwards was really only part of the thrill Hammond experienced during the game's development. A different kind of thrill happened in Springfield, Massachusetts, when Electronic Arts's ad agency went to shoot pictures of Erving and Bird in action for the game's package.

Amateurism is part of the feel of true-life one on one, so Electronic Arts chose a public school playground as the set for photos. "All the camera stuff was set up, but Dr. J and Bird weren't there yet. About fifty to a hundred kids were standing around wondering what was going to happen, and then Bird and The Doctor pull up and get out of the car. I swear, I never saw so many big eyes in my life."

Despite what the package for *One-on-One* says, designing the game wasn't just a ploy by Hammond to meet his idols. Nor were there hours and hours of game films to study. Actually, Hammond had a videotape of the 1982 all-star game that had lots of Erving-Bird footage on it. Apple hobbyists who have contacts in the public domain software market may soon run across one of Hammond's disks that resulted from that tape.

Digitized Dunk. The tapes were used mostly so Hammond could study animation. He took one sequence of an Erving dunk, digitized each frame, and put it on the Apple. Then, with a RAM disk, he worked the sequence to go through five frames a second, resulting in a full-screen, slow-motion hi-res animated dunk. Now *that's* a basketball fan.

These days, you can find Hammond practicing jumpers and dunks in his yard or playing pick-up games in his community men's league. Ironically, he never played a minute of high school ball. Certainly talented enough (he's playing college basketball now) and tall enough (a respectable six feet, two inches), Hammond didn't feel it was the thing for him in those days.

"All the guys got on my case for not playing," says Hammond. Instead, he played tennis, a sport that's "not as exhausting as basketball. Besides, I didn't have to go to as many practices; team sports take too much time."

Hammond doesn't like to spend more time than necessary to do things. It took him only about two days to learn Basic—the class moved too slowly for him. He realized then that computers might be an area he could excel in—not that he stunk in anything else (quite the contrary). An honor student, he graduated twentieth in his class. He received the Bank of America award for proficiency in math. He was good in physics. And English. And everything else. Unfortunately, the subject of computers wasn't taught as well as he would have liked.

Everything he learned about 6502 assembly language was self-taught. Classes were too slow; reading allowed him to move at a faster pace. His first programming project was a collaboration with friend Rorke Weigandt, whose Apple they worked on.

"It took us forever to do anything on the Apple because we didn't know anything about it. First we wrote a game in Basic and tried to convert it to machine language. With all the things I know now, it seems weird to think that at one time I didn't know assembly language; it's almost second nature now."

Rookies of the Year. Today, when Hammond looks at the code of *Marauder* (his first published program), he's embarrassed. "I'm surprised the game worked, it was so badly written." *Marauder's* documentation says that "it took the full power of the 48K Apple II to make it a reality." Translation: At the time, Weigandt and Hammond were such inexperienced programmers that they needed 48K to write a program that should have taken a lot less. "When we rewrote it for the Atari, it took only 20K," says Hammond with a chuckle.

Finding a software publisher has been a game of island-hopping for Hammond. *Marauder* was published by On-Line Systems (now Sierra On-Line); *Maze Craze Construction Set*, an eight-week project, was published through Data Trek; *Night Flight* was published by the pirate market and has been given to Sweet Micro Systems to use as a demo vehicle for its Mockingboard. But now Hammond seems to have found his niche.

This guy is in love with Electronic Arts. "The people there are real professionals who know what they're doing. Joe Ybarra is really a great guy to work with. I know what he expects from me, and I know what I can do. Even if I don't want to do something, he'll demand it of me for the good of the project. It's the first high-quality company that's come out of the software era; they're reviving the industry."

Kemal Atatürk Had a Whole Menagerie Called Simon. Beyond the work relations Hammond has with Ybarra, Trip Hawkins, and others, there are other reasons for his deep satisfaction. He likes the way everybody at Electronic Arts, from president to receptionist, is interested in him as Eric the person as well as Eric the programmer. During the weeks that he was working at Electronic Arts's office, he and Ybarra would occasionally shoot baskets together, for no other reason than pure recreation.

And recreation is something Hammond can't get enough of. "Sometimes after being cramped up for a whole day working on a routine, it's good to get out and run your butt off." Basketball is his passion, but recently he's been turning into more of a southern California beach bum, going on scuba excursions with friend Michael Abbot (*Hard Hat Mack*).

Even in San Diego, the sun doesn't last forever. Indoors, Hammond is quite an accomplished musician; he's studied piano for twelve years and also plays the bass, drums, and just about every other instrument used for vocal accompaniment ("Dad's a choral director; he taught me how to play so I could help him out in rehearsals. That was pretty smart, I thought").

So now he's off to college, all ready to nurture his career in computers, right? Wrong. Ten years down the road he sees himself in the computer biz, but as for the immediate future he plans to major in music.

Blocked Shots and Rebounds. Hammond has had a lot of chances to capitalize on his talents, but for various reasons he hasn't. With college ahead of him, he now has a second chance and is going for it.

You just can't hit a guy like that. And even if you did, he'd get a free throw.

No Comment

Adam Antics

Where did you get Benjamin Balsam—Atari or Commodore? That was a very misleading review (Infomania, January-February); I should know, I have an Adam. He was correct about the printer noise, and anyone who leaves a digital tape (or disk, for that matter) in a drive when turning the machine on or off deserves what he gets. Now to the incorrect statements: A six-sentence file may take forty-five seconds to save, but so does a six-*page* file. The time is spent in positioning, not data transfer. I don't know where you got the idea that Coleco's version of Applesoft is slow, but in the last couple of issues of *Creative Computing* they have had a benchmark program, so I ran it on my Adam. Results: forty-eight seconds. Apple IIe and Apple II Plus both rate at one minute fifty-three seconds! I sure would like to find out how you did your timings.

Last but not least, the major untruth that caught my eye and prompted this letter: "There seems to be no way to access assembly language through Basic." Sure, the manual is terrible, but you could at least take the time to look at it. Ever heard of *usr*, *brun*, or *call*? Try looking in the Basic manual before making airhead statements like that.

Send Ben back to Atari or Commodore (or Apple or wherever). Your magazine deserves better reviews than that!

Russel P. Neal, Tempe, AZ

Benjamin Balsam replies:

Where did you get this guy, Coleco? Mr. Neal, I can't alter the fact that you shelled out $700 for an Adam. No need to be defensive about it. In some of my more human moments, I have been known to turn a computer on or off with disk in drive. Thanks, but I'd rather not be punished for it. My assertion that "Smart Basic" runs slower than Applesoft was based on running the following program:

```
10 HGR
20 HCOLOR = 3
30 FOR I = 0 to 191
40 HPLOT 0, I to 200, I
50 NEXT
```

It took the Adam twenty seconds to draw 192 horizontal lines; it took five seconds on an Apple II. As for assembly language, my Adam "Smart Basic" manual makes absolutely no mention of *usr*, *brun*, or *call*. Perhaps you have a different manual. Even if the *call* statement does exist as an undocumented feature, there is no documentation explaining which memory (if any) is available for use by an assembly language program.

Whether the reason the Adam took forty-five seconds to save a six-sentence file was because of positioning or any other element is irrelevant; the fact stands.

Qua Maze?

Regarding the ongoing Apple versus Atari graphics capability argument, I'd like to see the Atari do Stickybear graphics. On the other hand, I doubt the Apple could support a decent *Pole Position* like the Atari's. Apparently, neither computer has an edge in graphics capability. They are just two different types of excellent game machines.

To Ultrasoft, in case they are reading this: If you people dare put another idiotic animated maze in *Mac Steele in Outer Space*, I will personally autodial your adventure hot line with the Apple-Cat voice synthesizer, which will be programmed to ask how to get out of the plane in *Mask of the Sun*. Seriously, though, these mazes are a royal waste of time. They prove nothing other than endurance and luck. Were it not for the mazes, your series of animated adventures would surpass those of Sirius in quality; as they stand, they are a distant third, behind both Phoenix and Sirius.

By the way, what's with the National Defense Mechanism number? Oh well, enough rambling. Happy gaming.

Theodore Chen, Potomac, MD

Teacher Gets Suspended

I wish to express my appreciation for your No Tipping contest and to thank you for the prize for my tips. With this unplanned windfall, I will get the game *Suspended*. I'll be looking for your hints on playing it.

With the help of your hints, I reached the 350th point and was carried out of the Adventurer's Repository on the elves' shoulders, and I hope my hints can help someone else through *Zork* and open the Stone Barrow southwest of the mailbox.

I've been teaching computer courses since 1966 and I get implicit and explicit criticism for allowing my students to play computer games. Now, those who cannot see the slightly abstract benefit of games as a way to learn problem-solving methods will see the direct benefit of a prize check.

David Dunmire, Linwood, NJ

Of Scams and Stowaways

Let's be blunt. Where's the filthy lucre? I sent in some hints, they got published, and they made volume four hugely successful. I want my cut. I've been good. Honest. I had nothing to do with the lost satellites. I suspect it was Lord British, jettisoning weight equal to his own so the orbiting vectors would work out. Anyway, happy trails.

Roger Carlson, New Britain, CT

While Mr. Carlson's contributions to the Zorktalk section of last issue's No Tipping were indeed swell, they were not that swell. They appeared under "general pleas and assistance," not "bounties." You licks your stamp and you takes your chances. Everyone clear on that?

The Paying Elite

First, after reading your publisher's statement, I found out that (at most) eight thousand people pay for your magazine, while about forty-two thousand get it free. I'm glad to be one of those eight thousand, as your mag is worth it, but how the hell do forty-two thousand get it free and I don't know about it?

Second, I was playing *Lode Runner* yesterday when I discovered that I could have unlimited men by pushing control-shift-P and go to any level with control-shift-N. Therefore, I respectfully submit my high score of at least twenty million.

Third, I generally get *Softline* about four days before the high-score deadline. This fact is overwhelmingly disturbing to me. Maybe you can do something about it.

Doug Kremer, Rochester, NY

First, the days of those subscribers still living off their subscriptions from the time when this august journal was a freebie are numbered. Second, the control-shift stuff in Lode Runner, *along with instructions on how to create your own levels, is in the documentation. Lose your copy? Third, the post office has all at their mercy; the high score deadlines are just the times when we pull in the nets with all the accumulated scores wriggling around in them. Send them in any old time.*

8 to 12

Here's a little tidbit you might wish to pass along to readers who own an Atari 1200XL and have problems with some disks for the 800 not running on the 1200s. Atari has produced a disk that loads the 800 operating system in the 1200XL, which allows 1200 owners to run 800 software. They have to call Atari at (800) 672-1404 to obtain this disk . . . and, by crackee, it's free!

Hartley Lesser, Hayward, CA

HIGH SCORES

«Endgame»

Like a young Dick Nixon on the trail of Alger Hiss, Tireless **Tim Toenjes** continues to seek to bring **Eric Gustafson** to ground for his *Night Mission Pinball* score. We were just as concerned when that unbeatable score came in, but Gustafson and family swear it was made in mode 0, the only one acceptable for these listings. Tim, you're welcome to call Eugene, Oregon, information, give Eric a call, and try to get to the bottom of this.

Theodore Chen reports that those three cheat keys on *Cannonball Blitz*, *AE*, and *Alien Ambush* that we refrained from printing last issue were taken from a list of 153 cheat keys that he had generously contributed to the ASCII Networks Bulletin Board, and were printed in the Washington Apple Pi newsletter without his permission. **Mike Palmieri** wants to know "who the - - - - does Theodore Chen think he is, sending in that low score" on *Robotron 2084*. **Thomas E. Burns** wants to know how Theodore Chen got a score of 120,100 on wave 10 of *Robotron*, while he, Burns, went through 17 waves and scored only 30,800. We'll soon find out, as asking Mr. Chen to write to us is like asking the sun to rise.

Kudos for Best Verification to "computer mom" **Kit Snider** for her *Gumball* and *Lode Runner* scores, witnessed by her son, **David "Midnight Magic" Snider**. Go ahead—challenge it.

Muted kudos to **Bud Foltyn**, who made us stay up late on deadline night so he could call in his ever-increasing score on *Dino Eggs* every hour on the half hour, beating the high score held by the esteemed editor of this very journal by as wide a margin as humanly possible. **Brian Walter** just mailed his in, and questioned said editor's extremely high score on such a low level, noting that its achievement would seem possible only by "killing all the spiders, who score only menial points." Some people happen to like menial points.

Wouldn't you know—after months of nothing, the minute we try to unobtrusively get rid of *People Pong*, we get a high score for it. Well, as San Francisco remarked about Oakland after the earthquake, "there are some things even the earth won't swallow." If you feel you must play this game, you'd be better off keeping that fact to yourself, n'est-ce pas?

Then there's **Reginald Greene** of Carson, California, as well as **Michael Paci** of Trenton, New Jersey, who got scores of 44,090 and 112,000 (verified), respectively, on the games that they played. There's just one more thing you need to let us know, guys. . . .

GAME PUBLISHER	SCORE	PLAYER
A.E. (Apple), Broderbund	*378,500	James Feinberg, Chapel Hill, NC
A.E. (Atari), Broderbund	*77,700	Bob Albright, York, PA
Action Quest, JV Software	*20/95	Michael Lurie, Lincolnshire, IL
Alien Ambush, Peter Fokos	*105,380	Mike Hammer, Gaithersburg, MD
Alien Defense, Soft Sector Marketing	*1,115,320	Chris Athanas, Poolesville, MD
Alien Invasion, Computermat	*10,880	Arturo Rivera, Coamo, Puerto Rico
Alien Munchies, Gentry	*13,008	Wayne R. Hendeer, Jr., Williamsville, NY
Alien Rain, Broderbund	*2,784,900	Lamonte Cates, Chapel Hill, NC
Alien Swarm, In-Home	*1,055,000	Stephen Pitt, Sunnymead, CA
Alien Typhoon, Broderbund	*886,480	Clark Alyea, Bloomington, IN
Anteater, Romox	*301,855	Edouard D. Cournoyer, Atlanta, GA
Apple Cider Spider, Sierra On-Line	*Level 6/1,253,000	Doug Zavodny, Bloomingdale, IL
Apventure to Atlantis, Synergistic	2,446	Brad Foltyn, Newton, NJ
Ardy the Aardvark, Datamost	*37,860	Jan Bossing, Poway, CA
Arena 3000, Med Systems	*1,758,250	James Feinberg, Chapel Hill, NC
Argos, Datamost	*79,050	Klaus Liebold, Poway, CA
Artesians, Rena-Soft	*13,178	Thomas Kirk Almroth III, Huntington Beach, CA
Bandits (Apple), Sirius	*1,689,890	James Feinberg, Chapel Hill, NC
Bandits (Atari), Sirius	*202,680	Jonathan Hopwood, Glastonbury, CT
Bats in the Belfry, Phoenix	*3,098	Mark Hessman, Andover, MA
Battlesight, Versa	Level 5/1,260,500	Michael Susor, San Francisco, CA
Bezare, Southwestern Data	*164,602	Carl Webb, Vista, CA
Blackout, MK Systems	224,630	Theodore Chen, Potomac, MD
Blister Ball, Creative Computing	(1 Player) Level 11/1,271	Bruce Mah, Fresno, CA
Blue Max, Synapse	*Novice/30,180	Wade Tweitmann, Hawthorne, FL
Bolo, Synergistic	*Level 9/10,017	Thomas Kirk Almroth III, Huntington Beach, CA
Bouncing Kamungas, Penguin	*88,994	David Bird, Thousand Oaks, CA
Bristles, First Star	*Level 4/14,130	David Zeve, Dallas, TX
Bug Attack (Atari), Cavalier	*54,385	John Thorning, Bayonne, NJ
Bug Attack (Apple), Cavalier	*1,000,924	James Feinberg, Chapel Hill, NC
Bug Battle, USA	*26,648	Klaus Liebold, Poway, CA
Buried Bucks, Analog	Level 27/274,231	Mike Kerekes, Colorado Springs, CO
Buzzard Bait, Sirius	*38,360	Jim Surine, Saint Paul, MN
Canyon Climber (Apple), Datasoft	*480,800	Mike Berry, Santa Ana, CA
Caverns of Callisto, Origin	*44,950	Greg Reade, Phoenix, AZ
Caverns of Freitag, Muse	*Level 9/158,142	Donald Stevenson, Shreveport, LA
Centipede, Atari	*5,445,080	John Mellott, Orlando, FL
Chicken (Atari), Synapse	*312,010	Karen Sowell, Jackson, MS
Computer War, EMI	*4,721	Toby Crew, Southbury, CT
Cosmic Fighter, Big Five	*532,110	Chris Athanas, Poolesville, MD
Crisis Mountain (Apple), Synergistic	*367,596	Wei-How Chung, Seal Beach, CA
Crisis Mountain (Atari), Synergistic	*186,403	Frank Pennisi Jr., Walnut Creek, CA
Crush, Crumble, and Chomp, Epyx	*2,393,410	Chris Athanas, Poolesville, MD
Cytron Masters (Apple), Strategic Simulations	*Level 3/27,200	Ken Mock, Walnut Creek, CA
David's Midnight Magic (Atari), Broderbund	*2,924,610	James Feinberg, Chapel Hill, NC
Defender, Atari	*19,620,725	Michael Smart, Annandale, VA
Defense Command, Big Five	*178,390	Chris Athanas, Poolesville, MD
Demon Attack, Imagic	*154,475	Barry Landgarten, Douglaston, NY
Demon Seed, Trend	*32,330	Chris Athanas, Poolesville, MD
Dig Dug, Atari	*797,110	Paul Cunningham, Oklahoma City, OK
Dino Eggs, Micro Lab	Level 9/916	Bud Foltyn, Newton, NJ
Donkey Kong, Atari	Level 47/2,285,700	Mike Rosefield, Pleasanton, CA
Double Trouble, Bez	*130,340	Mark Hessman, Andover, MA
Dragon Fire, Dakin5 Level 10	*Level 5/972,000	Brian Service, Exton, PA
Drelbs, Synapse	*51,730	Brad Labine, Concord, MA
Drol, Broderbund	*Wave 3, Level 3/23,010	Thomas Kirk Almroth III, Huntington Beach, CA
Epidemic, Strategic	*783	Alan Chang, Honolulu, HI
ET, Atari	*182,000	Warner Young, Fresno, CA
Evolution, Sydney	*17,284	Mark Hessman, Andover, MA
Exterminator, Winner's Circle	*Level 4/28,521	Klaus Liebold, Poway, CA
Firebug, Muse	*Level 37/10,060	Steven Kahn, Jamaica, NY
Flight Simulator (British Ace), Microsoft	85	Joey Rumsey, Princeton, NJ
Flip and Flop, First Star	*Level 14/16,775	Kevin Moore, Dallas, TX
Flying Saucers, Radio Shack	*1,210	Rajendra Singh, Chapel Hill, NC
Formula 1 Racer, Gentry	Intermediate/53,130	Theodore Chen, Potomac, MD

Graphics: Nancy James

GAME PUBLISHER	SCORE	PLAYER
Frenzy, Softape	*140,042	Klaus Liebold, Poway, CA
Frogger (Apple), Sierra On-Line	*55,450	Lamonte Cates, Chapel Hill, NC
Gumball, Broderbund	*71,710	Kit Snider, Franklin, MI
Handy Dandy, Gentry	*20,050	Clark Alyea, Bloomington, IN
Hard Hat Mack, Electronic Arts	*971,300	Chris Amendola, Huntington Beach, CA
Highrise, MicroLab	*Level 79/104,895	Clark Alyea, Bloomington, IN
Human Fly, CPU	*225	James Handy, West Chester, PA
Ice Cave, Winner's Circle	Level 5/39,300	Bud Foltyn, Newton, NJ
Jawbreaker II (Apple), Sierra On-Line	*1,348,010	Robert Clark, Clayton, MO
Jellyfish, Sirius	*448,100	Clark Alyea, Bloomington, IN
Joust, Atarisoft	Novice/1,333,050	Hartley Lesser, Hayward, CA
Jovian, Computer Shack	*29,720	Chris Athanas, Poolesville, MD
Juice, Tronix	*24,210	David Zeve, Dallas, TX
Jump Jet, Avant-Garde	*74,730	Greg Reade, Phoenix, AZ
Jumpman, Epyx	*Loop/Level 25/67,900	Ben Cohn, Saint Louis, MO
Jumpman Jr., Epyx	Speed 2/405,000	J. Ralph Sowell, Jr., Jackson, MS
Jupiter Lander, Commodore	47,300	Chris Chekel, Granada Hills, CA
K-Razy Kritters, CBS	*368,200	Tom Buchmann, New Brighton, MN
K-Star Patrol, CBS	*53,100	Alro Andersson, San Jose, CA
Labyrinth, Broderbund	*210,500	Darcy Higden, Livonia, MI
Lady Tut, Progame	*Level 28/72,310	Hank Johnstone, Fountain Valley, CA
Laf Pak: Mine Sweep, Sierra On-Line	*9,533	Chris MacChesney, Fresno, CA
Lancaster, Silicon Valley	*12,560	Buell Hollister III, Shelburne, VT
Lemmings, Sirius	*143,435	Jerry Ingram, Drayton Plains, MI
Livewire, Analog	*Level 16/68,400	Mike Jelarcic, Citrus Heights, CA
Lode Runner, Broderbund	*Level 512/5,297,750	James Feinberg, Chapel Hill, NC
Lunar Leeper, Sierra On-Line	*184,620	Chris Amendola, Huntington Beach, CA
Mad Netter, Computer Magic	*61,746	William Chiang, New York, NY
Marauder, Sierra On-Line	*Level 21/503,600	Clark Alyea, Bloomington, IN
Mars Cars, Datamost	*Level 30/64,670	Brian Ballard, Bow, NH
Match Racer, Gebelli	*86.7 Miles	Mike Kerekes, Colorado Springs, CO
Mating Zone, Datamost	Round 5/1,520	Theodore Chen, Potomac, MD
Megalegs, Megasoft	*19,009	Barry Landgarten, Douglaston, NY
Meteor Mission Two, Big Five	*120,680	Chris Athanas, Poolesville, MD
Microbe, Synergistic	17,016	Bob Enna, Huntington Beach, CA
Ming's Challenge, Micro Fun	*45,250	Buell Hollister III, Shelburne, VT
Missile Command, Atari	*10,985,275	John Mellott, Orlando, FL
Mountain King, CBS	Level 6/78,730	Dean Rekich, Prospect, PA
Mr. Cool, Sierra On-Line	Level 2, Round 12/749,337	Derrick Bang, Davis, CA
Mr. Robot and his Robot Factory, Datamost	*Screen Q/172,670	Mike Kerekes, Colorado Springs, CO
Ms. Pac-Man, Atari	*66,240	Reginald Greene, Carson, CA
Nautilus, Synapse	*(1 Player) 168,800	Dan Perrigan, Norwalk, CA
Necromancer, Synapse	*204,651	Ron White, Gt. Barrington, MA
Neutrons, Dakin5	*Level 10/35,000	Theodore Chen, Potomac, MD
Night Crawler, Softape	289,407	Michael Tsao, Watsonville, CA
Night Mission Pinball, SubLogic	*749,653,240	Eric Gustafson, Eugene, OR
Night Stalker, Mattel	7,100	Theodore Chen, Potomac, MD
Nightfall, Omega	*Level 1/14	Craig Morris, Menlo Park, CA
Nightmare Gallery, Synergistic	*878,850	James Feinberg, Chapel Hill, NC
Norad, Southwestern Data	*Level 2/14,840	Wayne Karolow, Danvers, MA
Odyssey, Synergistic	*1,173,400	Casey Lamson, Tucson, AZ
Oil's Well, Sierra On-Line	*Level 2/15,860	Kevin Moore, Dallas, TX
O'Riley's Mine, Datasoft	$74,000	Theodore Chen, Potomac, MD
Outworld, Tensor Technology	*102,930	Steve Gazis, Queens, NY
Pac-Man, Atarisoft	*Board 5/42,860	Theodore Chen, Potomac, MD
Pandora's Box, Datamost	*33,750	Colin DuPlantis, Bloomington, IN
Pathfinder, Gebelli	*26,271,422	Mike Buchmann, New Brighton, MN
Pest Patrol, Sierra On-Line	*Level 29/14,275	Lamonte Cates, Chapel Hill, NC
Phaser Fire, Datamost	*9,305	Mike Kerekes, Colorado Springs, CO
Pillbox, Lord of the Games	30,700	Neal Naydo, Montery Park, CA
Plasmania, Sirius	*Expert/72,523	Tim Tant, Northridge, CA
Pole Position, Atari	*Level 1/101,650	Toby Crew, Southbury, CT
Pollywog, Top Notch	*294,765	Matt Yuen, Van Nuys, CA
Pooyan, Datasoft	*1,007,750	Rob Maretsky, West Lafayette, IN
Popeye, Parker Brothers	*118,560	Barry Landgarten, Douglaston, NY
Potholes, Computer Gazette	*1,060,710	Arturo Rivera, Coamo, Puerto Rico
Preppie, Adventure Int'l	*140,120	Kermit Carter, Boise, ID
Preppie II, Adventure Int'l	*Level 5/69,850	Mike Kerekes, Colorado Springs, CO

Yes, there are cheaters, crackers, and rogues among us. But there is also another kind of gamer. There are the game heroes, those who go where none before them has dared to tread, and bring back those vital discoveries that make the world a little bit warmer, a little bit safer for us all. One such is **Joan Haller**. Her game is *Lode Runner*. She played it to level 286 and a score of nearly three million, only to discover that the disk had rolled over and recorded her score at level thirty because designer **Doug Smith** "didn't think anyone would get that high." Undaunted, Haller went right back to it, coolly noting the rollover at level 255 and keeping track of her true position. This is what she found: At level 357 (256 + 101) there are no possible alternative strategies; the guard always reaches the ladder before the player. She escaped with 132 men. Level 392: It's impossible to dig a hole fast enough to get past the guard in the lower right-hand corner. She got out with sixty-one men and continued on to level 428 (256 + 172). Unable to dig a hole fast enough to get past the guard in the center-right cube, and with only sixteen men left, she met her end at level 441, with a score approaching five million. Rest well, Ms. Haller. We'll not soon forget.

And before we do, let's see who won the high score sweepstakes this month. $34.95 goes to each member of the randomly selected trio, and they are:

Apple: H.A. Dickson, *Serpentine*
Atari: Tom Buchmann, *K-Razy Kritters*
Overall: Michael Susor, *Battlesight*

Send your full address to St.Gains, Box 60, North Hollywood, CA 91603, and the cash is yours.

Hall of Fame

Airstrike, Dean Rekich, 23,730
Astro Chase, Bob Albright, 448,947
Baja Buggies, Jim Burkhard, 67.40 MPH
Bell Hop, Curtis Weber, 56.31
Bez Off, Brian Ford, 160,844
Boa, Steven Schwartz, Level 9/43,830
Breakout, Fred Di Cesare, Double/2,656
Breakout, Fred Di Cesare, Progressive/4,580
Bug Attack (Atari), John Thorning, 54,385
Cosmo Mission, Kaveh Ghazi, 8,540
Cricketeer, Brian Gieler, 55,360
Crime Wave, Buell Hollister III, 10,600
Crolon Diversion, Chris Athanas, 7,300
Cyclod (Apple), Rich Dellinger, 966,518
Dawn Patrol, Murry Burns, 18,207
Deluxe Invaders, Bobby Moy, 26,850
Escape from Vulcan's Isle,
Roe R. Adams III, 13,580
Falcons, Joel Frickhoeffer, 301,750
Fly Wars, Russell Willis, 98,190
Fort Apocalypse, Darryl May, 190,809

Frazzle, Gary Penacho, 56,850
Free Fall, Alan Chang, 28,432
Galactic Attack, Joseph Rossi, 6,320
Gorgon, Clark Alyea, 85,220
Grapple, Mort Goodman, 215,500
Guadalcanal Campaign, Greg Munsell, 2,655
Guardian, Tom McDonald, 128,180
Heli-Lifter, Chris Athanas, 31,895
High Orbit, Gilbert Huang, 28,860
Ice Demons (1 player), Steve Williams, 48,990
Ice Demons (2 player), Dan Knight and Steve Williams, 612,290
Journey to the Planets, Ron Stanley, 4,205
Juggler, John Osborne, 961,460
Kid Grid, Bobby Moy, 33,860
Laf Pak: Creepy Corridors, Wilbur Luo, 9,740
Laf Pak: Space Race, Steve Rothenberg 14,500
Laser Defense, Chris Athanas, 44,960
Lazer Silk, Gilbert Huang, 896,550
Leaper, Travis Beauboin, 12,400
Microwave, Richard Olsen, 53,499
Miner 2049er (Apple), Gary Schmitt, 615,310
Miner 2049er (Atari), Robert Rendahl, Zone 8, Station 10/582,205
Money Muncher, Tom Luhrs, 705
Moon Landing, Kaveh Gazi, 1,017
Mutant, John Osborne, 2,500,266
Paddle Pinball, Chris Athanas, 215,010
Penetrator, Chris Athanas, 15,810
People Pong, Klaus Liebold, 18,925
Phobos, Bobby Moy, 172,200
Picnic Paranoia, Jonathan Hopwood, 144,360
Pie Man, Kevin Rehak, 425
Pogoman, Jim Mahoney, 673,430
Pollywog, Matt Yuen, 294,765
Raiders of the Lost Ring, Jeff James, 191,210
Rescue at Rigel, Bruce Schlickbernd, 1,557
Rocket Command, Pepe Caridad, 1,028,385
Russki Duck, Brian Schneider, 375,000
Shark Trap, Matt Yuen, 10,705
Shuttle Intercept, Clark Alyea, 13,900
Sigma 7, Blane Sibille, Jr., 1,800
Snake (IBM), Peter Quinn, 468
Snake, John Picken, Progressive/282,550
Snakebyte, Barry Landgarten, Level 110/320,670
Snapper, Jean-Michael Decombe, 144,759
Space Avenger, Chris Athanas, 10,350
Space Invasion, Chris Athanas, 6 Racks/74
Space Warp, Chris Athanas, Level 8/255
Star Island, Tim Caldwell, 87,370
Star Maze, David Sneid, Level 16/172,925
Stellar Escort, Chris Athanas, 19,300
Strike Force, Chris Athanas and Neil Kincaid, 433,900
Succession, Bill Elvey, 10,300
Survivor (1 player), David Schlater, 311,920,
Tharolian Tunnels, Steven Gazis, 134,607
Thief, Dennis Heaton, 303,600
Tubeway, Jim Stockla, 90,801
Tumblebugs (Atari), Tad Hamilton, 38,785
Turmoil, Bobby Moy, 83,540
U-Boat Command, Peter T. Clark, 2,189
Wavy Navy, John Osborne, 88,780
Zargs, Christopher Chan, 3,715
Zero Gravity Pinball, Joel Frickhoeffer, 19,100

GAME PUBLISHER	SCORE	PLAYER
Protector II, Synapse	*236,050	Brynn Ostin, San Jose, CA
Q-Bert, Parker Brothers	*130,940	Barry Landgarten, Douglaston, NY
Qix, Atari	*727,794	Paul Cunningham, Oklahoma City, OK
Quadrant 6112, Sensible	*300,100	Clark Alyea, Bloomington, IN
Queen of Hearts, SSI	*824,800	Mike Keck, La Palma, CA
Randamn, Magnum	*13,154	Wayne Karolow, Danvers, MA
Rear Guard (Apple), Adventure Int'l	*321,475	Peter Alyea, Bloomington, IN
Repton, Sirius	*9,874,650	Keith Armonaitis, Rochelle Park, NJ
Rings of Saturn, Dakin5/Level 10	Level 4/636,968	Pat Volk, Pittsburgh, PA
River Raid, Activision	*46,750	Barry Landgarten, Douglaston, NY
Roadblock, SubLogic	374	Theodore Chen, Potomoc, MD
Robot Attack, Big Five	*27,380	Chris Athanas, Poolesville, MD
Robot Battle, Universal	*11,325,500	James Feinberg, Chapel Hill, NC
Robotron: 2084, Atarisoft	*Wave 99/1,718,250	David Brown, San Mateo, CA
Roundabout, Datamost	*Level 11/10,260	Mark Hessman, Andover, MA
Sammy Lightfoot, Sierra On-Line	*281,080	Alan Chang, Honolulu, HA
Sanitron, Winner's Circle	13,390	Theodore Chen, Potomac, MD
Scarfman, Cornsoft	*453,400	James Feinberg, Chapel Hill, NC
Sea Dragon, Adventure Int'l	*463,200	Doug Zavodny, Bloomingdale, IL
Seafox, Broderbund	*202,850	Elizabeth Crowe, Saginaw, MI
Sentinel, Med Systems	*64,450	Tom Myers, San Jose, CA
Serpentine, Broderbund	*Level 275/75,433,900	H.A. Dickson, Dallas, TX
Shadow World, Synapse	*111,200	Robert Proudfoot, Long Beach, CA
Shamus: Case 2, Synapse	*400,805	Mike Kerekes, Colorado Springs, CO
Sheila, H.A.L. Labs	*155,917	Scott Brown, Riverside, CA
Shooting Gallery (Atari), Analog	*28,620	Pamela Duke, Lompoc, CA
Snapper, Silicon Valley	*297,946	Alan Chang, Honolulu, HA
Sneakers (Atari), Sirius	*Level 5/76,402	Frank Pennisi Jr., Walnut Creek, CA
Space Album: Asteroids, California Pacific	7,212	Kyle Naydo, Monterey Park, CA
Space Eggs (Atari), Sirius	*4,740	Michael Zenner, Portland, OR
Space Kadette, Funtastic	*7,552	James Feinberg, Chapel Hill, NC
Space Station Zulu, Avalon Hill	Level 1/786	Roe Adams III, Hyde Park, MA
Spare Change, Broderbund	*Level 81/2,876,700	Tim Tant, Northridge, CA
Spectre, Datamost	Level 6/23,230	Doug Zavodny, Bloomingdale, IL
Spider Raid, Insoft	6,700	Michael Susor, San Francisco, CA
Spy Strikes Back, Penguin	*5,321	Buell Hollister IV, Shelburne, VT
Star Cruiser, Nasir	*34,280	Tom McDonald, Huntsville, AL
Stargate, Atarisoft	Wave 3/28,150	Theodore Chen, Potomac, MD
Star Trek, Sega	*Sector 2.4/98,750	Theodore Chen, Potomac, MD
Stellar 7, Software Entertainment	*38,490	Patrick Michael Hunter, Eugene, OR
Stickybear Bop, Xerox Educational	Level 5/265,300	Michael Susor, San Francisco, CA
Stratos, Adventure Int'l	*Expert/91,100	Mike Kerekes, Colorado Springs, CO
Super Bunny, Datamost	89,980	Steve Gazis, Queens, NY
Super Puckman, ADO Software	*80,010	Stanley Chen, Potomac, MD
Survivor, Synapse	*(1 Player) 742,430	Bill Fedak, Union Lakes, MI
Survivor, Synapse	*(3 Player) 375,240	Alro Andersson, Warren Benson, and Brynn Ostin, San Jose, CA
Swashbuckler, Datamost	*6,702	Frank Pennisi Jr., Walnut Creek, CA
Sword of Fargoal, Epyx	*Level 11/116,226	Bill Brown and Barry Spencer, Brighton, MI
Tail of Beta Lyre, Datamost	*Sector 5/12,622	David Zeve, Dallas, TX
Talon, SE Software	*114,500	Mark Hessman, Andover, MA
Target Command, Computermat	*28,860	Arturo Rivera, Coamo, Puerto Rico
Taxman, H.A.L. Labs	*1,217,350	Eric Morson and Phillip Morson, Stamford, CT
Teleport, Cavalier	*Level 109/127,800	James Feinberg, Chapel Hill, NC
Tharolian Tunnels, Datamost	*137,385	Lamonte Cates, Chapel Hill, NC
Thunder Bombs, Penguin	*55,910	Chardin Wilson, Kremlin, MT
Thunderbird, Urban Software	*1,000,800	Brian Hazard, Huntington Beach, CA
Triad, Adventure Int'l	*Level 9/102,870	Anand Kaucherla, Sugarland, TX
Tubeway II, Datamost	*97,730	James Feinberg, Chapel Hill, NC
Tunnel Terror, Adventure Int'l	*751,790	James Feinberg, Chapel Hill, NC
Vindicator, HAL Labs	*Level 43/461,425	Frady Shiang, Rancho Palos Verdes, CA
Viper, RDA Systems	*7,917	Doug Zavondy, North Hollywood, CA
Vortex, Datamost	*48,230	Lee Stafford, Phoenix, AZ
Wall War, Sierra On-Line	*68,940	William Chiang, New York, NY
Warp Destroyer, Piccadilly	*96,540	David Levine, Potomac, MD
Wayout, Sirius	*7	Rod Bonios, Van Nuys, CA
Wizard of Wor, Roklan	*1,456,200	James Feinberg, Chapel Hill, NC
Zaxxon, Datasoft	*785,400	James Feinberg, Chapel Hill, NC
Zenith, Gebelli	*69,650	Jim Stockla, Shelton, CT
Zombies, Bram	*(1 Player) Easy/4,674	Michael Jelarcic, Citrus Heights, CA
Zombies, Bram	*(2 Player) 1,113	Kevin Moore and David Zeve, Dallas, TX
Zoo Master, Earthware	*15,113	Buell Hollister III, Shelburne, VT

A FOUR STAR WINNER

★ ★ ★ ★

- The Video Game Update
- Computers & Electronics

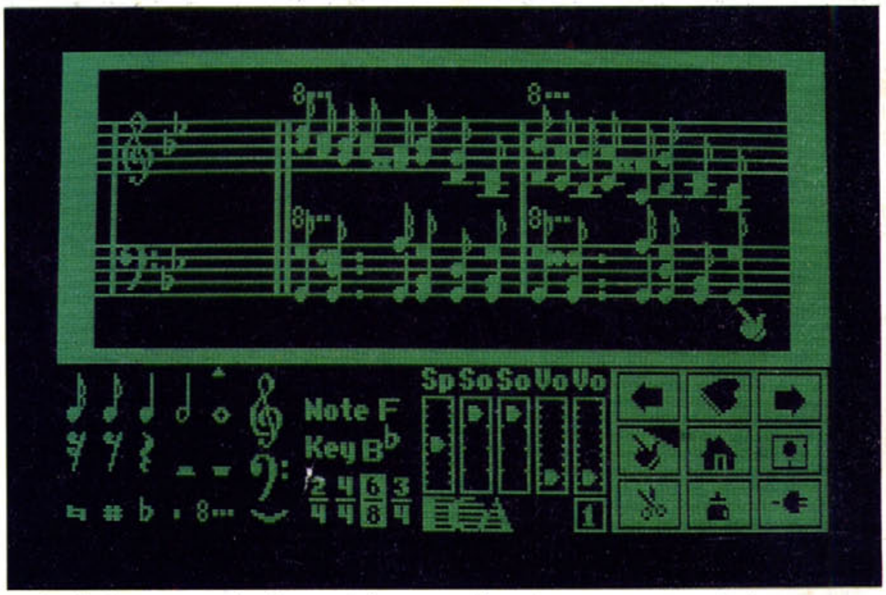
"QUEST FOR TIRES is a delightfully original game with very well animated, cartoon-like graphics and excellent play action."

- The Video Game Update
December, 1983

Sierra ON-LINE TM

The END *of* DINKETY-DINK-DINK.

Announcing the first computer music program that actually sounds like music.



LET'S FACE IT. Up till now, music programs for your home computer have all sounded, well, pretty lame. There were the ones that resembled little electronic music boxes, remember? And then there were those that sounded like so many burps.

Enter Music Construction Set.™ It's the first music program that really makes use of the power of that machine you've got. If you're a serious student, this means you'll be able to work with an intricacy and range of sound quality you've never heard before on a computer. And if you know nothing about music, you'll find something even more important. Namely, that this thing is simple enough to be a lot of fun.

Take a good look at this screen because it, you, and a joystick are the whole story here.

That's you at the right end of the staff of notes – the little hand. Move the joystick, and you move the hand. Use it to carry notes up to the staff. Lay in rests, signatures, clefs, then point to the little piano in the lower right and listen, because you'll hear the whole thing played back.

Move those little scales in the middle up and down to vary the music's speed, sound quality, and volume. Use the scissors to cut out whole measures, then use the glue pot to paste them in somewhere else. Got a printer? Great. Print the score out and show it off to your friends.

But what if you're not up to writing your own stuff yet? No problem. There are twelve pieces of music already in here, from rock 'n roll to baroque. They're fun to listen to, and even more fun to change. (Apologies to Mozart.)

The point is, the possibilities are endless. But if you're still skeptical, visit your nearest Electronic Arts dealer and do the one thing guaranteed to send you home with a Music Construction Set in tow.

Boot one up. Point to the piano. And listen.

NOW AVAILABLE FOR
ATARI HOME COMPUTERS

ELECTRONIC ARTS™

MUSIC CONSTRUCTION SET is now available for Apple II, II+, IIe, and Commodore 64 computers. The Apple version, with a Mockingboard,™ plays chords of up to six notes each The Commodore version plays chords of up to three notes each. Apple is a registered trademark of Apple Computer. Commodore is a registered trademark of Commodore Business Machines, Inc. For more information about Electronic Arts, write us at 2755 Campus Drive, San Mateo, CA 94403 or call (415) 571-7171. For a free catalog, send a stamped, self addressed #10 envelope.